Project Report

for

Intrusion Detection System Using Fuzzy Clustring Algorithm

Submitted By

Name of the Student	**Exam Seat No.**
Tapare Prashant Bharat	(B80784218)
Bhujbal Harishchandra Jalindar	(B80784243)
Walkunde Kiran Baburao	(B80784259)
Shinde Nandkumar Parshuram	(B80784278)

B.E. (COMPUTER)

Guided By

Mr.Danny J.Pereira

Department of Computer Engineering
Government College of Engineering and Research
Awasari(kd), Pune

2013-14

Acknowledgement

The satisfaction that accompanies that the successful completion of any task would be incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all efforts with success. We are grateful to our project guide Mr. Danny J. Pereira Sir for the guidance, inspiration and constructive suggestion that helpful us in the preparation of this project. I wish to extend my sincere gratitude to Mr. D.J. Pereira, HOD, Department of Computer Engineering for his valuable guidance and encouragement which has been absolutely helpful in successful completion of this project work.

Abstract

Nowadays Intrusion Detection System (IDS) which is increasingly a key element of system security is used to identify the malicious activities in a computer system andnetwork. There are different approaches being employed in intrusion detection systems, but unluckily each of the technique so far is not entirely ideal. The prediction process may produce false alarms in many anomaly based intrusion detection systems. To achieve that, this paper proposes IDS model based on Fuzzy Logic. Proposed model consists of three parts Client side model which include simple bank application, IDS model in which previously defined testing set and training set are defined with Fuzzy algorithm and Apriori algorithm and Admin model which are define some rule for user and show system result. Also IDS model contain Artificial Neural Network which is useful for self-intrusion detection system. This manually update database we discover self-detection and updating technique by using artificial neural network algorithm. Intrusion Detection System, can detect, prevent and react to the attacks. In our system when client attacks on server system our system detects that attack and blocks that client and that pattern of attack is stored at admin side. If another client attacks with same pattern then that client is detected and blocked. Admin performs Turing test for client by generating questions.

Contents

List of Figures

List of Tables

1 INTRODUCTION

1.1 Overview

An Intrusion detection system (IDS) is software and/or hardware designed to detect unwanted attempts at accessing, manipulating, and/or disabling of computer systems, mainly through a network, such as the Internet. Firewalls limits access between networks to prevent intrusion and do not signal an attack from inside the network. An IDS evaluates a suspected intrusion once it has taken place and signals an alarm. As the network of computers expands both in number of hosts connected and number of services provided, security has become a key issue for the technology developers. This work presents a prototype of an intrusion detection system for networks. There is often the need to update an installed Intrusion Detection System (IDS) due to new attack methods or upgraded computing environments. Since many current IDSs are constructed by manual encoding of expert knowledge, changes to IDSs are expensive and slow. To detect intrusions the process of learning the behavior of a given program by using machine-learning techniques.

1.2 Brief Description

With the enormous growth of computer networks usage and the huge increase in the number of applications running on top of it, network security is becoming increasingly more important. All the computer systems suffer from security vulnerabilities which are both technically difficult and economically costly to be solved by the manufacturers. Therefore, the role of Intrusion Detection Systems (IDSs), as special-purpose devices to detect anomalies and attacks in the network, is becoming more important. The research in the intrusion detection field has been mostly focused on anomaly-based a misuse-based detection techniques for a long time. While misuse-based detection is generally favored in commercial products due to its predictability and high accuracy, in academic research anomaly detection is typically conceived as a more powerful method due to its theoretical potential for addressing novel attacks. Conducting a thorough analysis of the recent research trend in anomaly detection, one will encounter several machine learning methods reported to have a very high detection rate of 98 while keeping the false alarm rate at 1. However, when we look at the state of the art IDS solutions and commercial tools, there is few products using anomaly detection approaches, and practitioners still think that it is not a mature technology yet. To find the reason of this contrast, we studied the details of the research done in anomaly detection and considered various aspects such as learning and detection approaches, training data sets, testing data sets, and evaluation methods. Our study shows that there are some inherent problems in the KDDCUP 99 dataset , which is widely used as one of the few publicly available data sets for network-based anomaly detection systems . KDD CUP 99 data set description: Since 1999, KDD99 has been the most wildly used data set for the evaluation of anomaly detection methods. This data set is prepared by Stolfo et al. and is built based on the data captured in DARPA98 IDS evaluation program . DARPA98 is about 4 gigabytes of compressed raw (binary) tcp dump data of 7 weeks of network traffic, which canbe processed into about 5 million connection records, each with about 100 bytes. The two weeks of test data have around 2 million connection records. KDD training dataset consists of approximately 4,900,000 single connection vectors each of which contains 41 features. Arbitral Strategy by Neural

Network: Artificial Neural network is a powerful tool to solve complex classification problem. We do not need to force much assumption on the problem. We only need to prepare a set of inputs and targets to train it, and let the neural network learn a model. The most popular neural network is the error back-propagation (BP) neural network. A conventional BP network is a three layers feed forward network. We choose to build a conventional BP network as our final arbiter because of its simplicity and popularity. The inputs of the BP network are the prediction confidence ratios from each binary classifier. The output with maximal value is interpreted as the final class.

1.3 Problem Definition

Thinking about the fuzz it is mainly used into the software testing . To analyze the quality and the stability of the software the fuzz which can also be called as the variable input is used . I shall give its example as let my request packet contains the string as 'bappa' so that the system is designed such a way that it should handle any type of input and of largest length . So considering the limitation of the human it cannot produce the input samples of the 1000 per second so that the software program is made for that type of tasks , which produce this kind of inputs so the above input can produce as 'baaaappa','baappppppa' that is any type of input it should capable of handling.

1.4 Applying Software Engineering Approach

Software Developement Model Used:Waterfall Model

There are various software development approaches defined and designed which are employed during development process of software, these approaches are also referred as software Development Process Models? Each process model follows particular life cycle in order to ensure success in process of software development. One such approach used in software development is waterfall model? It was first process model to be introduced and followed widely in software engineering to ensure success of the project. In the waterfall approach, the whole process of software development is divided into separate process phases. The phases in the waterfall model are: Requirement specification phase, Software design, Implementation and maintenance. All these phases are cascaded to each other so that second phase is started as and when defined set of goals are achieved for first phase. General overview of waterfall model is as follows.

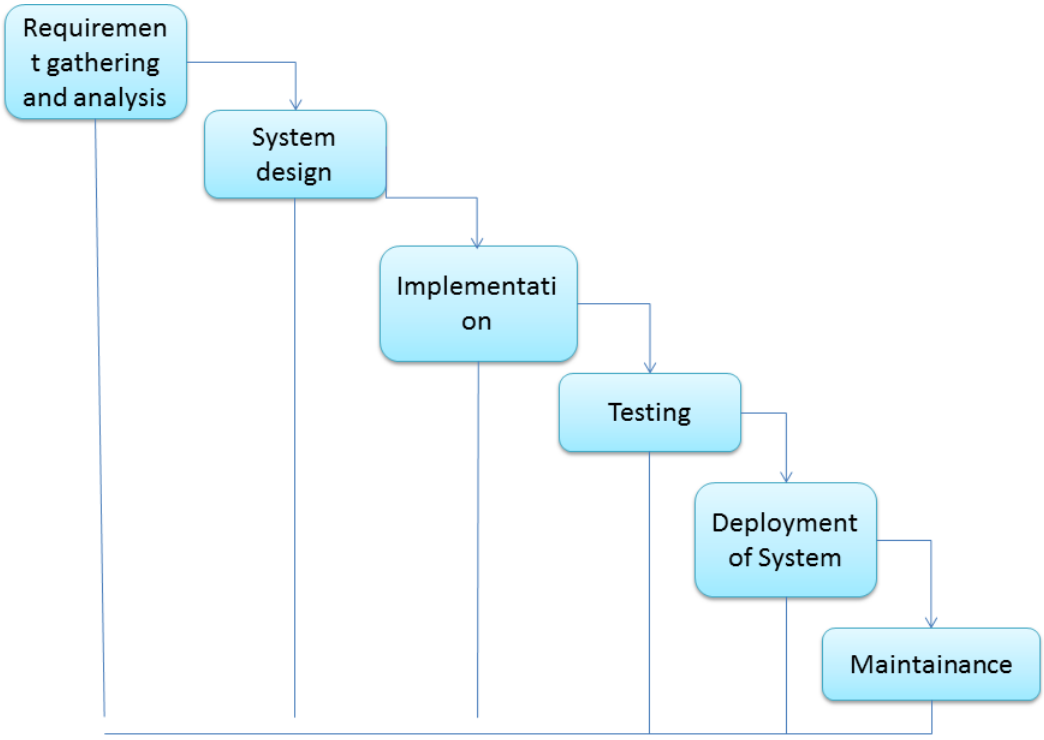

Figure 1.4.1 Stages of Waterfall Model

Stages of Waterfall Model:

1.Requirements Gathering:
Requirements from customer are collected by communicating with customer.
2.Planning and Analysis:
Analysis of gathered requirements is performed and planing and estimate of project cost and schedule is done.
3.Modelling and Design:
Model and Design of system is created as per analysis of requirements.
4.Implementation:
Actual system is implemented using 2 phases, coding and testing.
5.Deployment and Feedback:
System is deployed on user's machine and feedback is taken from user.

2 LITERATURE SURVEY

Two most significant motives to launch attacks are, either to force a network to stop some service(s) that it is providing or to steal some information stored in a network. An intrusion detection system must be able to detect such anomalous activities. However, what is normal and what is anomalous is not defined, an event may be considered normal with respect to some criteria, but the same may be labeled anomalous when this criterion is changed. applies to values inside the interval, i.e., all will be viewed as nor-mal to the same degree. Unfortunately, this causes an abrupt separation between normality and anomaly . With the fuzzy input sets defined, the next step is to write the rules to identify each type of attack. A collection of fuzzy rules with the same input and output variables is called a fuzzy system. We believe the security administrators can use their expert knowledge to help create a set of rules for each attack. The rules are created using the fuzzy system editor contained in the MATLAB Fuzzy Toolbox. This tool contains a graphical user interface that allows the rule designer to create the member functions for each input or output variable, create the inference relationships between the various member functions and to examine the control surface for the resulting fuzzy system. It is not expected, however, that the rule designer utterly relies on intuition to create the rules. Visual data mining can assist the rule designer in knowing which data features are most appropriate and relevant in detecting different kinds of attacks .The goal for using ANNs for intrusion detection is to be able to generalize from incomplete data and to be able to classify data as being normal or intrusive. An ANN consists of a collection of processing elements that are highly interconnected. Given a set of inputs and a set of desired outputs, the transformation from input to output is determined by the weights associated with the inter-connections among processing elements. By modifying these interconnections, the network is able to adapt to desired outputs. The ability of high tolerance for learning-by-example makes neural networks flexible and powerful in IDS.

Existing System:

In the literary of CAPTCHAs, most schemes were aimed at the Turing test that embeds characters in an image. However , illustrated that computer vision techniques by optical character recognition , have over 90 accuracy to recognize the character in an image. To improve the strength of a character image against to a program, tries to add more noise and distortion, but this will be harder for a human to recognize the characters too. Thus, adding too much noise and distortion will make the characters image to be unusable. Furthermore, proposed alternative image question CAPTCHAs which does not have the above issue and provided a combination of character and image CAPTCHA which possesses both of the above properties and users have to do simple mathematical computation in order to answer the question. . Two approaches to intrusion detection are currently used. The first one, called misuse detection, is based on attack signatures, i.e., on a detailed description of the sequence of actions performed by the attacker. This approach allows the detection of intrusions matching perfectly the signatures, so that new attacks performed by slight modification of known attacks cannot be detected.

