International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 06 | June - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

Intrusion Detection and Prevention System Using Machine Learning

Miss. Shruti Ratan Jadhav, Miss. Shital Dilip Moule, Mr. Anam Chunnilal Nagpure, Mr. Nayan Gokul Sonawane

Under the guidance of

Prof. Deepali S. Surywanshi.

DEPARTMENT OF COMPUTER ENGINEERING
BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)
MET's Institute of Engineering,
Adgaon, Nashik-422003
SAVITRIBAI PHULE UNIVERSITY, PUNE Nov 2022

Abstract

The idea of making everything available readily and universally has led to a rev- olution in the field of networks. In spite of the tremendous growth of technologies in the field of networks and information technology, we still lack in preventing our resources from theft/attacks. This may not concern small organizations but it is a serious issue as far as industry/companies or national security is concerned. Organizations are facing an increasing number of threats every day in the form of viruses, intrusions, etc. Since many different mechanisms were opted by organizations in the form of intrusion detection and prevention systems to protect themselves from these kinds of attacks, there are many security breaches which go undetected. In order to understand the security risks and IDPS(intrusion detection and prevention system), we will first survey about the common security breaches and then discuss what are different opportunities and challenges in this particular field. The growth of the Internet has no doubt changed the face of world but it has also pointed out various security areas that need to be addressed in order to provide a trustworthy environment for those who are a part of this system or those who wish to be Intrusion detection systems (IDS) have come as a savior but every day new attacks or intrusions provide a challenging atmosphere to even the most powerful tools available.

Chapter 1 Introduction

With the tremendous growth of network-based services and sensitive information on networks, network security is becoming more and more important than ever before. Intrusion detection techniques are the last line of defences against computer attacks behind secure network architecture design, firewalls, passwords, encryptions and personal screening. Despite number of intrusion prevention techniques available, attacks against computer systems are still successful. Thus, intrusion detection systems (IDSs) play a vital role in real time network security. An intrusion is defined as any set of actions that attempt to harm or damage the data which includes a deliberate unauthorised access to the information, manipulate information, or make a system unreliable. Intrusion detection system is a model designed to detect attacks among the various type of packets. It is the process of examining the events which occurs in a computer system or network and analysing

them for presence of intrusions.

To detect unusual malicious activity on a network or system by attackers through Machine learning results in better classification between normal traffic and malicious traffic with a reduced rate of false positive alarms. In today's world one of the most dangerous things in the security of computer or network is illegal access to a computer system. As network applications are growing rapidly, new types of network attacks are on the rise. To manage suspicious activities our system needs to be reformed. Once an attack is identified or unusual behaviour is detected, a notification can be sent to the network administrator. Access systems (IDS) take the path based on the network or host by detecting and diverting attacks. In any case, these products are subject to attack signatures often indicates a malicious or suspicious intent. When the ID is checked these patterns in network traffic are then derived from the network. Most of the techniques used in modern IDs they cannot manage the flexible and complex environment of Internet attacks on computer networks. So, effectively again using adaptive mechanisms that lead to high detection levels, low false alarm levels. Methods of machine learning provide appropriate accounting and communication costs. Matching pattern is fast enough to check the presence of a signature in the order of the incoming package and acquires a malicious behaviour and must meet both your expansion signature number and link speed. There are several ways to use IDS.

Integrating machine learning algorithms into SDN has attracted significant attention. A solution was derived to solve the problems in the KDD Cup 99 by making a plan ex- tensive exploratory research, using the NSL-KDD data set to achieve excellent accuracy entry discovery. The experimental study was performed on five well-known and well- performing individual machine learning algorithms (RF, J48, SVM, CART, and Na¨ıve Bayes). Relationships the feature selection algorithm was used to reduce the complexity of the features, which led to Only 13 features in the NSL-KDD database. A dynamic model of the "Intelligent Access Acquisition System" proposed based on a specific AI method of access acquisition. Strategies that integrate neural networks and abstract thinking have a network profile, using simple data mining techniques to process net- work data. The program includes confusing, abuse and host-based acquisitions. Simple comprehensive rules allow for rules that reflect common ways to define security attacks. There have been a number of methods used by machine learning applications to address the problem of selecting features for access. In the author used PCA to identify space features in the main character area and select features that correspond to high eigen values using the Genetic Algorithm. The same models were re-trained using 13 reduced features to achieve an average accuracy of 98%, 85%, 95%, 86%, and 73% in each model. A deep neural network model was proposed to detect and detect SDN intrusion.

Algorithm used here will mostly be support vector machine. The Vector Support Machine (SVM) machine falls under a supervised learning method, where different types of data are trained from different subjects. In the upper part, SVM creates multiple hyper planes or hyper planes. A hyper plane that properly separates data assigned to different categories over a wide range is considered a leading aircraft. To test genes between hyper planes, the indirect detector uses a variety of kernel functions.

Enhancement among hyper planes is the main goal of these kernel functions such as linear, polynomial, radial basis, and sigmoid. Due to the growing attention in SVMs, outstanding applications have been developed by developers and researchers. SVM plays a major role in image processing and pattern recognition applications.

1.1 Objective

- The Objective of the system is to develop an enhanced application for detection of network intrusion in early stages using SVM classifier which gives the network administrator plenty of time to take precautionary measures in early stages.
- It monitors all traffic on the network to identify any known malicious behavior.
- It is a monitoring system that detects suspicious activities and generates alerts when they are detected

Chapter 2 Literature Survey

Intrusion detection systems (IDSs) play a vital role in real time network security. An intrusion is defined as any set of actions that attempt to harm or damage the data which includes a deliberate unauthorised access to the information, manipulate infor- mation, or make a system unreliable. Intrusion detection system is a model designed to detect attacks among the various type of packets. It is the process of examining the events which occurs in a computer system or network and analysing them for presence of intrusions. .

1. **Network Intrusion Detection System Based On Machine Learning Algorithms**

Author: Vipin, Das Vijaya, Pathak Sattvik, Sharma Sreevathsan MVVNS.Srikanth Kumar T, Gireesh

Description:

This system explains how an efficient and effective system can be developed by making use of various machine learning and artificial intelligence algorithms.The paper talks about the system's currently being used by various organizations which contain some drawbacks and limitations due to improper selection of algorithms which fail to detect some important patterns of malicious users. These drawbacks may cause a serious issue or harm to the organization's security and data of the users it handles.

2. **A hierarchical detection and response system to enhance security against lethal cyber-attacks in UAV networks.IEEE Transactions on Systems, Man, and Cybernetics.**

Author: Sedjelmaci H, Senouci SM, Ansari N

Description:

An intrusion detection system developed by students from MIT takes into consideration mainly two algorithms, Genetic Algorithm (GA) for feature selection and Bagged Classifier with partial decision tree. The data set used in this system for training the model is the NLS-KDD99. The accuracy of this system was pretty satisfactory as it reduced high false alarm and provided great anomaly detection. The main drawback of this system was that it required a high time to build the model as well as to train the model for the system. The network intrusions are needed to be detected at a faster rate so that the network administrator has plenty of time for stopping the attacks and securing the systems.

3. **Comparative Performance Analysis of classification algorithm for Internal Intrusion Detection**

Author: Mohammed Anbar, Rosni Abdulah, Izan H. Hasbullah, Yung- Wey Chong.

Description:

this paper talks about a study conducted namely "Comparison of Classifi- cation Techniques applied for network intrusion detection and classification". The study talks about using multiple different types of machine learning algorithms on the same dataset for determining the accuracy of each algorithm to evaluate the best algorithm for classification of network activities. The algorithms used provcided efficient results and accuracy but couple of them stand out from the other like the Breadth-Forest

Tree (BFTree) approach gave a result of 98.24% , the Na¨ıve Bayes Decision Tree resulted in an accuracy of 98.44% , Random Forest tree returns an accuracy of about 98.34percent. Now use of such multiple algorithms achieved reduction in false positive, but the only drawback the system has is that it needs to be continuously updated by using the dataset that consists of the latest infor- mation and newer attack techniques to train the model and make the system more accurate and precise for detection of new network attacks.

4. A Novel Unsupervised Anomaly Detection Approach for Intrusion Detection System.

Author: Weiwei Chen, Fangang Kong, Feng Mei, GuiginYuan, Bo Li.

Description:

This paper presents, defines distinct attack patterns depicting Distributed Denial of Service (DDoS) attacks against target nodes within wireless sensor net- works for three most commonly used network topologies. A Graph Neuron (GN)- based, decentralized pattern recognition scheme is proposed for attack detection. The scheme does analysis of internal traffic flow of the network for DDoS attack patterns. We stipulate that the attack patterns depend on both the current energy levels, as well as the energy consumption rates of individual target nodes. The results of varying pattern update rates on the pattern recognition accuracies for the three network topologies are included in the end to test the effectiveness of our implementation. Organization and framework security is of foremost significance in the current information correspondence environment. Programmers and gatecrashers can make numerous fruitful endeavours to cause the accident of the organizations and web administrations by unapproved interruption. New dangers and related answers for forestall these dangers are arising along with the frame- work advancement. Interruption Recognition Frameworks (IDS) are one of these arrangements. The primary capability of Interruption Identification Framework is to shield the assets from dangers. It breaks down and predicts the ways of behaving of clients, and afterward these ways of behaving will be viewed as an assault or a typical way of behaving. Network and system security is of paramount importance in the present data communication environment, Hackers and intruders can cre- ate many successful attempts to cause the crash of the networks and web services by unauthorized intrusion. New threats and associated solutions to prevent these threats are emerging together with the secured system evolution. Intrusion De- tection Systems (IDS) are one of these solutions. The main function of Intrusion Detection System is to protect the resources from threats. It analyzes and predicts the behaviours of users, and then these behaviours will be considered an attack or a normal behaviour. We use Rough Set Theory (RST) and Support Vector

Ma- chine (SVM) to detect network intrusions. First, packets are captured from the network, RST is used to pre-process the data and reduce the dimensions. The features selected by RST will be sent to SVM model to learn and test respectively.