Proposed System:

In our proposed system we are performing this task in different modules. We are providing a multistage detection to more precisely detect the possible attackers and a text-based Turing test with question generation module to challenge the suspected requesters who are detected by the detection module. We implemented the proposed system and evaluated the performance to show that our system works efficiently to mitigate the DDoS traffic from the Internet. In our system when client attacks on server system our system detects that attack and blocks that client and that pattern of attack is stored at admin side. If another client attacks with same pattern then that client is detected and blocked. Admin performs Turing test for client by generating questions. We are using KDDCUPSET for storing types of attacks. The client packets go through the comparing of packets with defined packets and if new pattern is detected it is stored in KDDCUPSET for prohibiting further attacks by different clients. The client who attacked with new pattern is blocked after detecting new pattern. In KDDCUPSET we are storing predefined attacks for out testing. From that KDDCUPSET we are taking patterns for attacks. We can store new patterns in that KDDCUPSET.

3 SOFTWARE REQUIREMENT SPECIFICATION

3.1 Introduction

3.1.1 Document purpose

The Project concept is to achieve the new method for extracting the information from the KDDCUP Dataset that will help to automatically detect the attack like Dos.How such attack are Detected by ANN module and provide the security from the any anomaly data which is slow your system.and provide security to the server.

3.1.2 Document conventions

The format of this SRS is simple. Bold face and indentation is used on general topics and or specific points of interest. The remainder of the document will be written using the standard font, Arial. Main Headings are indicated using Times-18 and sub headings are indicated by Times-14.

3.1.3 Intended audience and reading suggestions

This document is intended to be read by the customers like net developers, project managers, staff, users, testers and documentation writers?.This is a technical document and the terms should be understood by the customer. This SRS should be read starting with Introduction. This document is intended for: Developers: In order to be sure they are developing the right project that fulfill requirements that provided in this document. Testers: In order to have an exact list of the features and functions that has to respond according to requirements and provided diagrams. Users: In order to get familiar with the idea of the project and suggest other features that would make it even more functional. Documentation Writers: To know what features and in what way they have to explain. What security technologies are required, how the system will response on each users action etc. Advanced end users, end users/desktop and system administrators: In order to know exactly what they have to expect from the system, right inputs and outputs and response in error situations.

3.1.4 Product scope

One of the most important issues about our proposed architecture is the interaction between system-user and intrusion detection system, in order to verify predictions of the system. As means to reduce the number of interactions, system updates in presence of the user could be done in a periodically manner or at specified times that the number of wrong predictions reaches a predefined threshold.

3.2 Overall Description

3.2.1 Product perspective

This feature will give the user a secure and simple login screen.This means rather than creating try catches for a handful of error types, it just has only a handful of available and possible inputs, to prevent any improper logging in, which might cause unexpected errors, and therefore limiting the systems capabilities.and also client attack on the server by sending multiple selecting multiple attributes from KDDCUP Dataset.

3.2.2 Product functions

In this extraction framework, intermediate output of IDS is stored so that only the improved component has to be deployed to the entire database KDDCUP data set. Extraction is then performed on both the previously processed data from the unchanged components as well as the updated data generated by the improved component. Performing such kind of incremental extraction can result in a tremendous reduction of processing time. To realize this new information extraction framework, project propose to choose database management systems over file-based storage systems to address the dynamic extraction needs.

The proposed key phrase extraction method consists of four primary components: ?Document pre-processing ?Candidate phrase identification ?Information Extraction from Database Elements of the system with their functions as follow:

1. User management-username, password, add, update, login
2. Attack On Server by Providing query
3. Query sugestor-process query, map equivalent query
4. Checking source
5. Attack detection
6. log generation-user records,result,add,update,search
7. data management-Attributes,user detail,add,search
8. data extraction-query,search,extract
9. request management-request accept,Block,Unblock

3.2.3 User classes and characteristics

User classes will be Database(KDDCUP dataset), Administrator, User, Server.

3.2.4 Operating environment

This product is web-based. This product can be viewed by any web browser, and has been tested for compliance with Mozilla, Internet Explorer, Netscape Navigator, and Opera.

3.2.5 Design and implementation constraints

There are no constraints at this point in time

3.2.6 User documentation

1. Software Requirement Specification.
2. Required softwares.
3. User manual.
4. Data Flow Diagram.

3.2.7 Assumptions and dependencies

We assume that extra documentation beyond this SRS would not be necessary in order for the user to utilize this product.

3.3 External Interface Rquirements

3.3.1 User interface

The first interface is the log-in screen of Banking Application. This is where the user and Admin has a specific User-name and Password so that they can gain access to the database. Next is the Search Hints interface. Using this interface user can get hints for searching database for particular domain. Also client attacks on the server by providing anomaly query.Another is admin log in for for view all the logs of detected attacks.

3.3.2 Hardware interface

Though not necessarily interfacing with the hardware, the system must make use with an internet connection.

3.3.3 Software interface

Along with the internet connection, the system makes indirect use of an internet browser.KDD CUP 99 data set is new Database. Operating System: Windows XP/7/8

3.3.4 Communication interfaces

The system uses an internet connection to connect to the database.

3.4 System Features

3.4.1 System feature 1

Secure interface to login:
Description and Priority:
This feature will give the user a secure and simple login screen. It is based on professor Cubert? exclusionary principle. This means rather than creating try catches for a handful of error types, it just has only a handful of available and possible inputs, to prevent any improper logging in, which might cause unexpected errors, and therefore limiting the system? capabilities.

Stimulus/Responses sequences:
It will consist of two basic fields, Username and Password. There are two buttons: Login and Lost or Forgot Password. Login will submit the entered data for approval followed by access, and the forgot password will direct the user to access his/her password which has been forgotten.

Functional Requirements:
The most important function is to only grant access to users that are listed in the database. The customer will provide the information on who will be allowed access. To implement the security, the web page must check the database to see if the Username and Password are valid. If they are not, the user will receive an ?nvalid login. Please try again.?response.

3.4.2 System feature 2

Quality and Efficiency:
Such using the training and testing set automaticlly detect the attacks. Our approach minimizes the need of reprocessing the entire collection of attack in the presence of new extraction goals and deployment of improved processing speed of the server.

3.5 Other Nonfunctional Requirements

3.5.1 Performance requirements

Considering our project is totally based on the client server architecture . so that the client and server should be client to serve the request as well as the send the request. Also as the number of clients are going to be larger then that indirectly or directly server is overloaded .So that the server should client to serve all the request coming from the clients. So the hardware or the software as the server must have the networking capability. The network architecture should such a that the request/response time is measured .So that the time between request and the response should be as minimum as possible. Also the network should be scalable so that the number of clients can be increased as needed.

3.5.2 Software quality attributes

1.Adaptability:
The compiler must be able to accommodate changes to the language implementation as

well changes in the machine architecture.

2.Correctness:
The compiler generated code should give the exact output as that of the output of the script run using the interpreter.

3.5.3 Safety requirements

Other requirements should be the power supply should be uninterrupted. The networking devices should be properly connected . And faulty networking devices should be removed as early as possible such as router ,switch and the hub etc.

3.5.4 Security requirements

Access to the database should be restricted to people that are required to view information about users. Passwords and IDs should be regulated to be at least a certain length and must contain non-alphanumeric characters in both the password and ID. Access to the database should be restricted to people that are required to view information about users. Passwords and IDs should be regulated to be at least a certain length and must contain non-alphanumeric characters in both the password and ID. As we are giving the control of the whole system to the IDS and the server . So the our overall data or the database could be totally accessed by the IDS or the system/server administrator. So the any secret key and the other information about the server could not be tell elsewhere. Any security system has the limitation so that our IDS could not prevent them totally . Our software could not get the full control of the system .So try to avoiding the system calls. We will also try to implement the jre7 to take kernel level privileges to try to differentiate between the http,tcp and other types of packets.

3.6 Analysis Models

3.6.1 Data flow diagram

A data-flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. DFDs can also be used for the visualization of data processing (structured design).

On a DFD, data items flow from an external data source or an internal data store to an internal data store or an external data sink, via an internal process.

•**Level 0:** This is called as Fundamental/ context level DFD. It represents the entire software element as a single bubble with input and output data.

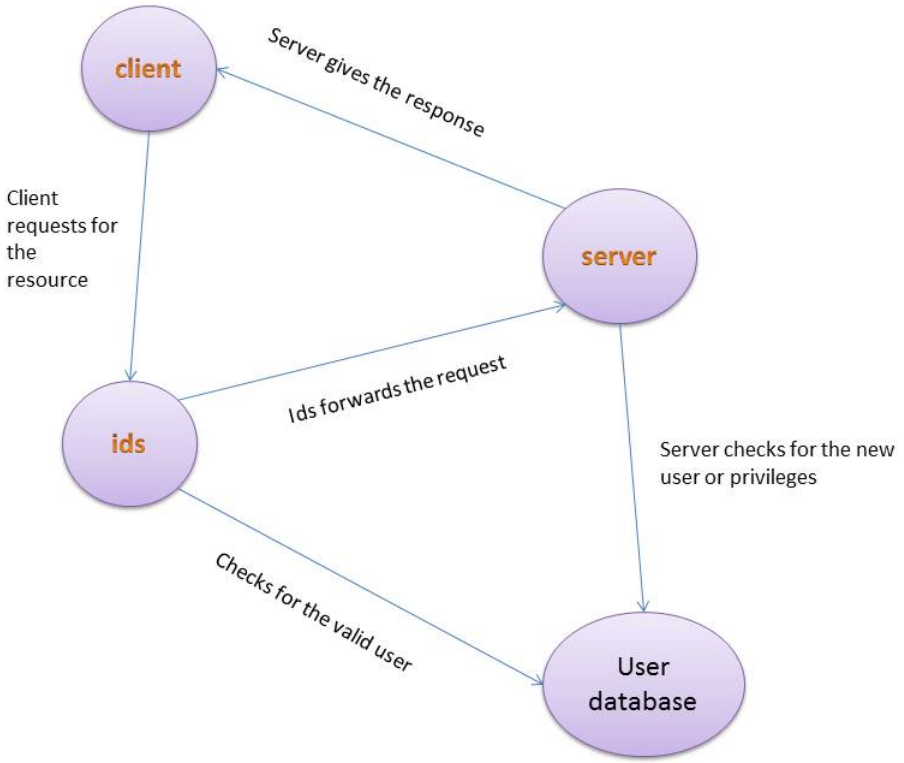

Fig. 3.6.1 Level0 DFD

•**Level 1:** In this level there is a detail description of the software where the entire software is represented by 2/3 or more bubbles.

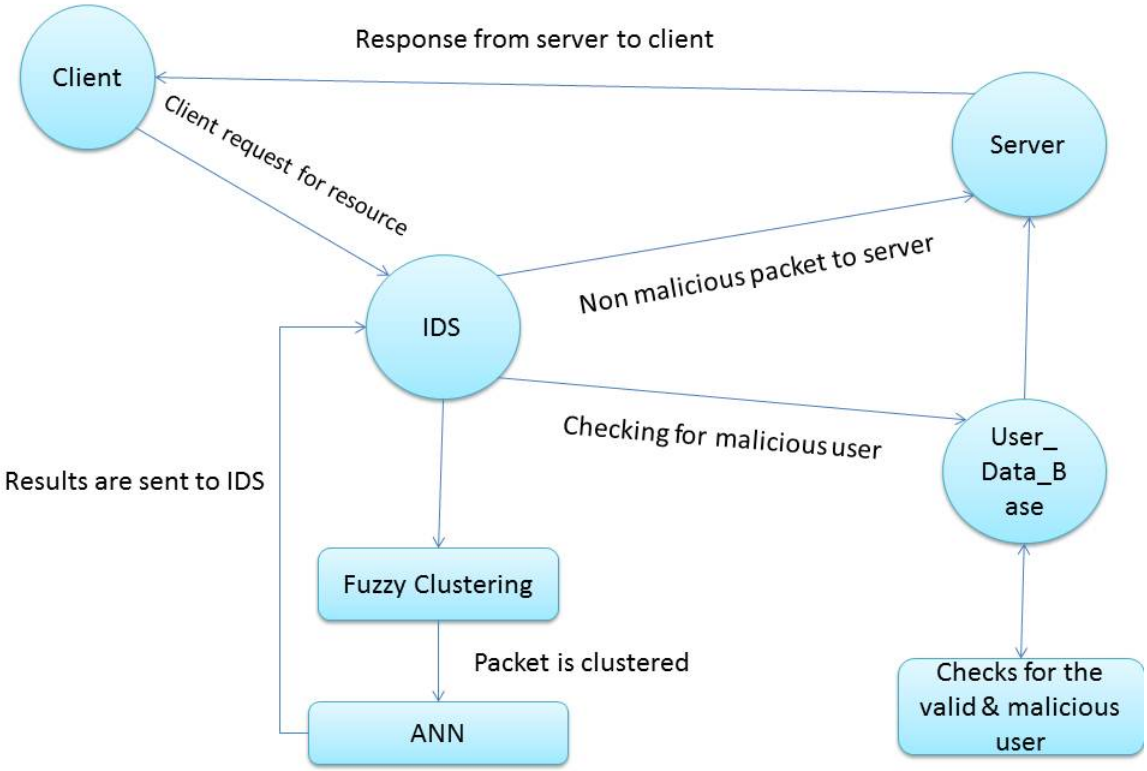

Fig. 3.6.2 Level1 DFD

A DFD provides no information about the timing or ordering of processes, or about whether processes will operate in sequence or in parallel. It is therefore quite different from a flowchart, which shows the flow of control through an algorithm, allowing a reader to determine what operations will be performed, in what order, and under what circumstances, but not what kinds of data will be input to and output from the system, nor where the data will come from and go to, nor where the data will be stored (all of which are shown on a DFD).

3.7 System Implementation Plan

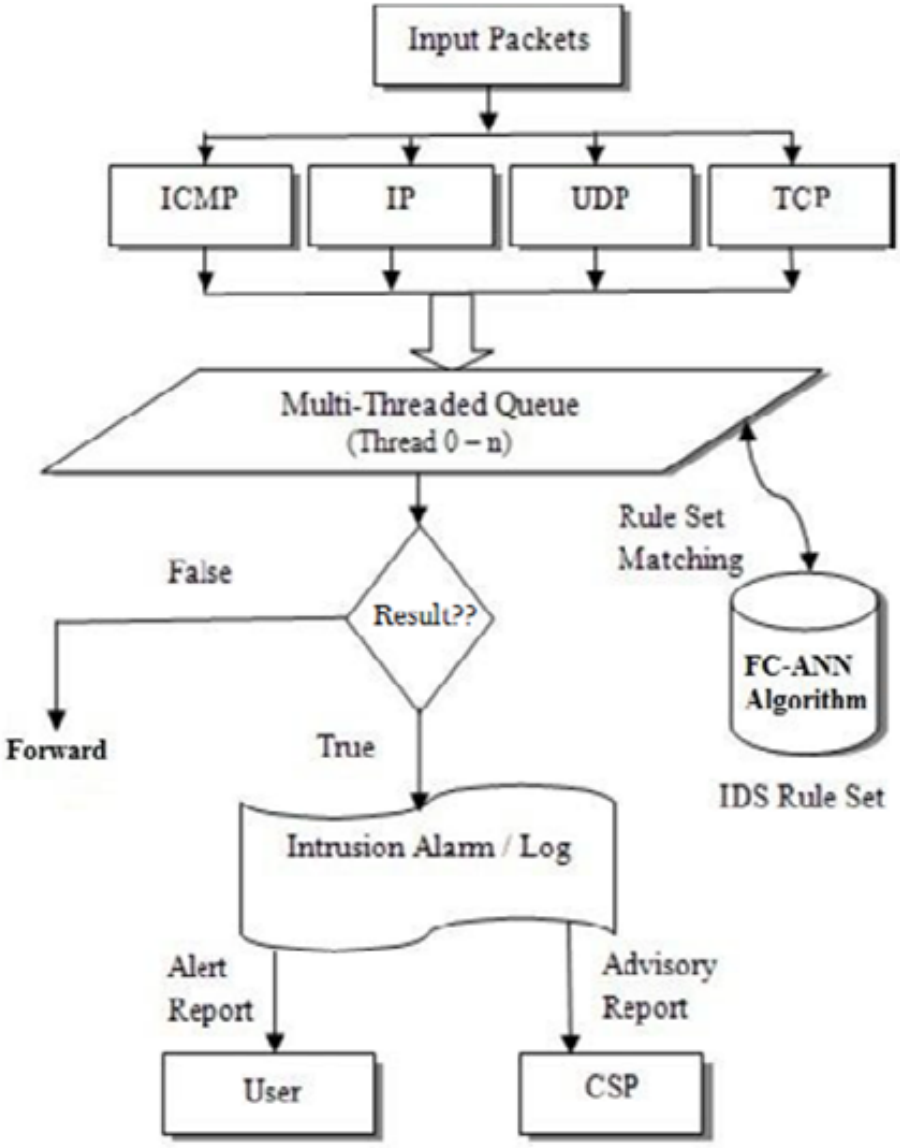

Fig. 3.7.1 System Imlementation Plan

4 SYSTEM DESIGN

4.1 System Architecture

The prime goal of our project is to protect server side resources that is to make the clients a valid request and if the any malicious activity is found then it should be handled at the IDS side and not at the server side. In the sense we can also call our system as "The Packet Inspection system". The architecture of the our system is : Figure 4.1.1 illustrates the system architecture of our approach. The architecture of the our system is :

The architecture of the our system is :

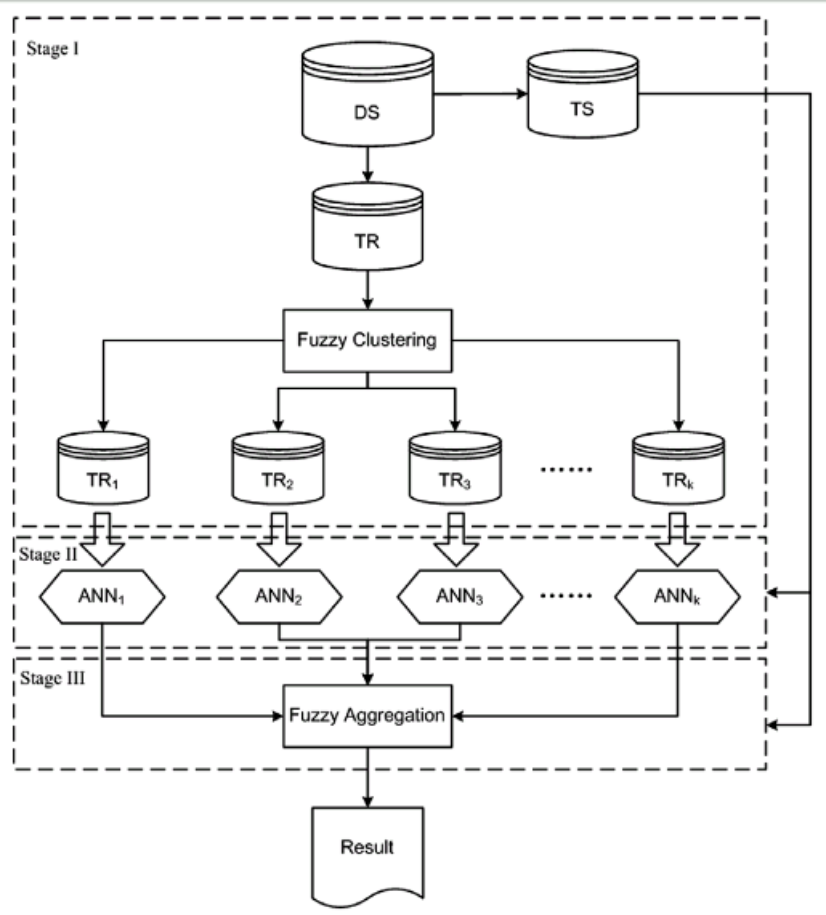

Fig. 1. Framework of FC-ANN for IDS.

Fig. 4.1.1 System Architecture

In the above to say that the IDS is situated between the client and the server , there can have multiple number of clients as well as the servers . So that each of the packet going from the client to the server is inspected at the IDS .

4.2 UML Diagrams

4.2.1 Class Diagram

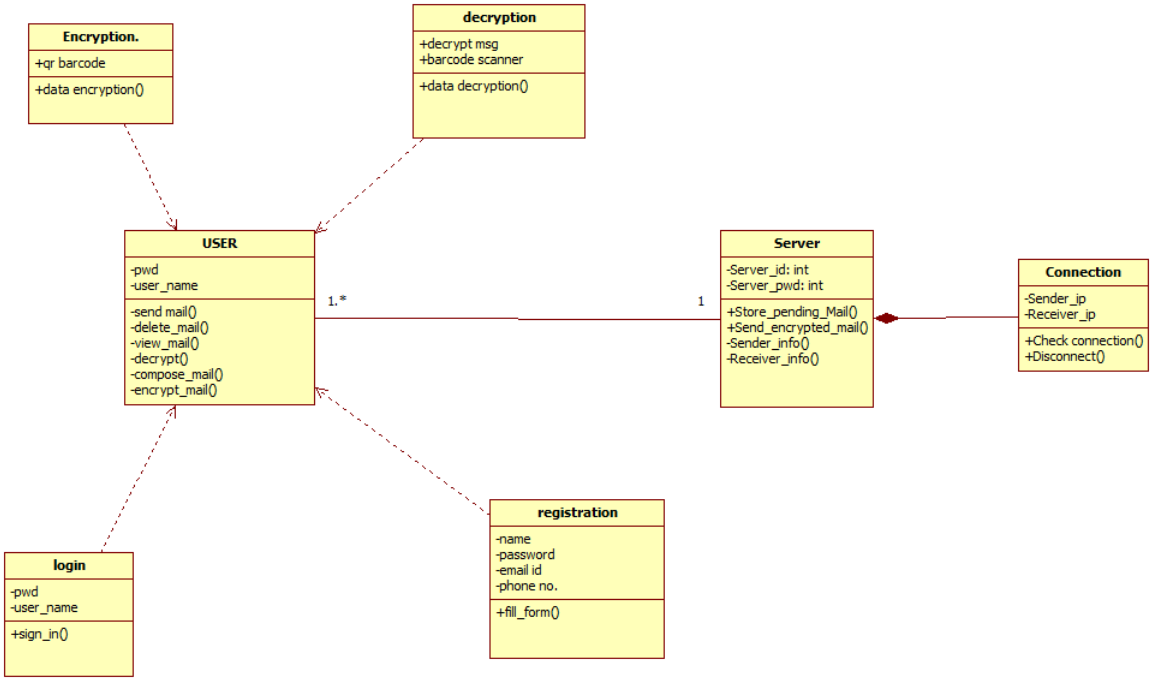

Figure 4.2.1: Class diagram.