The method is effective to decrease the space density of data. The experiments compare the results with Principal Component Analysis (PCA) and show RST and SVM schema could reduce the false positive rate and increase the accuracy.

5. A Survey of Data Mining and Machine Learning methods for cybersecurity intrusion detection.

Author: Anna L. Buczak, Erha n Guven.

Description: In this paper,It protected from many malicious programs us- ing intrusion detection systems. Building an efficient intrusion detection system is very important to protect our system/network from the many forms of mali- cious programs. There are four types of IDS, they are Network intrusion detection system (NIDS), Host-based intrusion detection system (HIDS), Perimeter Intru- sion Detection System (PIDS), VM based Intrusion Detection System (VMIDS). NIDS monitor the network, through which the system is connected, to check and to protect the system from outside unauthorized access. The HIDS always checks the system(host) for any abnormal behavior. If abnormal behavior is observed by HIDS, it immediately takes the necessary action. The best and efficient IDS is very important to protect the data related to our business. By this efficient IDS system, organizations' clouds can be monitored for suspicious events. The combination of NIDS and HIDS gives the best results to identify the suspicious behavior. This can be improved by adding optimization algorithms. There are many optimization algorithms some of them are whale optimization algorithm, ant colony optimiza- tion algorithm, Be optimization algorithm, firefly optimization algorithm, dragonfly optimization algorithm etc. Conventionally many techniques are used to protect systems from malicious programs. Some of them are firewalls, anti-virus, signature-based malware detection, genetic network programming.Optimization techniques can bring-out best results. So, among many optimization techniques, the dragonfly optimization algorithm can be used to bring-out the efficient intrusion detection algorithm.

2.1 Summary

In this chapter we discussed the various researches conducted in order to achieve a clear view of Intrusion Detection

Chapter 3

Problem Definition

Intrusion Detection and Prevention System (IDS) plays a vital role in en- suring the security of modern computer installations. These systems are necessary in order to detect hostile activity and to respond appropriately.

3.1 Goals and Objective

The Objective of the system is to develop an enhanced application for de- tection of network intrusion in early stages using SVM classifier which gives the network administrator plenty of time to take precautionary measures in early stages.

3.2 Statement of Scope

- It is difficult or almost impossible to develop an intrusion detection system with 100 percent success rate.
- The next version of the proposed system may include a mechanism where the system would take network traffic logs from a network monitoring ool like for eg. WireShark Packet Analysis tool and provide robust and accurate detection

Chapter 4 Analysis

4.1 Project Estimates

4.1.1 Effort Estimate Table:

Task	Effort weeks	Deliverables	Milestones
Analysis of existing systems & compare with proposed one	4weeks		
Literature survey	1weeks		
Designing & planning	2weeks		
System flow	1weeks		
Designing modules & it ś de-liverables	2week	Modules: design document	
Implementation	7weeks	Primary system	
Testing	4weeks	Test Reports	Formal
Documentation	2weeks	Complete project port re-	Formal

Table 4.1: Effort Estimate Table

4.1.2 Project Description:

Stage and Phase	Task	Description
Stage 1 - Phase 1	Analysis	Analyse the information given in the IEEE paper.
Stage 1 - Phase 2	Literature survey	Collect raw data and elaborate on literature surveys.
Stage 1 - Phase 3	Design	Assign the module and design the process flow control.
Stage 2 - Phase 1	Implementation	Implement the code for all the modules and integrate all the modules.
Stage 2 - Phase 2	Testing	Test the code and overall process weather the process works properly.
Stage 2 - Phase 3	Documentation	Prepare the document for this project with conclusion and future enhancement.

Table 4.2: Project Scheduling

4.1.3 Estimation of KLOC:

The number of lines required for implementation of various modules can be estimated as follows:

Sr.No.	Modules	KLOC
1	Graphical User Interface	0.20
2	Back-end Algorithm Implementation	1.2
3	Front-Side Coding	1.2
4	Back-end Connectivity	0.6

Thus the total number of lines required is approximately 2.60 KLOC. D = (Total KLOC / KLOC in a day)/30

= (3.6/0.025)/30

= 4.8

4.2 Risk Management

4.2.1 Overview of Risk Mitigation, Monitoring, Management Risk management organizational role

Each member of the organization will undertake risk management. The development team will consistently be monitoring their progress and project status as to identify present and future risks as quickly and accurately as possible. With this said, the mem- bers who are not directly involved with the implementation of the product will also need to keep their eyes open for any possible risks that the development team did not spot. The responsibility of risk management falls on each member of the organization, while William Lord maintains this document.

Business Impact Risk

- Amount and quality of documentation that must be produced and delivered to customer the customer will be supplied with a complete online help file and users manual for Game Forge. Coincidentally, the customer will have access to all de- velopment documents for Game Forge, as the customer will also be grading the project.
- Governmental constraints in the construction of the product none known.
- Costs associated with late delivery Late delivery will prevent the customer from issuing a letter of acceptance for the product, which will result in an incomplete grade for the course for all members of the organization.
- Costs associated with a defective product Unknown at this time.

Customer Related Risks

- Have you worked with the customer in the past? Yes, All team members have completed at least one project for the customer, though none of them have been to the magnitude of the current project.
- Does the customer have a solid idea of what is required? Yes, the customer has access to both the System Requirements Specification, and the Software Require- ments Specification.

- Will the customer agree to spend time in formal requirements gathering meetings to identify project scope? Unknown. While the customer will likely participate if asked, the inquiry has not yet been made.

Process Risks

- Does senior management support a written policy statement that emphasizes the importance of a standard process for software development? N/A. PA Software does not have a senior management. It should be noted that the structured method has been adopted. At the completion of the project, it will be determined if the software method is acceptable as a standard process, or if changes need to be implemented.

- Has your organization developed a written description of the software process to be used on this project? Yes.

- Are staff members willing to use the software process? Yes. The software process was agreed upon before development work began.

- Is the software process used for other products? N/A. PA Software has no other projects currently.

Technical Issues

- Are facilitated application specification techniques used to aid in communication between the customer and the developer? The development team will hold frequent meetings directly with the customer. No formal meetings are held (all informal). During these meetings the software is discussed and notes are taken for future review.

- Are specific methods used for software analysis? Special methods will be used to analyze the software progress and quality. These are a series of tests and reviews to ensure the software is up to speed. For more information, see the Software Quality Assurance and Software Configuration Management documents.

- Do you use a specific method for data and architectural design? Data and architectural design will be mostly object oriented. This allows for a higher degree data encapsulation and modularity of code.

Technology Risk

- Is the technology to be built new to your organization? No
- Does the software interface with new or unproven hardware? No
- Is a specialized user interface demanded by the product requirements? Yes.

Development Environment Risks

Is a software project management tool available? No. No software tools are to be used. Due to the existing deadline, the development team felt it would be more productive to begin implementing the project than trying to learn new software tools. After the completion of the project software tools may be implemented for future projects.

4.3 Project Schedule

4.3.1 Project task

- Getting Basic Knowledge of python
- Going through the previous existing system
- Looking for required libraries in python
- Collecting Dataset
- Training Model using ML algorithm
- Building GUI for better outlook
- Dividing the assign task among group members.

4.3.2 Time Line Chart

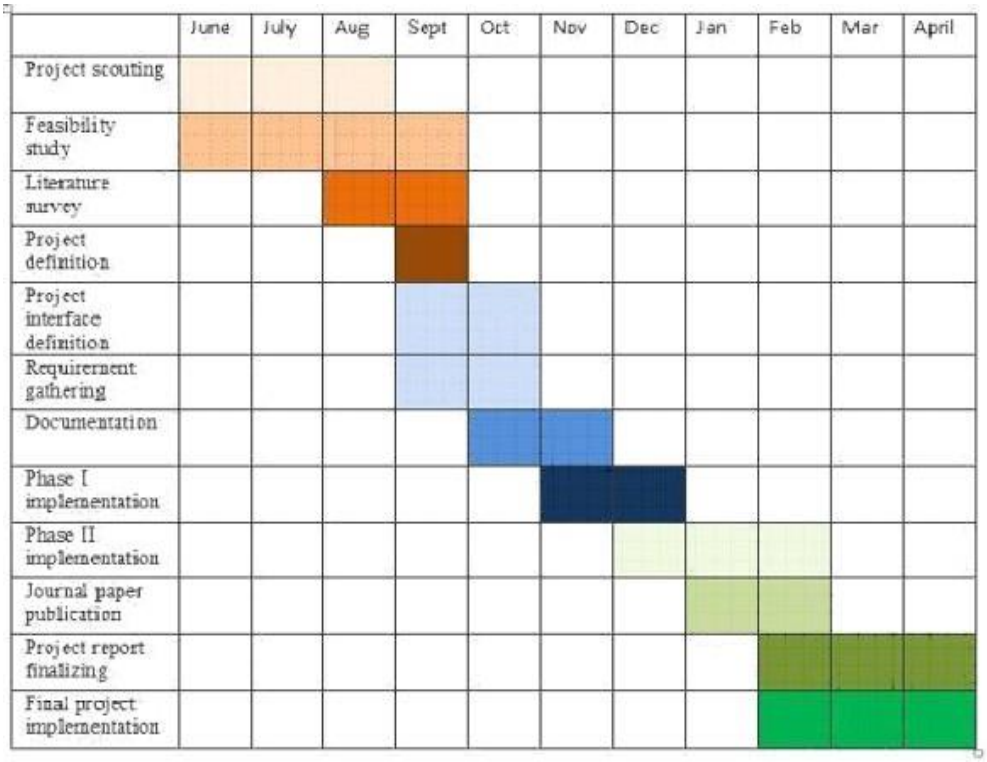


Figure 4.1: Time Line Chart

Chapter 5

Software Resource Specification

The project contains four modules: Data Preprocessing, Feature Selection, Train- ing the models and Attack prediction and testing. The dataset selected is NSL-KDD. It is the refined version of the KDD Cup 99 dataset. The KDD Cup 99 dataset is one of the most widely used datasets for training Intrusion Detection Systems(IDS) and Intru- sion Prevention Systems(IPS). There is a lack of labelled datasets for network security. This is because it is difficult to predict new types of attacks and know their signature. Some of the other popular datasets are DARPA, CAIDA, LBNL, CDX, Kyoto, UMASS, ISCX2012 and ADFA. The KDD Cup 99 Dataset has a lot of redundant values and instances. The NSL-KDD is created after removing the redundant values. It has 41 attributes and is classified into 4 types of attacks along with the normal network packet values. The next stage is Data Pre-processing. All the values in the dataset have to be in a numerical format for the classifiers to take in as input. Some of the features are categorical and have string values. This is then converted into numerical values using Label Encoder. A One-hot Encoder is then used to split the columns to each different category.