4.2.2 Use Case Diagram

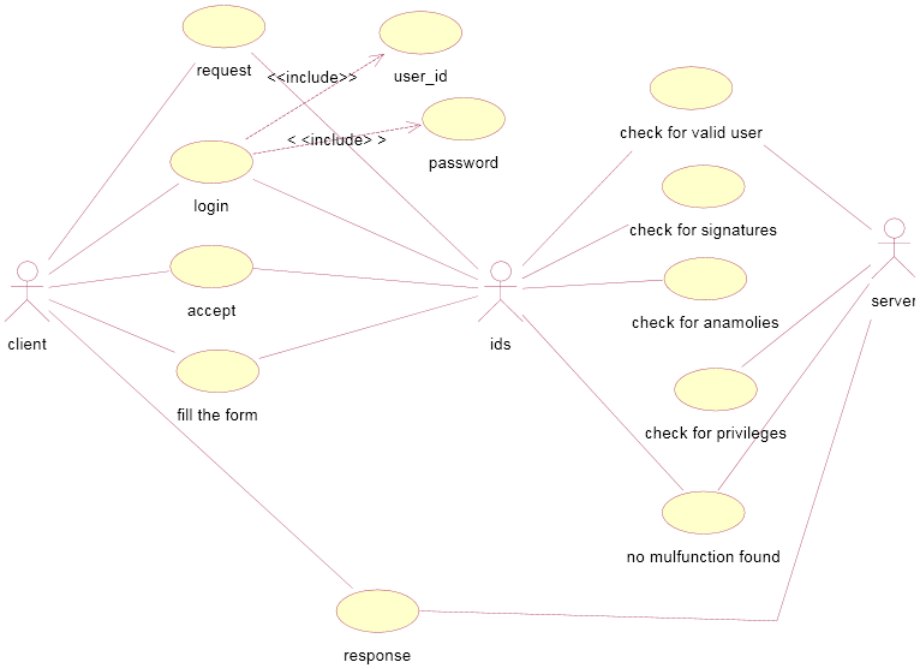

Figure 4.2.2: Usecase diagram.

4.2.3 Activity diagram

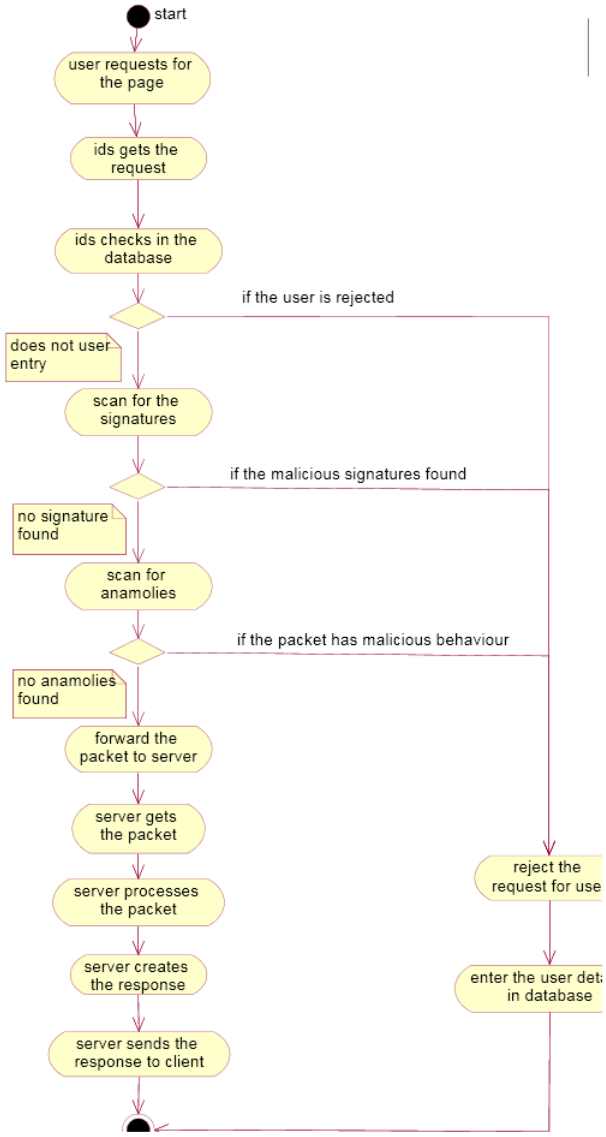

Figure 4.2.3: Activity diagram.

4.2.4 State diagram

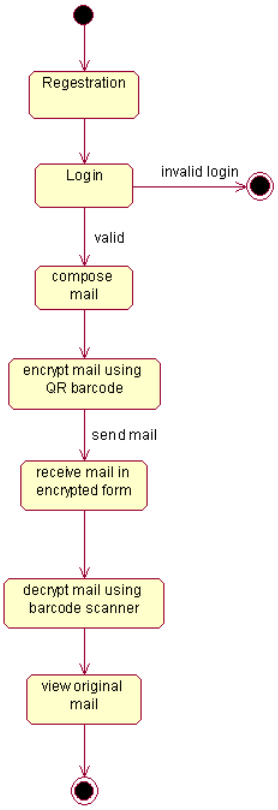

Figure 4.2.4: State machine diagram.

4.2.5 Sequence diagram

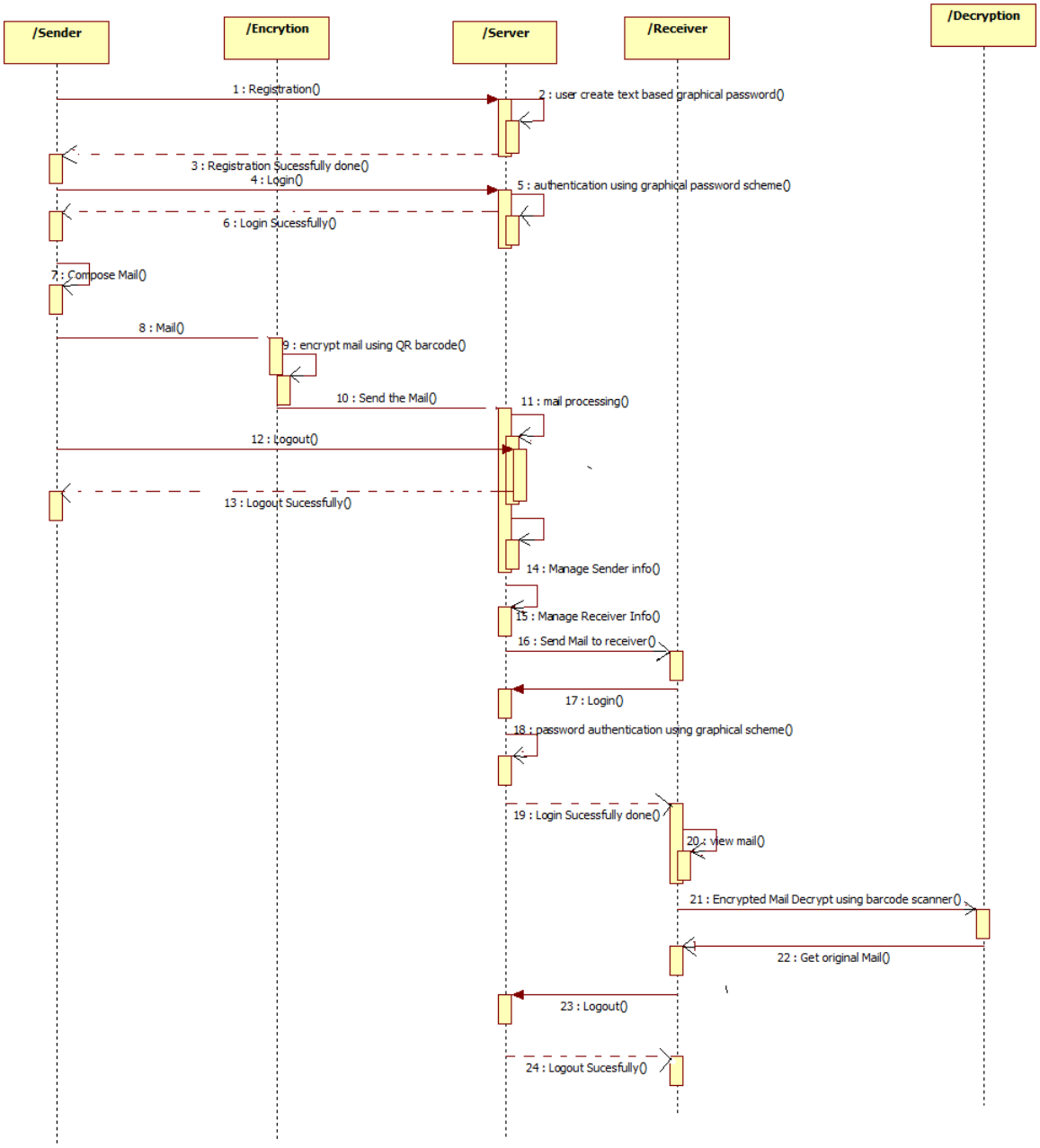

Figure 4.2.5: Sequence diagram.

4.2.6 Component diagram

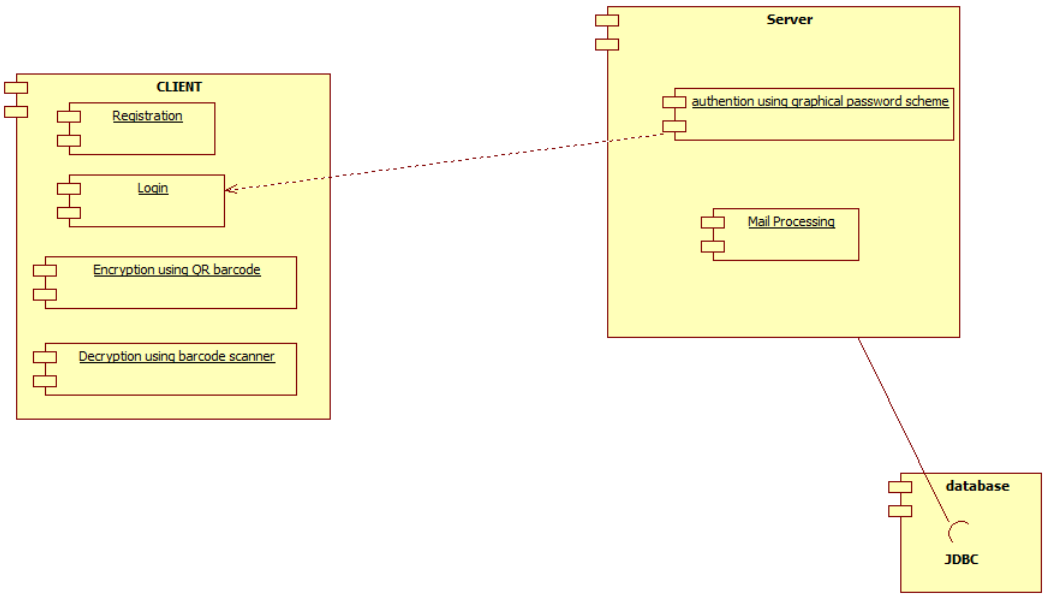

Figure 4.2.6: Component diagram.

4.2.7 Deployment diagram

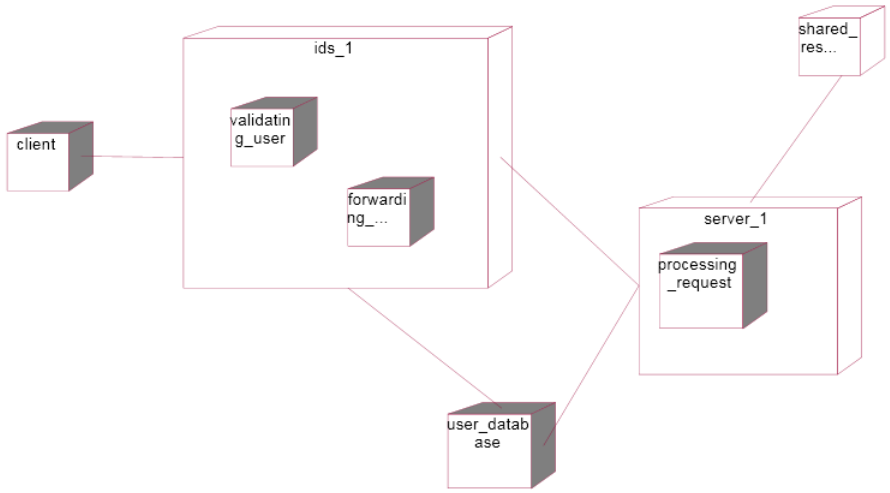

Figure 4.2.7: Deployment Diagram.

4.2.8 Package diagram

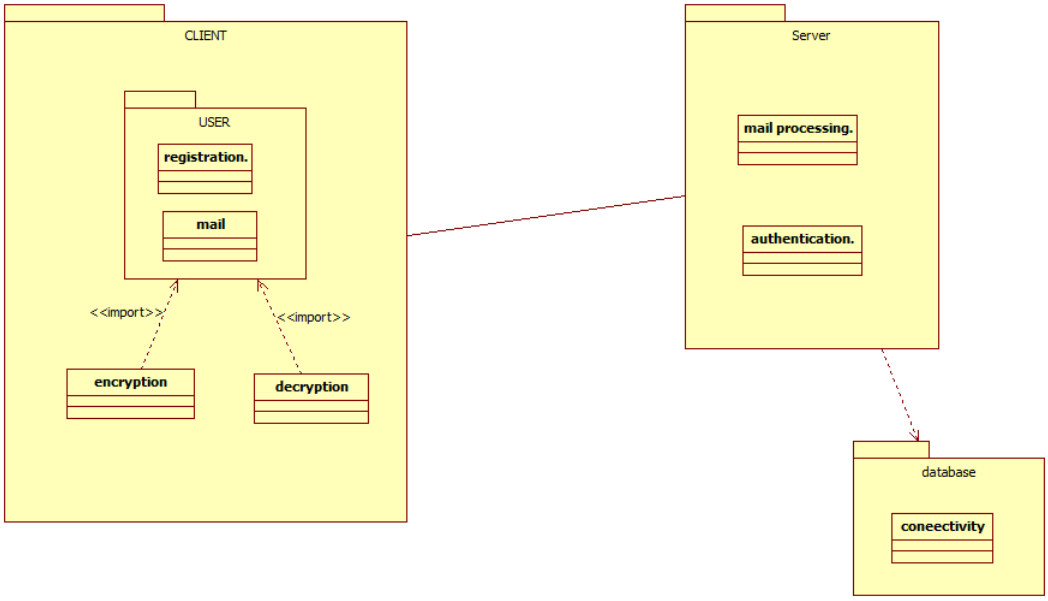

Figure 4.2.8: Package Diagram.

5 TECHNICAL SPECIFICATION

5.1 Technology Details used in project

1.JAVA

James Gosling, Mike Sheridan, and Patrick Naught on initiated the Java language project in June 1991.Java was originally designed for interactive television, but it was too advanced for the digital cable television industry at the time. The language was initially called Oak after an oak tree that stood outside Gosling's office; it went by the name Green later, and was later renamed Java, from Java coffee, said to be consumed in large quantities by the language's creators Gosling aimed to implement a virtual machine and a language that had a familiar C/C++ style of notation. Sun Microsystems released the first public implementation as Java 1.0 in 1995. It promised "Write Once Run anywhere" (WORA), providing no-cost run-times on popular platforms. Fairly secure and featuring configurable security, it allowed network- and file-access restrictions. Major web browsers soon incorporated the ability to run Java applets within web pages, and Java quickly became popular. With the advent of Java 2 (released initially as J2SE 1.2 in December 1998 1999), new versions had multiple configurations built for different types of platforms. For example, J2EE targeted enterprise applications and the greatly stripped-down version J2ME for mobile applications (Mobile Java). J2SE designated the Standard Edition. In 2006, for marketing purposes, Sun renamed new J2 versions as Java EE, Java ME, and Java SE, respectively.

Why Java?

Principles of Programming language to be efficient and java supports most of them
1.It should be "simple, object-oriented and familiar"
2.It should be "robust and secure"
3.It should be "architecture-neutral and portable"
4.It should execute with "high performance"
5.It should be "interpreted, threaded, and dynamic"

2.KDD DataSet

Since 1999, KDD99 has been the most wildly used data set for the evaluation of anomaly detection methods. This data set is prepared by Stolfo et al. and is built based on the data captured in DARPA98 IDS evaluation program . DARPA98 is about 4 gigabytes of compressed raw (binary) tcp dump data of 7 weeks of network traffic, which canbe processed into about 5 million connection records, each with about 100 bytes. The two weeks of test data have around 2 million connection records. KDD training dataset consists of approximately 4,900,000 single connection vectors each of which contains 41 features.

5.2 References to Technology

Now a day as many programming languages and platforms are getting introduced and out of those Java and .NET are the two most popular platforms which are gaining popularity and for our system development we have chosen JAVA as a platform for

different reasons such as
1. Open Source Community. The number of excellent open-source tools for Java is staggering. Look at HSqlDb, BeanShell, Eclipse, Recoder, JGraph, Tomcat, JBoss, and many more. More importantly, the Java community has proven much more interested in doing it the open-source way.
2. Eclipse. Already mentioned, but it deserves a point of its own. Eclipse is a better IDE than VS.NET!
3. Checked Exceptions.
4. Less Native Code more code reliability. .NET still has some weird crashes. Despite much improvement, I have still experienced DLL-Hell light.
5. More mature libraries.

6 PROJECT ESTIMATE,SCHEDULE AND TEAM STRUCTURE

6.1 Team Structure

Table 6.1.1: Team Structure

Sr. No.	**Name of Member**	**Designation**
1.	Tapare Prashant Bharat	Member
2.	Bhujbal Harishchandra Jalindar	Member
3.	Walkunde Kiran Baburao	Member
4.	Shinde Nandkumar Parshuram	Member

6.2 Project Estimates

- Manpower required for this project is 4 members.
- Time required for this project is 7 months for 4 members.

6.3 Schedule

Table 6.3.1: Project Scheduling

Sr. No.	Date	Topic of discussion
1.	20th July 2013	Notification About Submitting Project idea.
2.	25th July 2013	Got two areas: 1.Security in Image Processing. 2.Information Security.
3.	27th July 2013	Initial data collection related those two topics.
4.	29th July 2013	Comparative Analysis of Both Ideas.
5.	2nd Aug 2013	Selection of topic Selected topic: Information Security Intusion Detection System.
6.	3rd to 8th Aug 2013	Gathering user requirements and analysis.
7.	13th Aug 2013	Submission of Abstract.
8.	16th Aug 2013	Approval About Subject.
9.	17th to 18th Aug 2013	Literature survey About Existing systems.
10.	21th Aug 2013	Platform Selection.
11.	24th to 27th Aug 2013	Information Collection Related to platform.
12.	2th Sept 2013	Requirement Elaboration.
13.	4th,6th Oct 2013	Modeling Behavioral View.
14.	8th 12th Oct 2013	Modeling Structural View.
15.	15th to 20th Oct 2013	SRS Creation.
16.	22nd to 23th Oct 2013	Project analysis regarding NP-hard NP-Complete.
17.	24th to 25th Oct 2013	Mathematical Model Designing.
18.	25th to 30th Oct 2013	Document creation for Term-1.
19.	29th Nov 2013	Presentation of Term-1.
20.	12th to 27th Jan 2014	Simple Banking Application in Java.
21.	28th to 29th Jan 2014	Data Extraction from KDD cup dataset .
22.	30th to 10th Jan 2014	Packet Analysis.
23.	11th to 12th Feb 2014	CAPTCHA and Turing test.
24.	13th to 20th Feb 2014	Online DoS attack .
25.	21st to 22nd Feb 2014	Log generation for administrative purpose.
26.	23rd to 25th Feb 2014	Grammer Check and Information Extraction module.
27.	4th to 10th March 2014	Combined testing of all modules.
28.	11th to 15th March 2014	Bug fixing and Error Handling.
29.	25th March 2014	Document creation for Term-2.

7 SOFTWARE IMPLEMENTATION

7.1 Introduction

To overcome this drawback we are developing new model of Intrusion Detection System which has capacity of self detecting or updating attacks. In proposed IDS model we are develop Artificial Neural Network algorithm with fuzzy logic to detect and update database for newly attacks. in proposed model we define two separate set of data. 1] Training set 2] Testing set. In training set every user query checked using apriori algorithm and fuzzy algorithm .In training set we use apriori, artificial neural network, clustering algorithm for train the user query and database. In testing set we compare every user query with exiting database. We use KDD CUP dataset as exiting database which is developed in 1999 by Sun Microsystems computers..

7.2 Databases

For database operations and to store tables KDD Data Cupset is used.

On client machine client has to provide only Server name,Database name and User ID while setting connection.

7.3 Important Modules and Algorithms

This project mainly focuses on natural language processing and domainwise extraction using Automated Query Generation. To implement this following modules are used:

1. Checking source
    - In this module we are checking the source of attack. We are providing authentication for client for login. If client attacks with some pattern then by identifying that clients IP address we finding its source.
2. Counting
    - In this module we are recording the source address destination address and the time at which client performs login test. After login successful the counting module is reset. It will be enable by the Attack Detection module when there are some suspected traffic been detected.
3. Attack detection
    - In this section, we elaborate our new approach; FC-ANN. FC-ANN firstly divides the training data into several subsets using fuzzy clustering technique. Subsequently, it trains the different ANN using different subsets. Then it determines membership grades of these subsets and combines them via a new ANN to get final results
4. Turing Test Module

- In this module the client is provided with some CAPTCHA code which client will input through keyboard, doing this admin will understand that the client is a human not a machine.

5. Question Generation Module

   this module if client fails to perform Login then admin will ask some questions which client has to answer perfectly. The question will be stored by admin at the time of client registration.

8 SOFTWARE TESTING

8.1 Introduction

Testing is process of ensuring that software function as per user needs.

Types of software testing:

1. White Box Testing
   It is a test case design philosophy that uses the control structure described as part of component level design to derive test cases.

   - Basis Path Testing
     Basis path method enables designer to derive a logical complexity measure of a procedural design and use this measure as a guide for defining a basis set of execution paths.
   - Contol Structure Testing
     It tests control structures of program.

2. Black Box Testing
   Black box testing enables the software designer to derive sets of input conditions that will fully exercise all functional requirements of a program.