5.1 Functional Requirements

- Supports Complex Environments.
- Detects in Several Modes.
- Identifies All Attacks.
- Alerts on Anomalous Behaviour

- Provides Unified Data.
- Maintains Compliance.
- A Modern Intrusion Detection Platform Helps Reduce Risk Over Time.

5.2 Project Plan

5.2.1 Project Plan for semester I

The following Table 5.1 describes the project plan for semester I. It describes the various activities and accountability of the developers for the respective modules. Following are the major activities carried out in this plan :

- Identifying the functional requirements.
- Implementing Dataset.
- Studying the necessary Rules for Prediction.

Phase	Activity	Start Date	End Date	Group Members
1	Selection of Project Topic	22-09-2022	28-09-2022	Team
2	Study literature survey in detail	15-09-2022	30-09-2022	Team
3	Functional Requirement Specification(FRS)	21-09-2022	28-09-2022	Team
4	UML Diagram Prototype	29-09-2022	03-10-2022	Team
5	Model Selection	04-10–2022	11-10-2022	Team
6	Implementing Dataset	07-11-2022	13-11-2022	Team
7	Split Dataset for Testing and Training	07-11-2022	13-11-2022	Team
8	Software Requirement Specification	12-11-2022	17-11-2022	Team
9	Test Plan	18-11-2022	22-11-2022	Team

Table 5.1: Planner and Progress Report I for IDS

5.2.2 project plan for semester II

The following Table 5.2 describes the project plan for semester II. It describes the various activities and accountability of the developers for the respective modules. Following are the major activities carried out in this plan :

- Define Training and Testing.
- Implementing highest accuracy algorithm.
- Development of project in 3 Milestones.
- Studying the necessary and Proper Information.

Phase	Activity	Start Date	End Date	Group Members
1	Training Algorithms	09-04-2023	11-04-2023	Team
2	Development of Milestone No.1	16-04-2023	19-04-2023	Team
3	Searching for Changes in System	20-04-2023	23-04-2023	Team
4	Development of Milestone No.2	26-04-2023	29-04-2023	Team
5	Found Different Prediction	05-05–2023	09-05-2023	Team
6	Development of Milestone No.3	11-05-2023	15-05-2023	Team
7	Making Final Report	04-05-2023	29-05-2023	Team
8	Paper Presentation	29-05-2023	03-06-2023	Team

Table 5.2: Planner and Progress Report II for IDS

5.3 Summary

In this chapter we described the implementation details of the project plan for Semester I and Semester II. We also studied the necessary functions and the desirable functions of our system.

Chapter 6 Design

6.1 Software Requirement Specification

This software requirement specification (SRS) report expresses complete description about proposed System. This document includes all the functions and specifications with their explanations to solve related problems.

6.1.1 Problem Statement

In recent years there is a lot of public shaming in online social networks and related online public forums. Social media's accessibility has given people around the world a mouthpiece to raise and exchange ideas on meaningful issues. Platforms like Twitter and Facebook have expanded our knowledge of society, allowing people to be genuinely curious about each other. More and more, users are encouraged to share damning accusations online, often with little to no context.

6.1.2 User Classes and Characteristics

Basic knowledge of using computers is adequate to use this application. Knowledge of how to use a mouse or keyboard and internet browser is necessary. The user interface will be friendly enough to guide the user.

6.1.3 Assumptions and Dependencies

- Assumptions:

1. All the software such as python,etc are installed and running on the computers.
2. The cluster of nodes is formed and running.

- Dependencies:

1. It is assumed that user know his/her tasks in organizations.
2. All parameters are as per the dataset.
3. Well Trained dataset.

6.2 Functional Requirement

6.2.1 System Feature 1(Functional Requirement)

Functional requirement describes features, functioning, and usage of a product/system or software from the perspective of the product and its user. Functional requirements are also called as functional specifications were synonym for specification is design. Provide User friendly Interface and Interactive as per standards.

6.3 Non-Functional Requirement

6.3.1 Performance Requirements

- High Speed :- System should process requested task in parallel for various action to give quick response. Then system must wait for process completion.
- Accuracy :- System should correctly execute process, display the result accurately. System output should be in user required format.

6.3.2 Safety Requirements:

The data safety must be ensured by arranging for a secure and reliable transmission media. The source and destination information must be entered correctly to avoid any misuse or malfunctioning. Password generated by user is consisting of characters, special character and number so that password is difficult to hack. So, that user account is safe.

6.3.3 Security Requirements

Secure access of confidential data (user's details). Information security means protecting information and information systems from unauthorized access, use, disclosure, disrup- tion, modification or destruction.The terms information security, computer security and information assurance are frequently incorrectly used interchangeably. These fields are interrelated often and share the common goals of protecting the confidentiality, integrity and availability of information; however, there are some subtle differences between them.

6.3.4 Software Quality Attributes

1. Runtime System Qualities: Runtime System Qualities can be measured as the system executes.

2. Functionality: The ability of the system to do the work for which it was intended.

3. Performance:The response time, utilization, and throughput behavior of the system. Not to be confused with human performance or system delivery time.

4. Security:A measure of systems ability to resist unauthorized attempts at usage or behavior modification, while still providing service to legitimate users.

5. Availability: (Reliability quality attributes falls under this category) the measure of time that the system is up and running correctly; the length of time between failures and the length of time needed to resume operation after a failure.

6. Usability: The ease of use and of training the end users of the system. Sub qualities: learn ability, efficiency, affect, helpfulness, control.

7. Interoperability: The ability of two or more systems to cooperate at runtime.

6.4 System Requirement

6.4.1 Software Requirements(Platform Choice)

- Tools - Python IDE
- Programming Language - Python
- Software Version - Python 3.5

6.4.2 Hardware Requirements

- Processor - Pentium IV/Intel I3 core
- Speed - 1.1 GHz
- RAM - 512 MB (min)
- Hard Disk - 20GB
- Keyboard - Standard Keyboard
- Mouse - Two or Three Button Mouse
- Monitor - LED Monitor

6.5 Analysis Models: (SDLC model to be applied)

One of the basic notions of the software development process is SDLC models which stands for Software Development Life Cycle models. SDLC is a continuous process, which starts from the moment, when its made a decision to launch the project, and it ends at the moment of its full remove from the exploitation. There is no one single SDLC model. They are divided into main groups, each with its features and weaknesses. Evolv- ing from the first and oldest waterfall SDLC model, their variety significantly expanded.

The SDLC models diversity is predetermined by the wide number of product types starting with a web application development to a complex medical software. And if you take one of the SDLC models mentioned below as the basis in any case, it should be adjusted to the features of the product, project, and company. The most used, popular and important SDLC models are given below:

1. Waterfall Model
2. Iterative Model
3. Spiral Model
4. V-shaped Model
5. Agile Model

Waterfall Model

Waterfall is a cascade SDLC model, in which development process looks like the flow, moving step by step through the phases of analysis, projecting, realization, testing, im- plementation, and support. This SDLC model includes gradual execution of every stage completely. This process is strictly documented and predefined with features expected to every phase of this software development life cycle model.