   - Graph Based Testing Method
   - Equivalence Partitioning
   - Boundary Value Analysis
   - Orthogonal Array Testing

3. Integration Testing
   Intgration Testing is a systematic technique for constructing software architecture while at the same time conducting tests to uncover errors associated with interfacing

4. Regression Testin
   Each time a new model is added as part of integretion testing ,the software changes. At this time regression testing is applied.

8.2 Test Cases

Test Cases for project:-

Table 8.2.1: Test case for New Registration Module

Test No.	**Test Description**	**Test Condition**	**Expected result**
1.	New registration	New registration of new client	If client is not registered then he has to do newly registration

Table 8.2.2: Test case for User Log in Module

Test No.	Test Description	Test Condition	Expected result
1.	Input for user name	Client has to enter user name	Client has to enter proper user name which start with letter not with digit or special character
2.	Input for password	Client has to enter password	Password should have at least 6 characters, special Symbols, digits except white space

Table 8.2.3: Test case for Client provide attack and displaying result

Test No.	Test Description	Test Condition	Expected result
1.	Select Attributes	Fire Attack	User has fired
2.	Display result	check for correct domain and generate result	result is displayed successfully

Table 8.2.4: Test case for Detect Attack.

Test No.	Test Description	Test Condition	Expected result
1.	Admin Login	Show Block Users	View all Logs of attack on server with IP

8.3 Snapshots of Test Cases and Test Plans

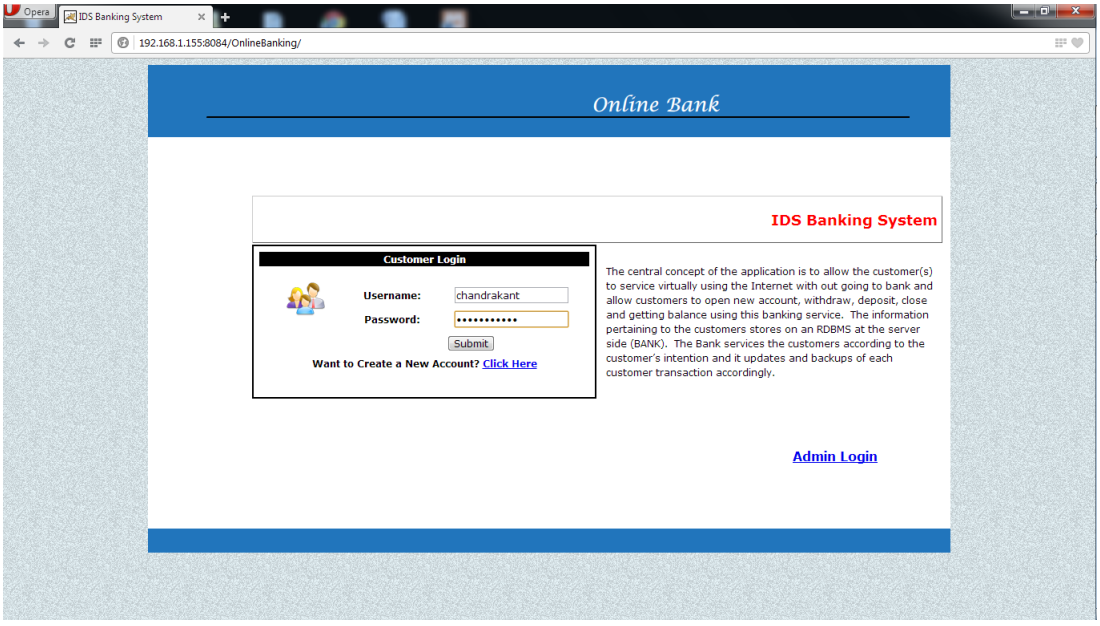

Figure 8.3.1: Simple user login:

S

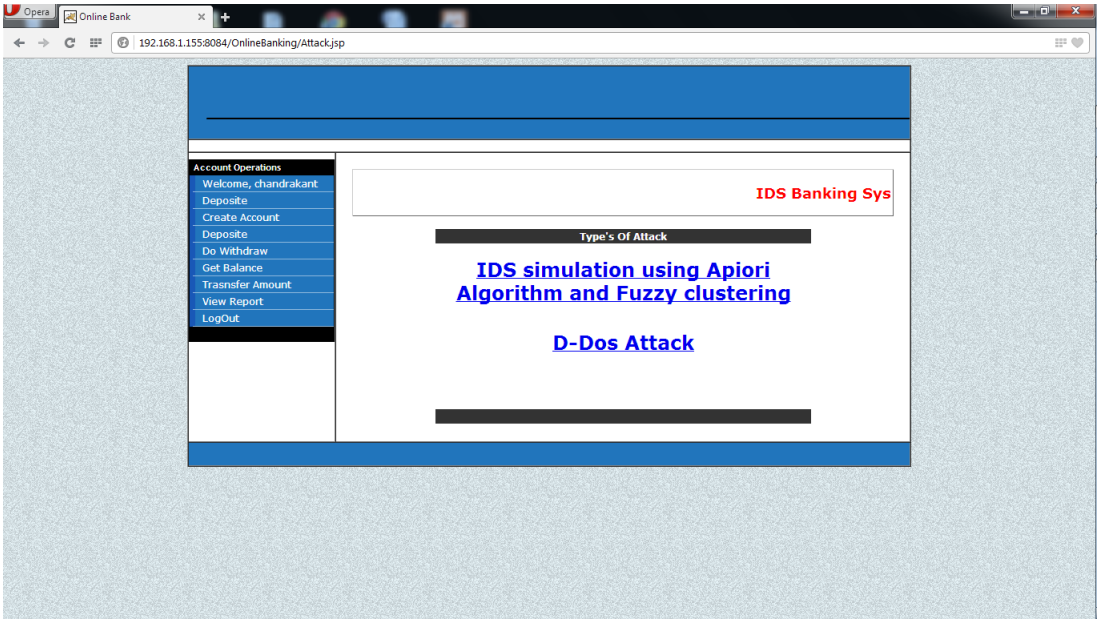

Figure 8.3.2: Attack options for the user

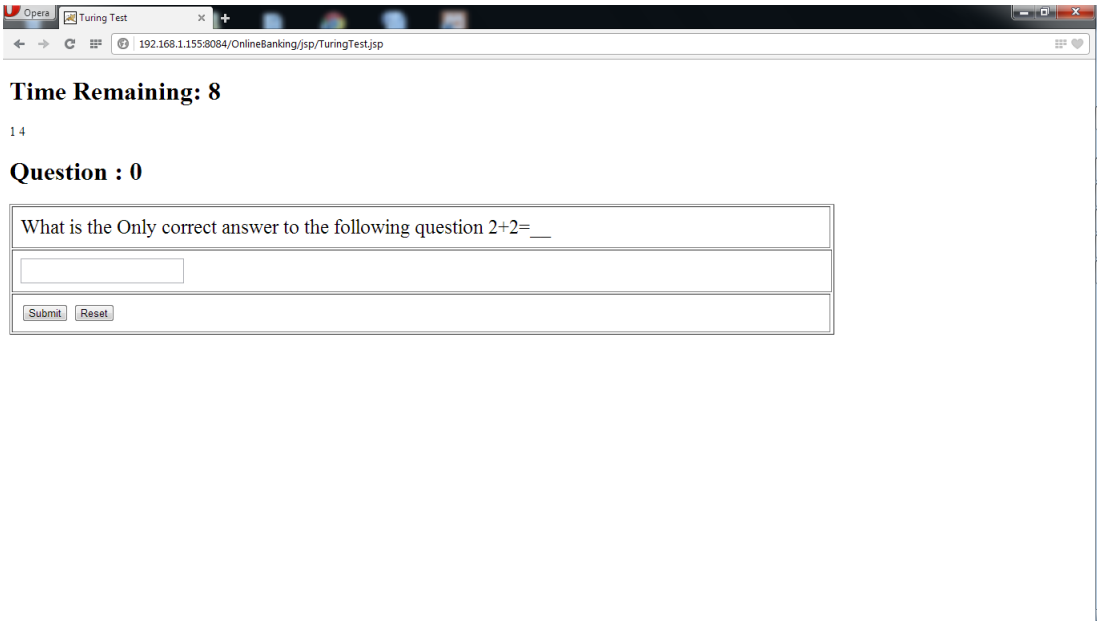

Figure 8.3.3: Turing test

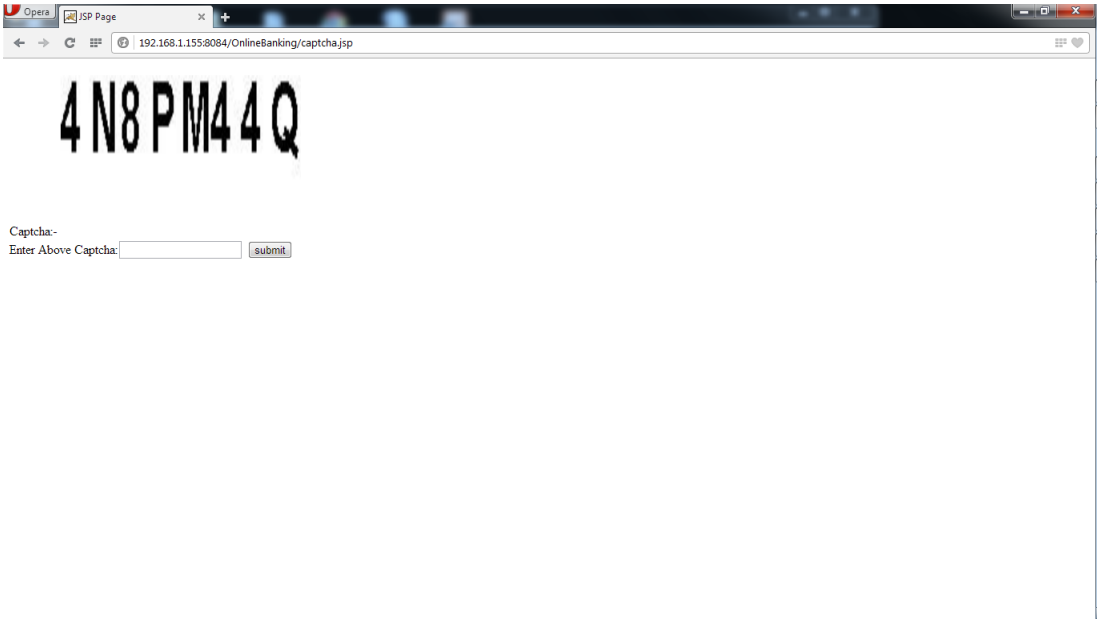

Figure 8.3.3: CAPTCHA

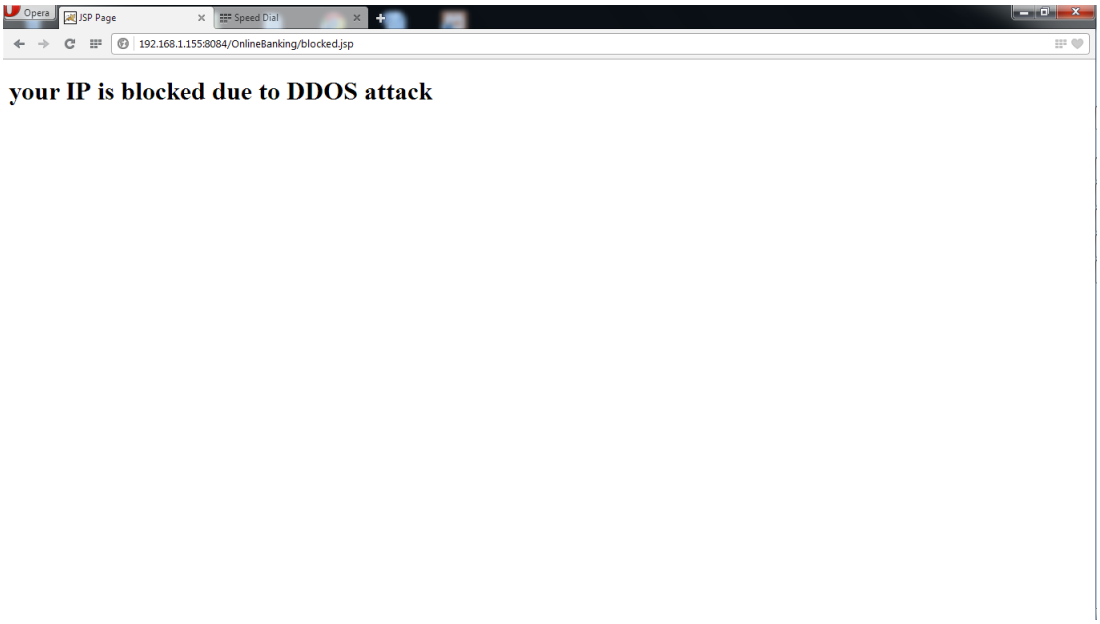

Figure 8.3.3: Block IP

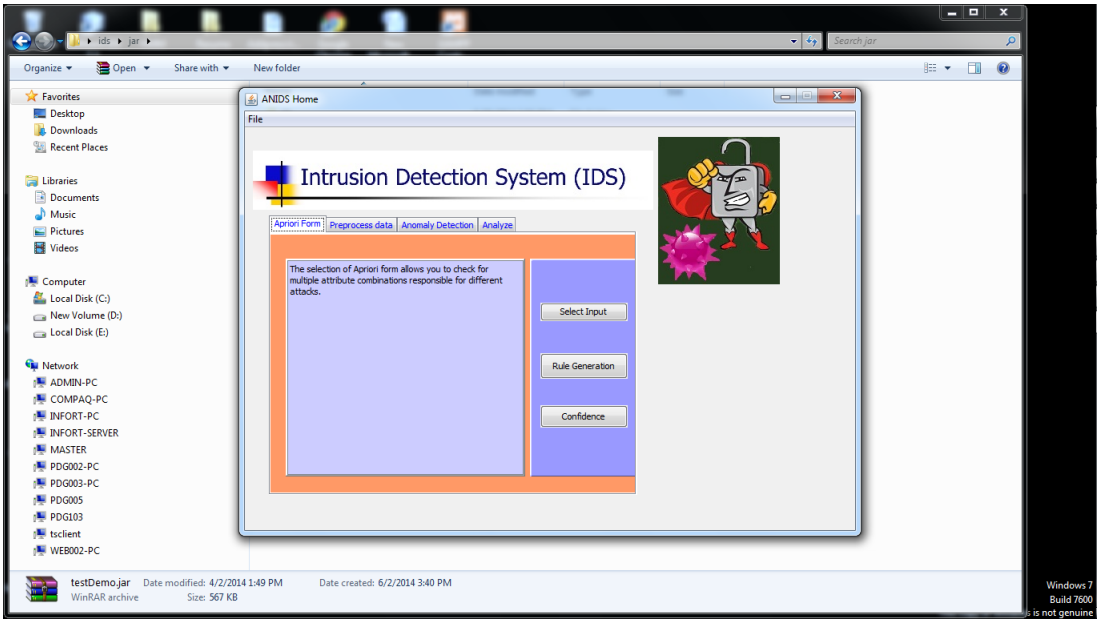

Figure 8.3.4: User generated Attack

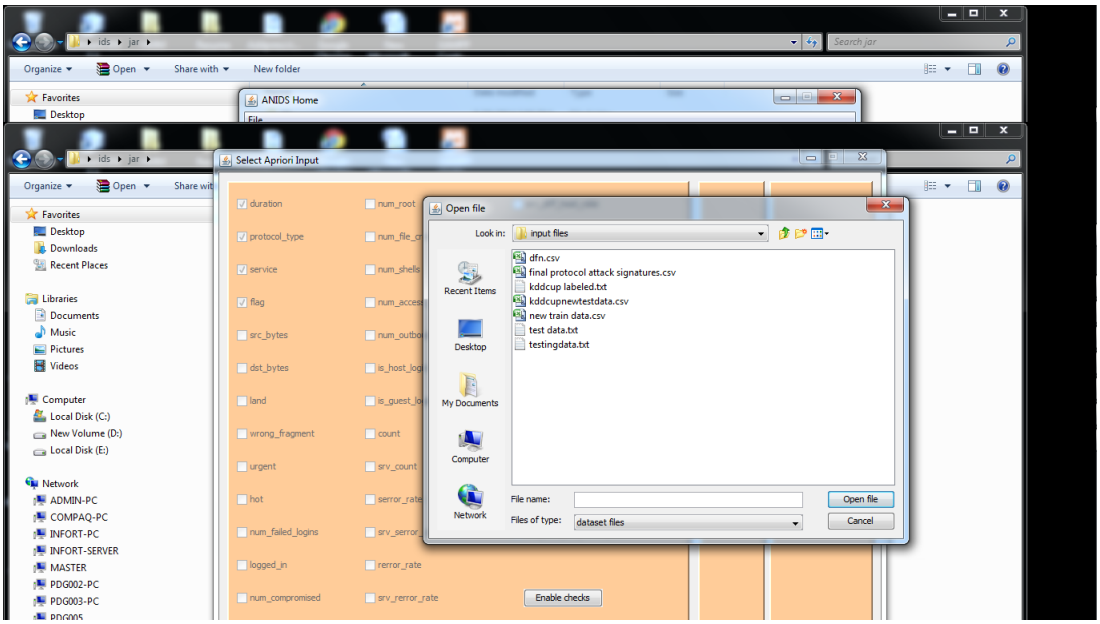

Figure 8.3.5: Selecting attribute set:

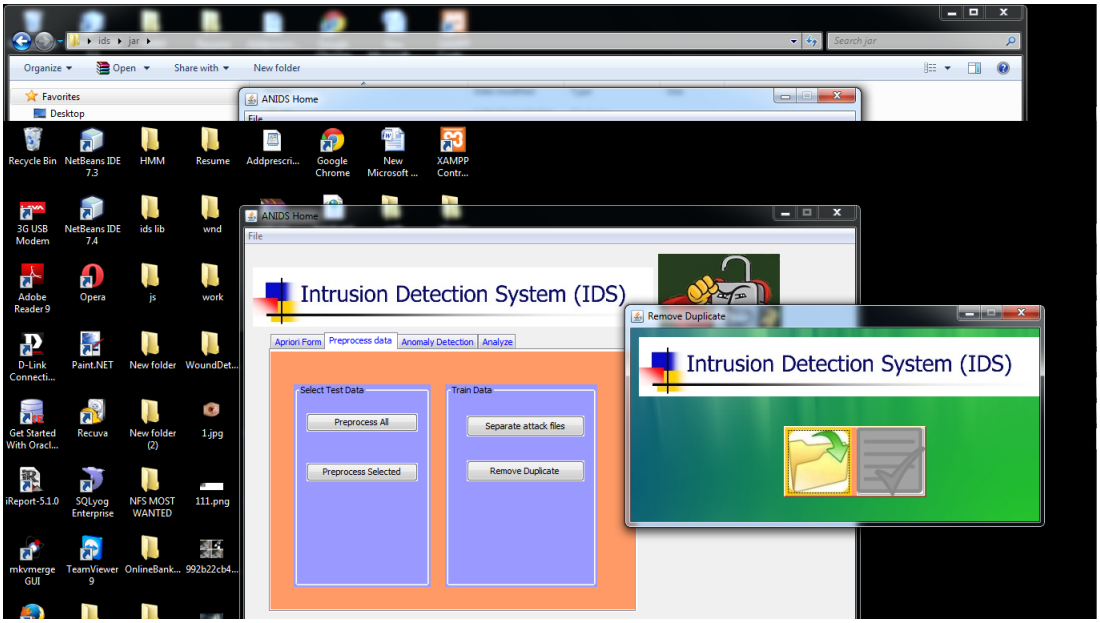

Figure 8.3.6: Testing and Training set:

9 RESULTS

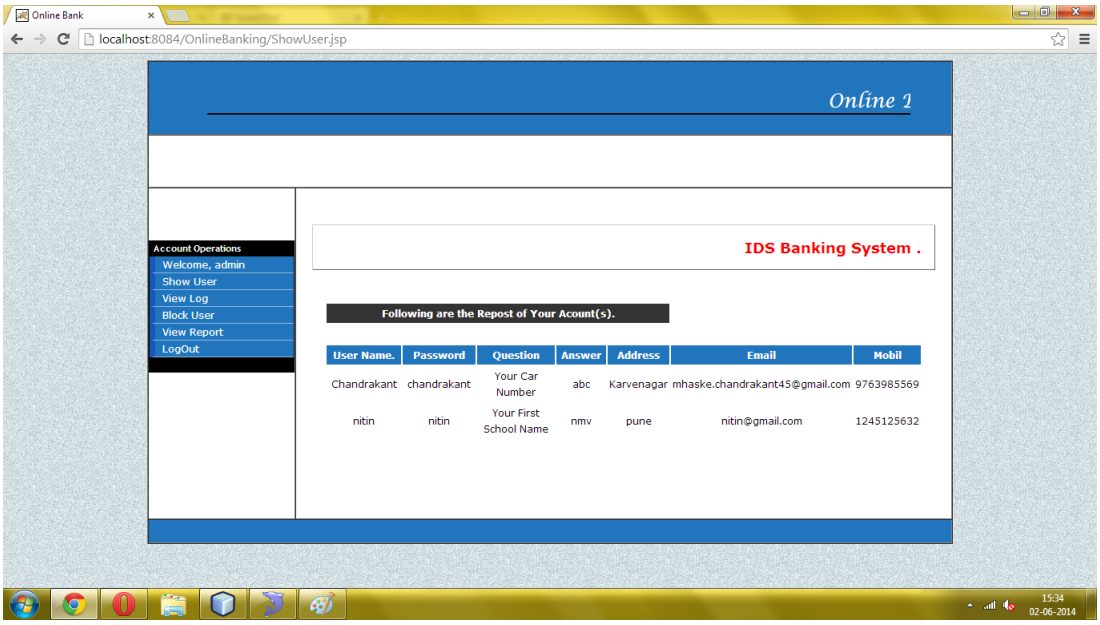

Figure 9.1: Admin Login

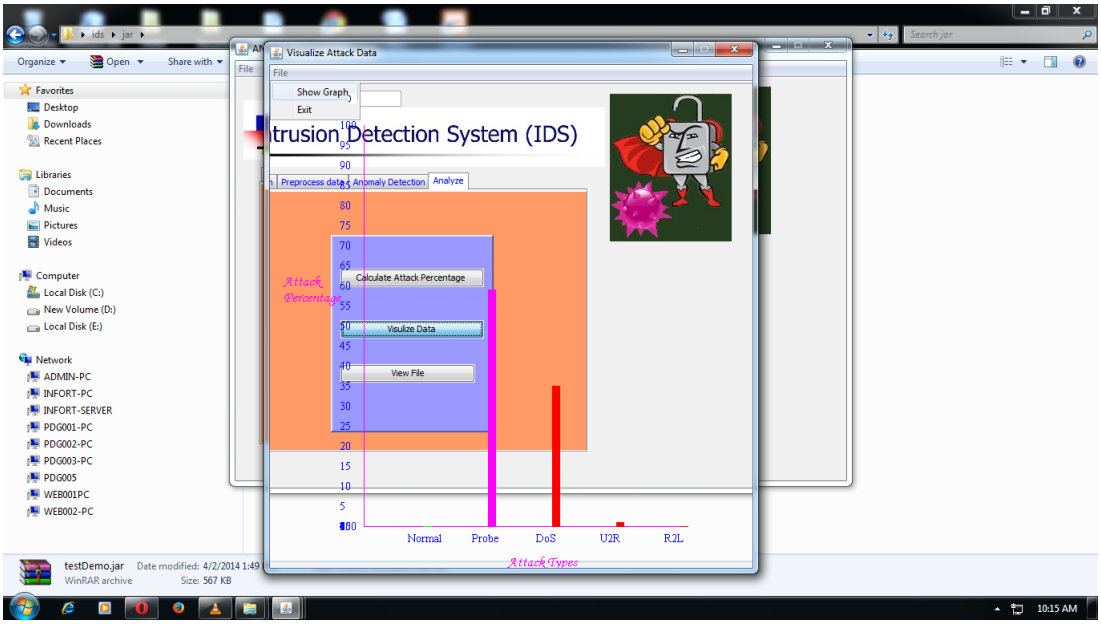

Figure 9.2: Anomaly Detection of attack :