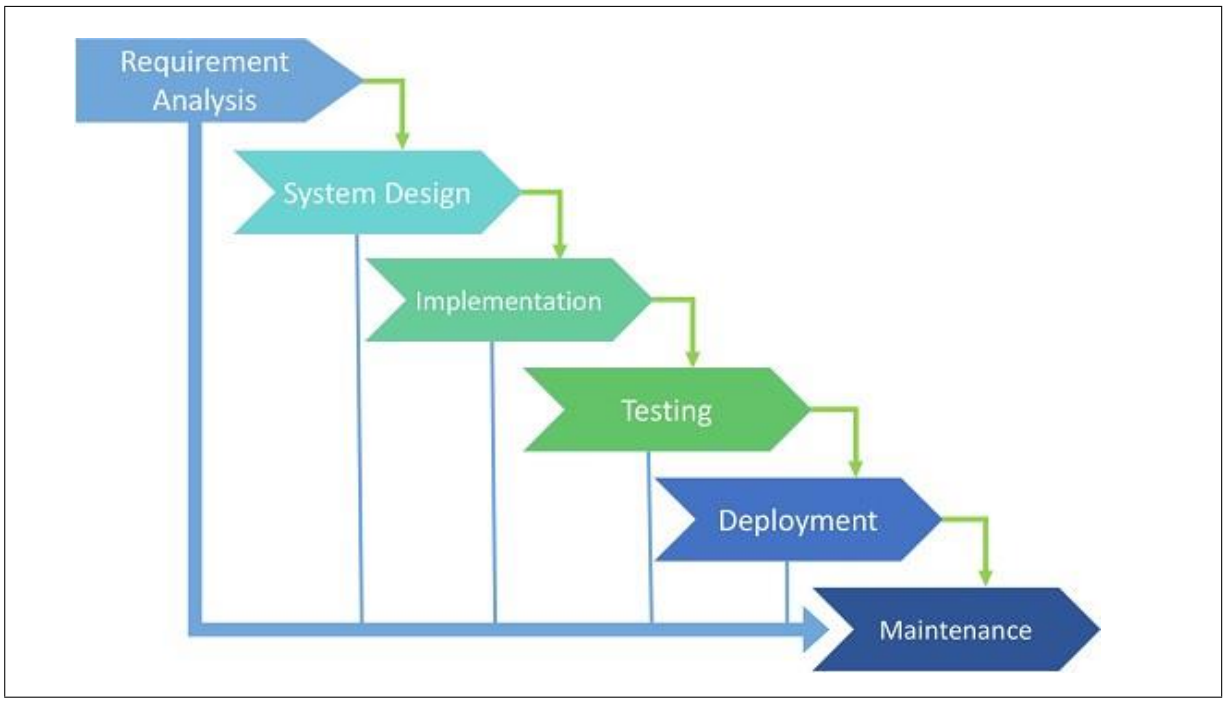


Figure 6.1: Waterfall Model

1. Planning and requirement analysis

Each software development life cycle model starts with the analysis, in which the stakeholders of the process discuss the requirements for the final product. The goal of this stage is the detailed definition of the system requirements. Besides, it is needed to make sure that all the process participants have clearly understood the tasks and how every requirement is going to be implemented. Often, the discussion involves the QA specialists who can interfere the process with additions even during the development stage if it is necessary.

2. Designing project architecture

At the second phase of the software development life cycle, the developers are ac- tually designing the architecture. All the different technical questions that may appear on this stage are discussed by all the stakeholders, including the customer. Also, here are defined the technologies used in the project, team load, limitations, time frames, and budget. The most appropriate project decisions are made accord- ing to the defined requirements.

3. Development and programming

After the requirements approved, the process goes to the next stage actual develop- ment. Programmers start here with the source code writing while keeping in mind previously defined requirements. The

system administrators adjust the software environment, front-end programmers develop the user interface of the program and the logics for its interaction with the server. The programming by itself assumes four stages:-

- Algorithm development
- Source code writing
- Compilation
- Testing and debugging

4. Testing

The testing phase includes the debugging process. All the code flaws missed during the development are detected here, documented, and passed back to the developers to fix. The testing process repeats until all the critical issues are removed and software work ow is stable.

5. Deployment

When the program is finalized and has no critical issues it is time to launch it for the end users. After the new program version release, the tech support team joins. This department provides user feedback; consult and support users during the time of exploitation. Moreover, the update of selected components is included in this phase, to make sure, that the software is up-to-date and is invulnerable to a security breach.

Chapter 7 Modeling

This chapter includes the various modeling techniques which describes the various users of the web application It also describes the functionality of the different features of the web application.

7.1 Data Flow Diagrams

A data flow diagram (DFD) is a graphical or visual representation using a stan- dardized set of symbols and notations to describe a business's operations through data movement. They are often elements of a formal methodology such as Structured Systems Analysis and Design Methods.
The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts

as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.

DFD 0, also called context diagram of the result management system. As the bubbles are decomposed into less and less abstract bubbles, the corresponding data flow may also be needed to be decomposed.

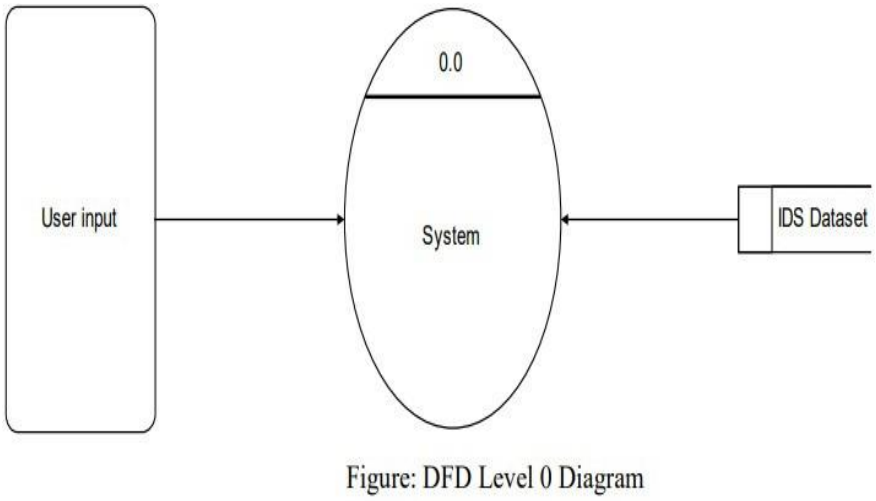


Figure: DFD Level 0 Diagram

Figure 7.1: DFD 0 Diagram

DFD 1, a context diagram is decomposed into multiple bubbles/processes. In this level, we highlight the main objectives of the system and breakdown the high-level process of 0-level DFD into subprocesses.

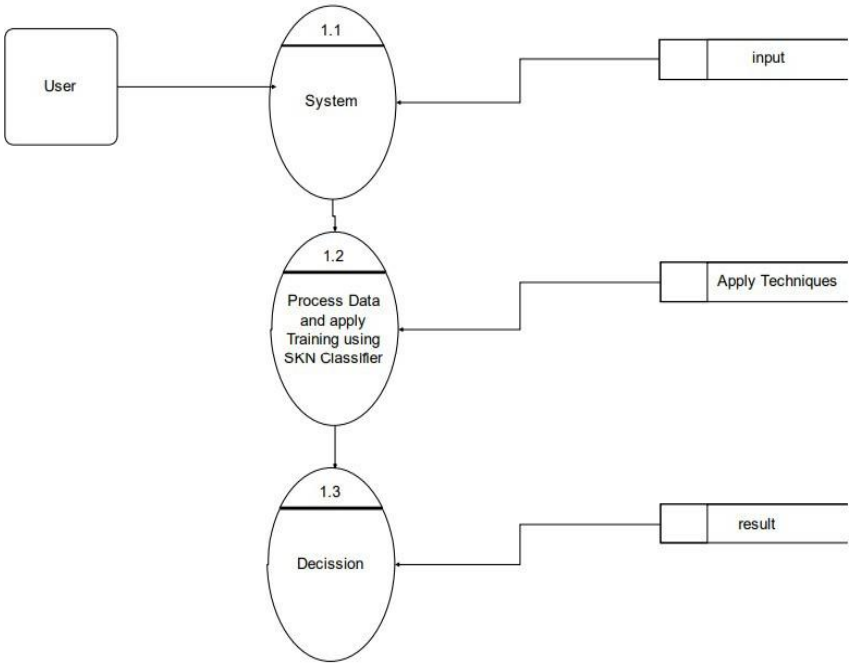


Figure: DFD Level 1 Diagram

Figure 7.2: DFD 1 Diagram

7.2 UML Diagram

7.2.1 Activity Diagram

Use cases show that your system should do. Activity diagrams allow you to spec- ify how your system will accomplish its goals. Activity diagrams show high-level actions chained together to represent a process occurring in your system. An activity diagram is essentially a flowchart, showing flow of control from activity to activity. Unlike a tra- ditional flowchart, an activity diagram shows concurrency as well as branches of control. Activity diagrams focus on the dynamic flow of a system.

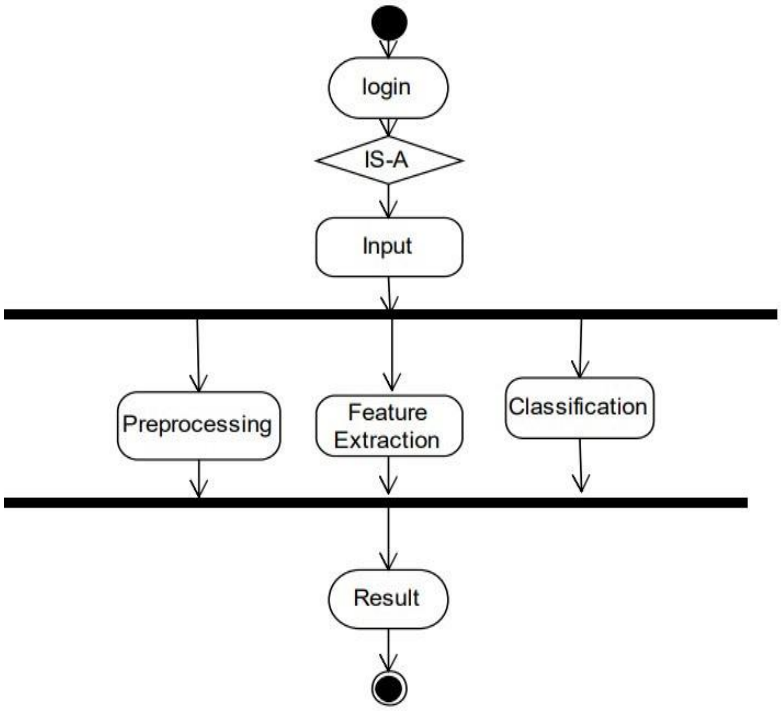


Figure 7.3: Activity Diagram

7.2.2 Sequence Diagram

The sequence diagram is used primarily to show the interactions between objects in the sequential order that those interactions occur. Developers typically think sequence diagrams were meant exclusively for them. However, an organization's business staff can find sequence diagrams useful to communicate how the business currently works by showing how various business objects interact.Sequence diagrams illustrate how objects interact with each other. They focus on message sequences, that is, how messages are sent and received between a number of objects. The main purpose of sequence diagram is to show the order of events between the parts of system that are involved in particular interaction.

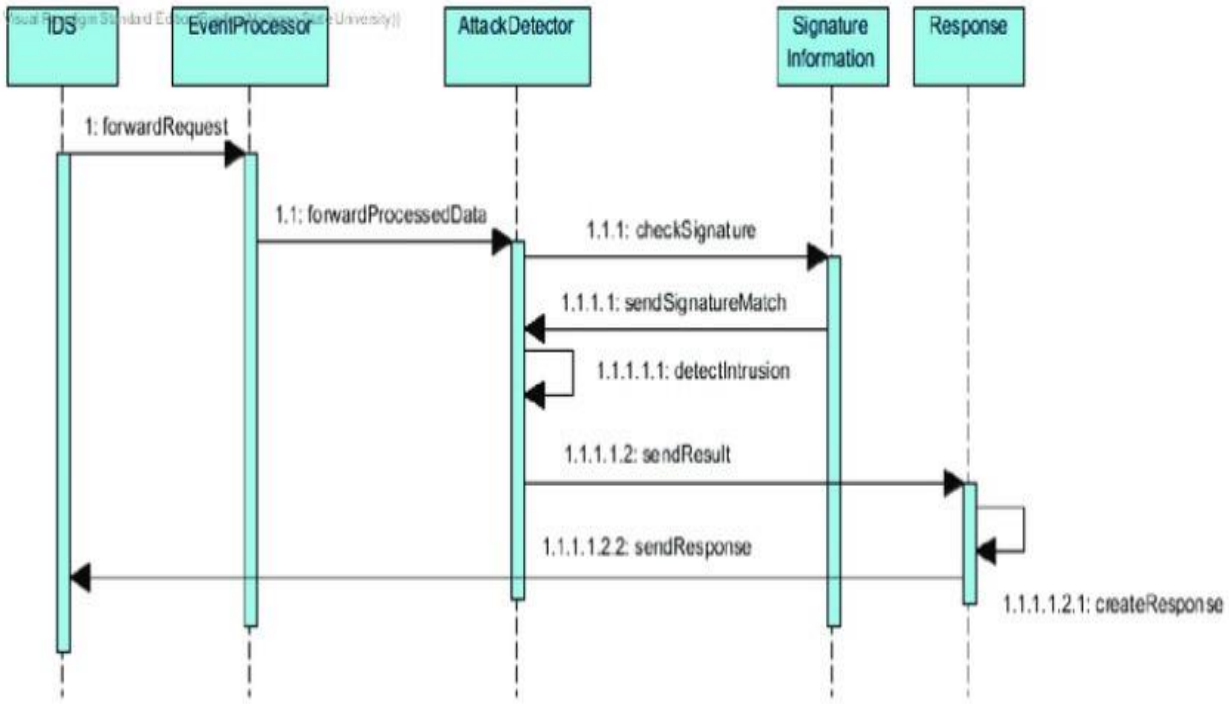


Figure: Sequence Daigram

Figure 7.4: Sequence Diagram

7.2.3 Use Case Diagram

Four modeling elements make up the use case diagram; these are:

- **Actors:** Actors refer to a type of users, users are people who use the system. In this case student, teacher developer are the users of the framework and application

- **Use cases:** A use case defines behavioral features of a system. Each use case is named using a verb phrase that express a goal of the system. The name may appear inside or outside the ellipse.

- **Associations:** An association is a relationship between an actor and a use case. The relationship is represented by a line between an actor and a use case.

- **The include relationship:** It is analogous to a call between objects. One use case requires some type of behavior which is fully defined in another use case.

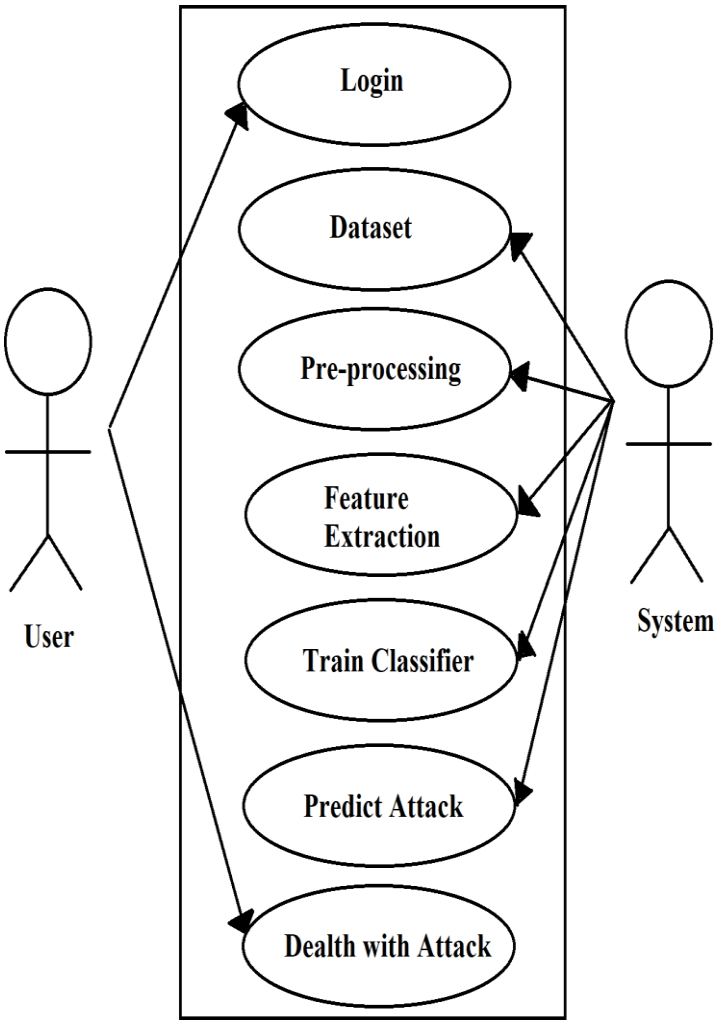


Figure 7.5: usecase diagram

7.2.4 Component Diagram

Component diagram are one of the two kinds of diagrams found in modeling the physical aspects of object oriented systems. A component diagram shows organization and dependencies among set of components. Component diagram can be seen to model the static implementation view of a system. This involves modeling the physical things that resides on a node, such as executables, libraries, tables, files and documents.

Component diagram shows a set of components and their relationships. Graph- ically a component diagram is a collection of vertices and arcs. Component diagrams commonly contain,

- **Components**
- **Interfaces**
- **Dependency, generalization, association and realization relationships.**

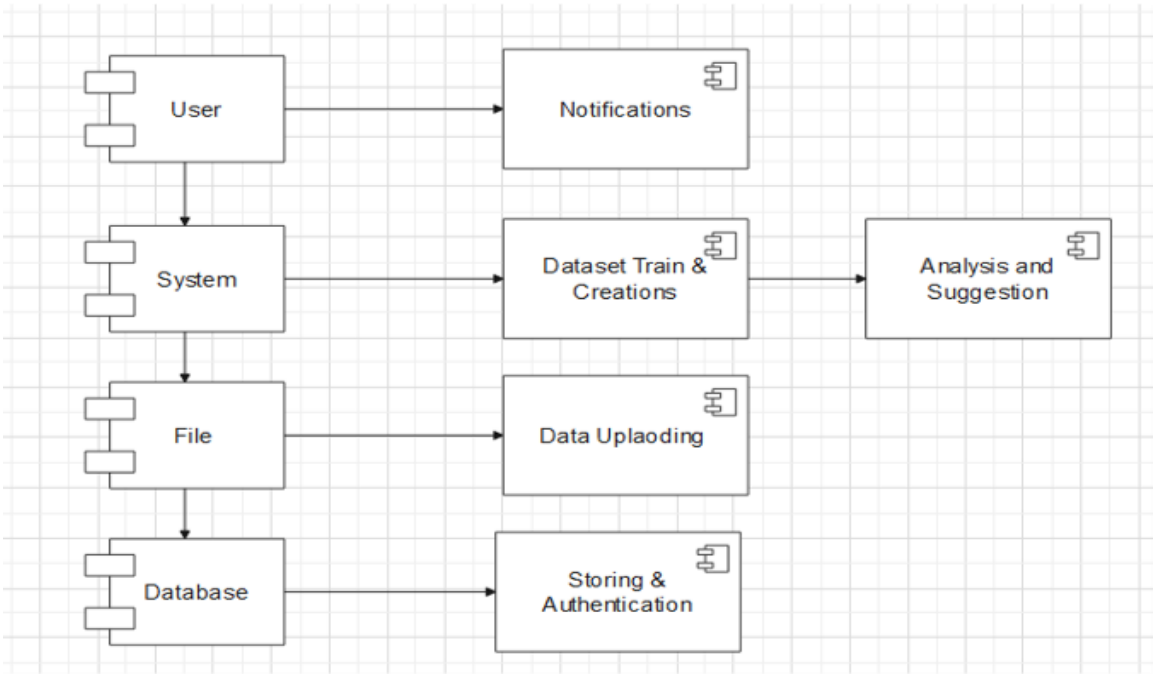


Figure 7.6: Component diagram

7.3 Summary

Thus we saw the various modeling techniques used for the design of web Applications

Chapter 8 System Design

8.1 Methodology and Models

Our overall methodology is divided into four sections: Data Preparation, Model training, Result visualization, and Fine-tuning Data pre-processing: loading a dataset in memory and processing it to gather image-label pairs and CSV files for the classification task.
Intrusion detection and prevention systems use different methodologies such as signature based, anomaly based, stateful protocol analysis, and a hybrid system that combines some or all of the other systems to detect and respond to security threats.
The growth of systems that use a combination of methods creates some confusion when trying to choose a methodology and system to deploy

Type of attack detection :-

1. Signature (Rule based) based IDS

(a) Pattern

(b) Database of attack pattern

(c) Detect Known attack

(d) Cannot identify new attack

2. Anomaly Based

(a) Deviation

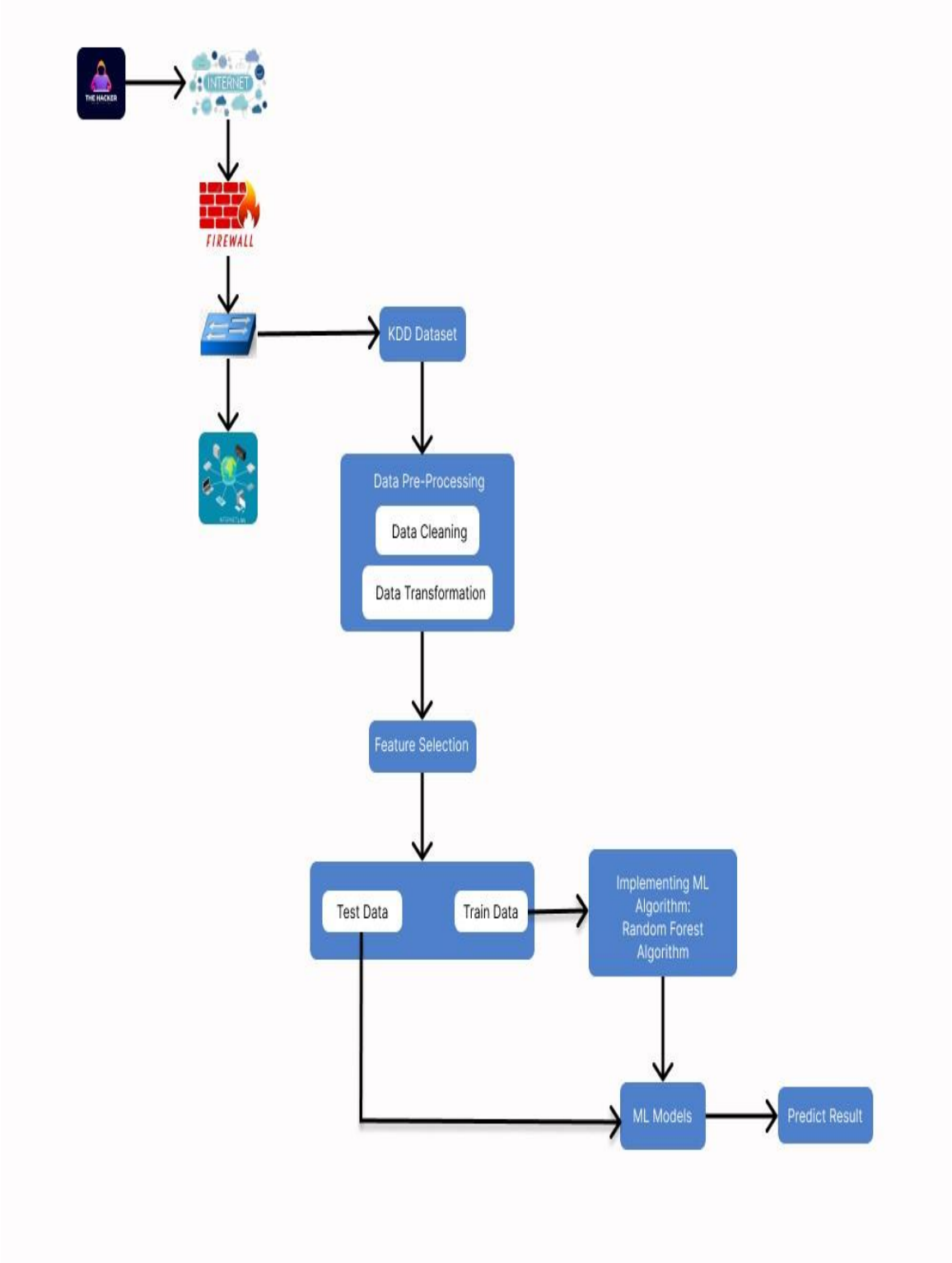


Figure 8.1: System Architecture

8.2 Algorithm

RFC ALGORITHM STEPS:

Step 1 : Load the dataset.

Step 2 : Apply pre-processing technique Discretization. Step 3 : Cluster the dataset into four datasets.

Step 4 : Partition the data set into training and test.

Step 5 : Select the best set features using feature subset selection measure Symmetrical uncertainty (SU) compensates information gain.

Step 6 : Data set is given to Random forest for training.

Step 7 : The test data set is then fed to random forest for classification. Step 8 : Calculate accuracy, Detection rate.

8.3 Python Programming

Python is a cross-platform programming language, which means that it can run on multiple platforms like Windows, macOS, Linux, and has even been ported to the Java and .NET virtual machines. It is free and open-source. Even though most of today's Linux and Mac have Python pre-installed in it, the version might be out-of-date. So, it is always a good idea to install the most current version. The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.

Why Python?

- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
- Python has a simple syntax similar to the English language.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.

- Python can be treated in a procedural way, an object-oriented way or a functional way. Python was designed for readability, and has some similarities to the English language with influence from mathematics.

- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.

- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly brackets for this purpose.

8.3.1 My SQL

MySQL is a relational database management system (RDBMS) developed by Or- acle that is based on structured query language (SQL). A database is a structured col- lection of data. It may be anything from a simple shopping list to a picture gallery or a place to hold the vast amounts of information in a corporate network.MySQL has all essential SQL numeric data types. These data types can include the exact numeric data types (For example, integer, decimal, numeric, etc.), as well as the approximate numeric data types (For example, float, real, and double precision). It also supports BIT datatype to store bit values

8.3.2 SVM (Support Vector Machine)

- We present an attribute-based signature scheme with multiple authorities, in which a patient endorses a message according to the attribute while disclose.

- SVM is a learning method for the Classification and Regression analysis of both linear and nonlinear data. It uses a hypothesis space of linear functions and maps input feature vectors into a higher dimensional space all the way through some nonlinear mapping.

- SVM constructs a hyper plane or set of hyper planes only the good separation is achieved by the hyper plane. The hyper plane searching process in SVM is achieved by the leading margin

- The related margin gives the major separation between classes. While training an SVM it creates a quadratic optimization problem

- SVM uses a function called kernel to solve this problem. ing no information other than the evidence that he has attested to it.

8.3.3 KNN (K-Nearest Neighbour)

- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.

- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.

- KNN is implemented with different K values. KNN is executed by changing the number of testing records. K Value is also decided by making keen observations. KNN algorithm gives on an average 92.3% accuracy. Naive Bayes algorithm is also executed by changing the number of testing records. Naive Bayes algorithm gives on average 70.66% accuracy. Time complexity of NB algorithm is less than KNN. Comparison of both the methods is presented by comparing their evaluation metrics like Accuracy, Precision, Recall, Specificity and F- Measure. This paper is concluded by identifying the pros and cons of both the algorithms and by providing the future scope of this paper.

8.3.4 RANDOM FOREST ALGORITHM

- Random Forest proposes new systematic frameworks that apply a data mining algorithm called random forests in misuse, anomaly, and hybrid detection.

- The random forests algorithm is an ensemble classification and regression approach, which is one of the most effective data mining techniques.

- The random forests algorithm has been used extensively in different applications. For instance, it has been applied to prediction, and probability estimation. How- ever, the algorithm has not been applied in automatic intrusion detection.

- In our proposed system, the misuse component uses the random forests algorithm for the classification in intrusion detection, while the anomaly component is based on the outlier detection mechanism of the algorithm.

- Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction.

8.4 KDD Dataset

The KDD dataset, also known as the KDD Cup dataset, refers to a large dataset used in the field of data mining and machine learning. KDD stands for ”Knowledge Dis- covery in Databases.” It was created for the purpose of the Third International Knowledge Discovery and Data Mining Tools Competition, held in 1999. The KDD dataset repre- sents network traffic data collected from various sources, including simulated attacks, as well as normal network activity. The dataset was designed to encourage research and development in the area of intrusion detection systems, which aim to detect and prevent unauthorized access and malicious activities in computer networks.

Uses of the KDD dataset

- Intrusion detection system (IDS) development: The KDD dataset is widely used for developing and evaluating intrusion detection systems. Researchers can train IDS models on the dataset to identify various types of attacks and abnormal network behavior.

- Machine learning research: The KDD dataset serves as a valuable resource for researchers working on machine learning and data mining algorithms. It allows them to explore and experiment with different techniques to improve the accuracy and efficiency of intrusion detection systems.

- Security algorithm evaluation: The dataset enables researchers to evaluate the per- formance of different algorithms and techniques for network security and intrusion detection. By comparing the results of different approaches on the KDD dataset, researchers can determine the effectiveness of various methods.

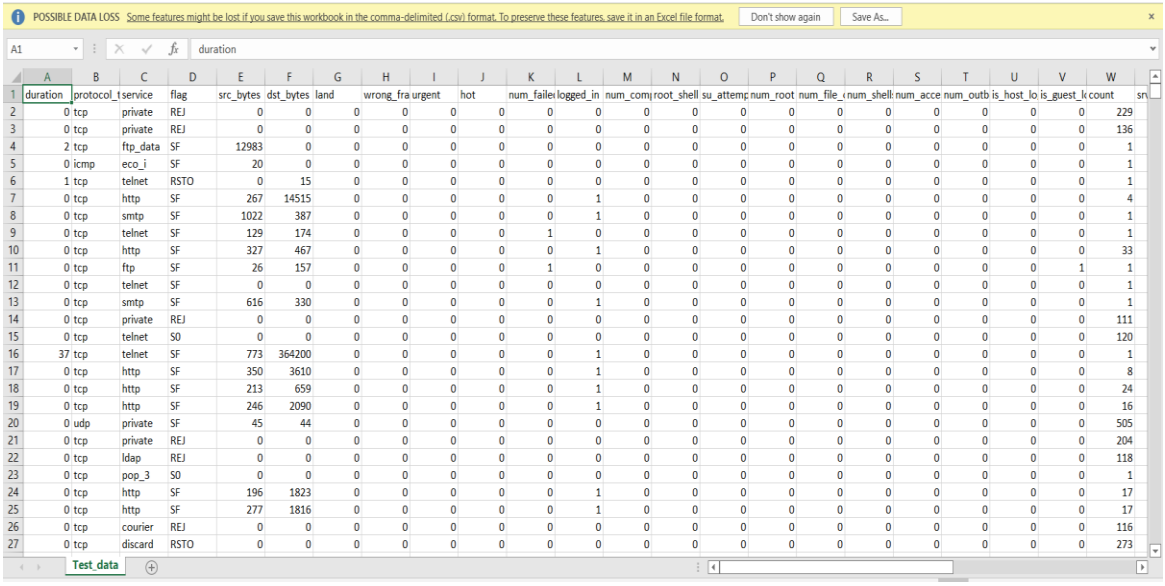
Benefits of the KDD dataset

- Realistic representation: The KDD dataset provides a realistic representation of network traffic and includes both normal and abnormal activities, including simu- lated attacks. This makes it valuable for developing and testing intrusion detection and network security systems

• Standard benchmark: The KDD dataset has become a standard benchmark in the field of intrusion detection and network security. It enables researchers and practi- tioners to compare and evaluate the performance of different algorithms, techniques, and systems on a common dataset.

• Training and evaluation: The dataset can be used for training and evaluating ma- chine learning models and algorithms. By utilizing the KDD dataset, researchers can develop and improve intrusion detection systems, identify patterns and anoma- lies in network traffic, and enhance network security.

8.5 Testing Dataset



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	duration	protocol_t	service	flag	src_bytes	dst_bytes	land	wrong_fra	urgent	hot	num_faile	logged_in	num_com	root_shell	su_attemp	num_root	num_file_	num_shell	num_acce	num_outb	is_host_lo	is_guest_l	count
2	0	tcp	private	REJ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	229
3	0	tcp	private	REJ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	136
4	2	tcp	ftp_data	SF	12983	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
5	0	icmp	eco_i	SF	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
6	1	tcp	telnet	RSTO	0	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
7	0	tcp	http	SF	267	14515	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	4
8	0	tcp	smtp	SF	1022	387	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
9	0	tcp	telnet	SF	129	174	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1
10	0	tcp	http	SF	327	467	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	33
11	0	tcp	ftp	SF	26	157	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1
12	0	tcp	telnet	SF	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
13	0	tcp	smtp	SF	616	330	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
14	0	tcp	private	REJ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	111
15	0	tcp	telnet	S0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	120
16	37	tcp	telnet	SF	773	364200	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
17	0	tcp	http	SF	350	3610	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	8
18	0	tcp	http	SF	213	659	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	24
19	0	tcp	http	SF	246	2090	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	16
20	0	udp	private	SF	45	44	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	505
21	0	tcp	private	REJ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	204
22	0	tcp	ldap	REJ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	118
23	0	tcp	pop_3	S0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
24	0	tcp	http	SF	196	1823	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	17
25	0	tcp	http	SF	277	1816	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	17
26	0	tcp	courier	REJ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	116
27	0	tcp	discard	RSTO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	273

Figure 8.2: Test Dataset

8.6 Training Dataset

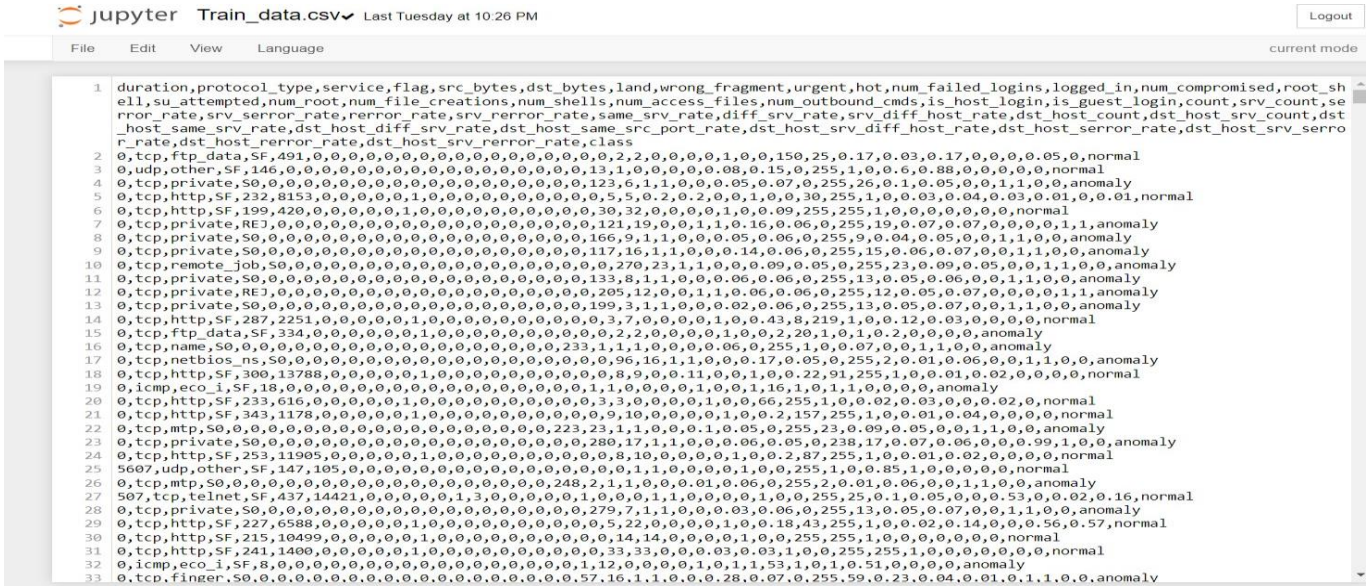


Figure 8.3: Training Dataset

8.7 Output

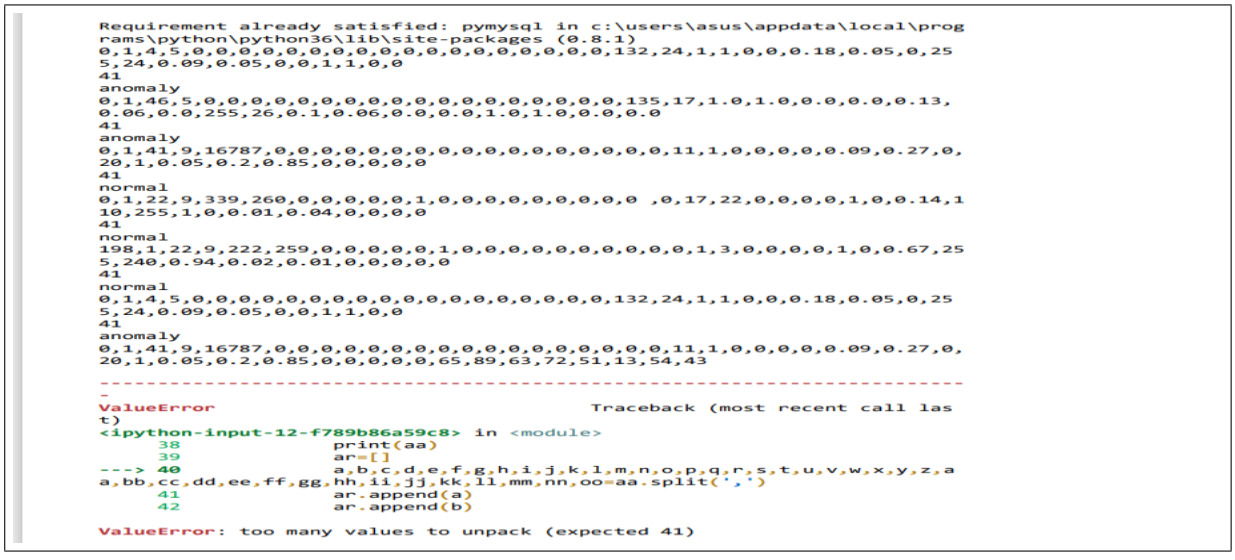


Figure 8.4: Result Error and Success

Figure 8.5: Login Page

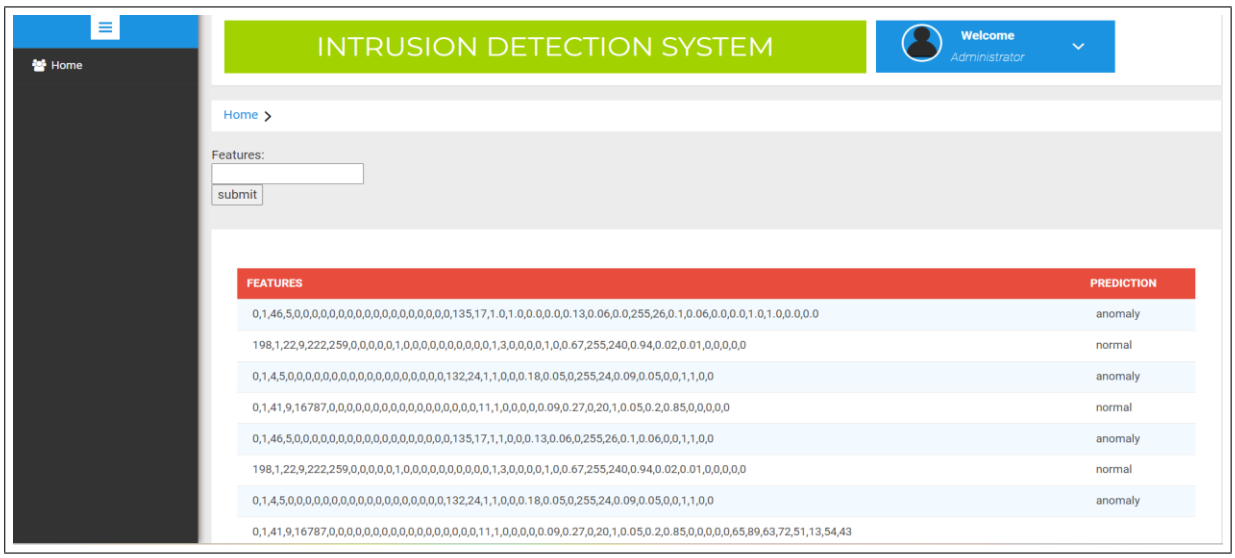


Figure 8.6: Admin Input Prediction Page

8.8 Hardware Requirements

- Windows 7 or above
- Ram 4 GB or More
- 500 GB or above Hard Disk

8.9 Software Requirements

- Python (Jupyter Notebook)
- HTML/CSS
- MySQL
- PHP

8.10 Summary

In this chapter we were made aware of the various advantages of the framework and also the limitations of the project. We also saw the hardware and software requirements of the project.

Chapter 9 TESTING

9.1 Test Plans

A system should always be tested thoroughly before implementing it, as regards its indi- vidual programs. This is because implementing a new system is a major job which a lot of man hours and a lot of other resources, so an error not detected before implementation may cost a lot. Effective testing early in the process translates directly into long term cost saving from reduced number of errors. This is also necessary because in some cases, a small error is not detected and corrected before installation, which may explode into much larger problem. Programming and testing is followed by the stage of installing the new computer based system. Actual implementation of the system can begin at this point using either a parallel or a direct changeover plan, or some blend of two. Testing and implementation of fire fighting robot controlled using android application is carried out as below. Software testing is a critical element of Software Quality Assurance(SQA) and represents the ultimate review of specification, design and coding. The purpose of product testing is to verify and validate the various work products viz. units, integrate unit, final product to ensure that they meet their respective requirements.

9.2 Testing Procedure

Software Testing is the critical element of the Software Quality Assurance and represents the ultimate review of specification, design and coding. Testing is the process of checking whether software works according to the specification. Testing will be performed by running the program using the test data. Testing is vital to the success of the system. It will also test whether the system identifies the problem correctly.

System is tested by following steps:

- Unit Testing: Each program is tested individually using dummy records to see whether that program produce satisfactory reports.
- Sequential Testing: The program, whose output will affect the processing done by another program, will be tested using dummy records.
- System Testing: The system is corrected in such a way that it does not affect the forced system failure. This testing is done with low volumes of data

9.3 Test Strategy

The test strategy consists of a series of different tests that will fully exercise the system. The primary purpose of the test is to uncover the system limitations. Following are the several tests that will be conducted:

9.3.1 Unit Testing:

Testing conducted to verify the implementation of the design for one software element (e.g., unit, module) is called unit testing. The purpose of unit testing is to ensure that the program logic is complete and correct and ensuring that the component works as designed. In this module, each unit will go through Unit testing after the completion of the module. The bugs in module testing will be reported in Test Log document and will be reported to the developers. After fixing the bug successfully, one more iteration of module testing (Regression Testing) is done. This process is repeated till all critical test cases pass.

9.3.2 Integration Testing:

Testing conducted in which software elements, hardware elements, or both are combined and tested until the entire system has been integrated. The purpose of integration testing is to ensure that design objectives are met and ensures that the software, as a complete entity, complies with operational requirements. This type of testing will be done after all module test cases are passed through module testing, security testing, performance testing, user interface testing and regression testing

9.3.3 Performance Testing:

In developing the system, we are going to use Java which will reduce the response time. In Performance Testing, We are going to test Response time for each Screen. It is a type of non-functional testing. Performance testing is testing that is performed; to determine how fast some aspect of a system performs under a particular workload. It can serve different purposes like it can demonstrate that the system meets performance criteria. It can compare two systems to find which performs better. Or it can measure what part of the system or workload causes the system to perform badly. This process can involve quantitative tests done in a lab, such as measuring the response time or the number of MIPS (millions of instructions per second) at which a system functions.

9.3.4 Regression Testing:

Testing done to ensure that, the changes to the application have not adversely affected previously tested functionality. Here testing will take care of the test cases passed during the first module testing will not be affected in the subsequent rounds of module testing.

9.4 Test Cases

The listed tests were conducted in the software at the various developments stages. Unit testing was conducted. The errors were debugged and regression testing was performed. The integration testing will be performed once the system is integrated with other related systems like Inventory, Budget etc. Once the design stage was over the Black Box and White Box Testing was performed on the entire application. The results were analyzed and the appropriate alterations were made. The test results proved to be positive and henceforth the application is feasible and test approved.

9.5 Test Result

Sr.No	Description	Test Case I/P	Expected	Actual Result	Test Criteria (P/F)
1	Install Python	Python Exe	Should get install properly	Proper Installed	PASS
2	Installing Libraries	Library command for install	Should Get installed	Libarary Installed Successfully	PASS
3	Training Dataset	Dataset Training	Error in Training Model	Trained Model	FAIL
4	Training Dataset	Dataset Training	Trained Model	Trained Model	PASS
5	Login Credentials	User Name and Password	Login Unsuccessful	Unsuccessful Login	FAIL
6	Login Credentials	User Name and Password	Login Successful	Successful Login	PASS
7	Password	Current and New Password	Password Updated	Update Password	PASS
8	Prediction	Text as input	Should Predict the result	Result Predicted and shows normal	PASS
9	New Prediction activity	Text as input	Should Predict different activity	Result not Predicted different activity	FAIL
10	New Prediction result	Text as input	Should Predict the result	Result Predicted	PASS

Figure 9.1: Test Case

Chapter 10

Technical Specifications

10.1 Advantages

- Highly Accurate in terms of security
- Achieve high level security
- Quick Response Time and Low Power Consumption
- Fully Automated System

10.2 Limitations

To participate in mobile learning one must have a tablet or mobile devices with android as its base operating system,these can have high ranges of cost, due to this reason it cannot be affordable by everybody in todays world.
Another aspect to be considered is the size of the device, this is only a challenge if one incorrectly plans mobile learning content to be nothing more than compressed e-Learning. If your users are already using their mobile device that you plan to push learning to, your strategy should be what content do they need in the context of using the device.

10.3 Applications

The M-Learning framework can be used in following areas:
Intrusion Detection and Prevention System Using Machine Learning

- Institutions for teaching the learning material developed by the developer for mobile learning.
- Students can study with ease.
- The education application developer can use this framework for developing number of applications that can be imported on mobile devices.

10.3.1 Hardware Requirements

- AMD/Intel Processor
- 2GB RAM for application development
- Min. 16 GB Hard Disk

10.3.2 Software Requirements

- HTML5
- PHP
- Python
- MySQL

10.4 Summary

In this chapter we were made aware of the various advantages of the framework and also the limitations of the project. We also saw the hardware and software requirements of the project.

Chapter 11 Conclusion

A New Machine Learning Based Data Classification system using Support Vector Machine algorithm has Been Established for The Intrusion Detection Problem. In Order to Achieve Superior Performance and To Enhance Accuracy Rate and Faster Running Time. The Performance of The Proposed IDS Framework Will Be Evaluated in Terms of Detection Rate, Precision, F1 Score, Recall, And False-positive Rate. The KDD CUP 1999 Dataset Will Be Used to Test the Proposed IDS Framework.

Bibliography

[1] *Network Intrusion Detection System Based On Machine Learning Algo- rithms , International Journal of Computer Science Information Tech- nology* ; Vipin Das, Vijaya Pathak , Sattvik Sharma, Sreevathsan, MVVNS.Srikanth, Kumar T Gireesh, 2010.

[2] *A hierarchical detection and response system to enhance security against lethal cyber-attacks in UAV networks. IEEE Transactions on Systems, Man, and Cybernetics: Systems*; Sedjelmaci H, Senouci SM, Ansari
N. 2018 Sep; 48(9):1594-606.

[3] *Comparative Performance Analysis of classification algorithm for Inter- nal Intrusion Detection*; Mohammed Anbar, Rosni Abdulah, Izan H. Hasbullah, Yung- Wey Chong; Omar E. Elejla, 2016 14th Annual Conference on Privacy Security and Trust (PCT), Dec 12-14,2016, Penang, Malaysia.

[4] *a novel unsupervised Anamoly detection Approach for Intrusion Detec- tion System*; Weiwei Chen, Fangang Kong, Feng Mei, GuiginYuan, Bo Li, 2017 IEEE 3rd International Conference on big data security on cloud, May 16-18,2017,
Zhejiang, China

[5] *An Advanced method for detection of botnet traffic using Interhnal In- trusion Detection*; Manoj s. Koli, Manik K. Chavan, 2017 International Conference on (ICICCT), March 10-11, 2017, Sangli, India

[6] *Building an efficient intrusion detection system based on feature selec- tion and*

ensemble classifier ;Yuyang Zhou, Guang Cheng, Shanqing Jiang, Mian Dai, 2019

[7] *Evaluation of Machine Learning Techniques for Network Intrusion De- tection*; Marzia z, Chuag-Horng L, 2018, IEEE (PP. 1-5)

[8] *Network Intrusion Detection using Supervised Machine Learning Tech- nique with feature selection. International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)*; Kazi A., Billal M. and Mahbubur R. (2019), (pp. 643-646). IEEE.

[9] *A Survey of Data Mining and Machine Learning methods for cybersecu- rity intrusion detection*; Anna L. Buczak, Erha n Guven, IEEE communication surveys and tutorials, vol. 18, Issue 2,2016.

[10] *Network Intrusion Detection using Clustering and Gradient Boosting. International Conference on Computing, Communication and Network- ing Technologies*; Verma P, Shadab K, Shayan A. and Sunil B. (2018) (pp. 1-7). IEEE.