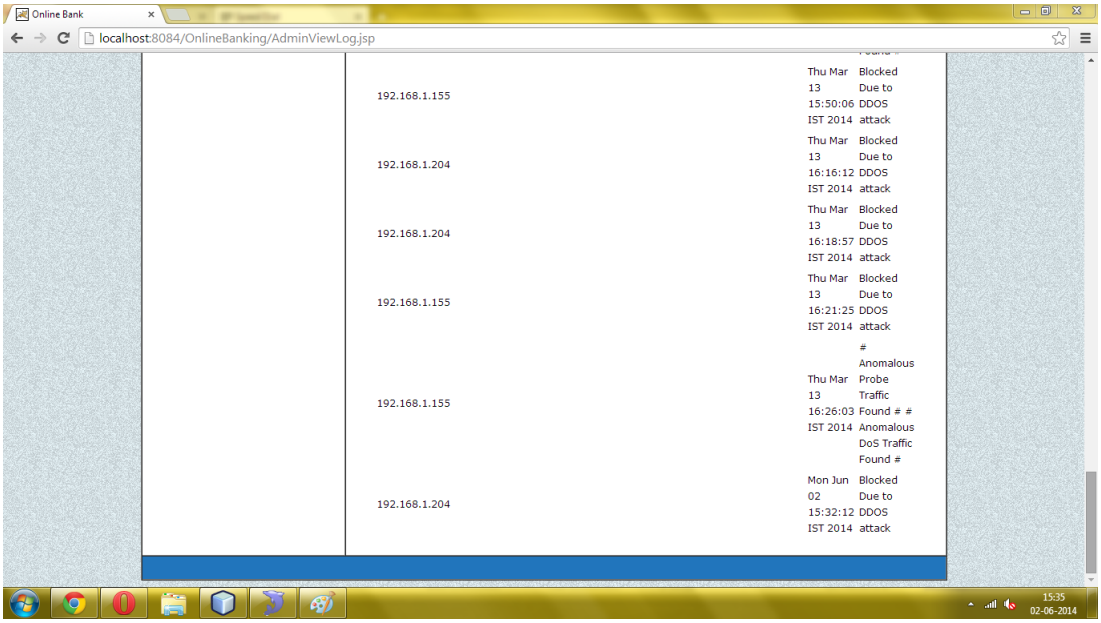

Figure 9.3: Logs Of All Attacks.

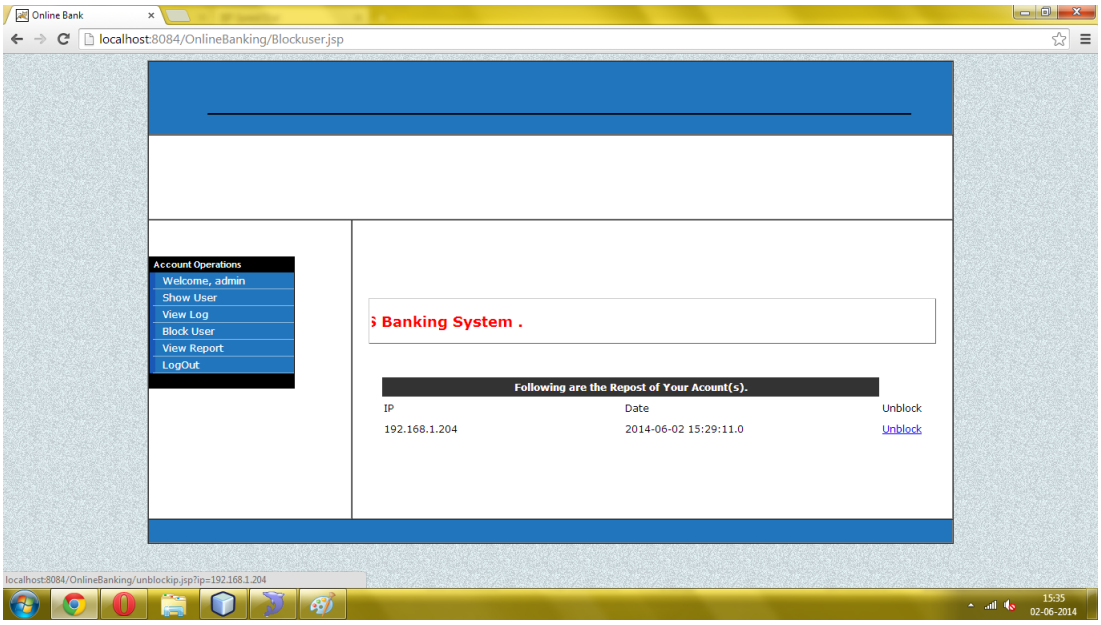

Figure 9.3: Block IP.

10 DEPLOYMENT AND MAINTANANCE

10.1 Installation and Un-Installation

Deployment starts directly after the code is appropriately tested, approved for release, and sold or otherwise distributed into a production environment. This may involve installation customization (such as by setting parameters to the customer's values), testing, and possibly an extended period of evaluation. Software Deployment is all of the activities that make a software system available for use. All machines should have JDK installed on them.Software training and support is important, as software is only effective if it is used correctly. Maintaining and enhancing software to cope with newly discovered faults or requirements can take substantial time and effort, as missed requirements may force redesign of the software. Bug fixes, Patches, Service Packs, New Releases, etc. Client have compulsory install of jdk.

10.2 User Help

1.Register using username and password

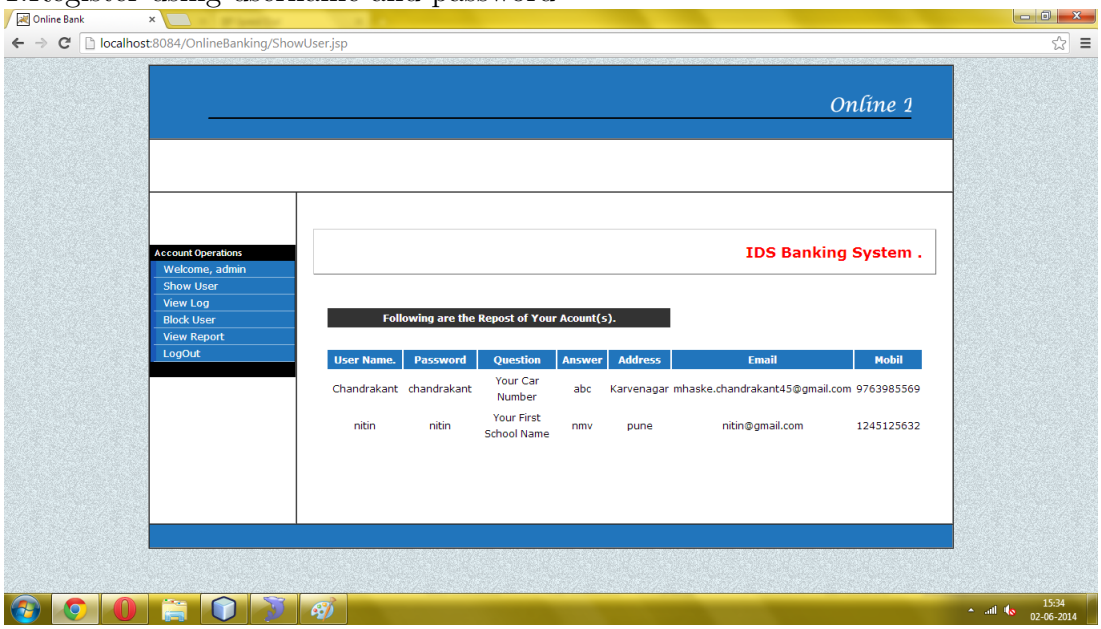

Figure 10.2.1: Registeration

2.Admin Login to get Result

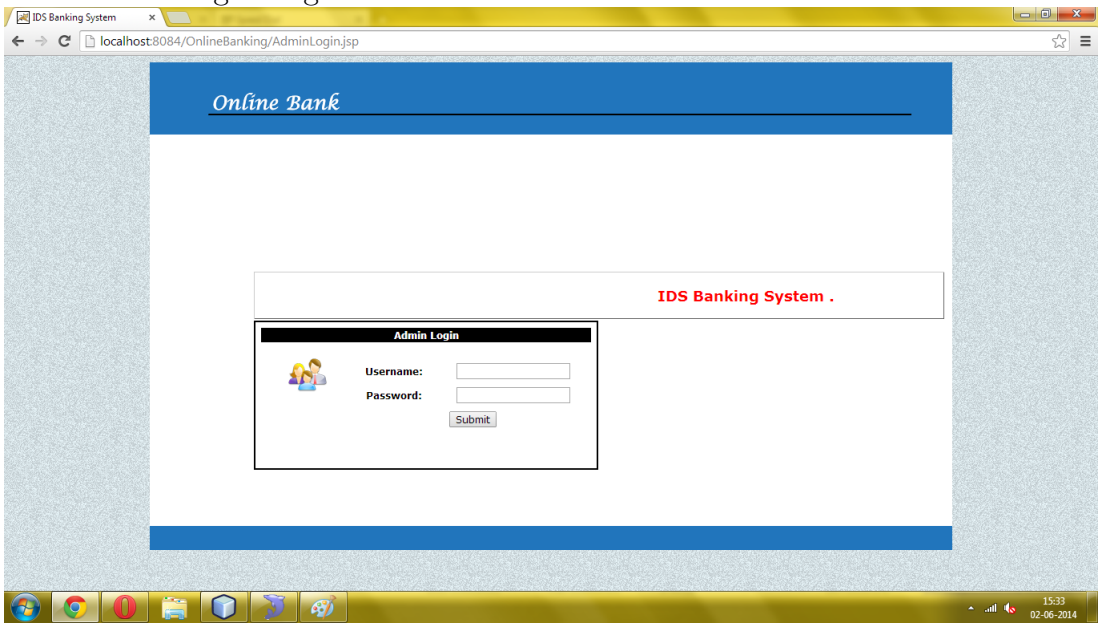

Figure 10.2.6: Get Result of Query

3.Block Of IP

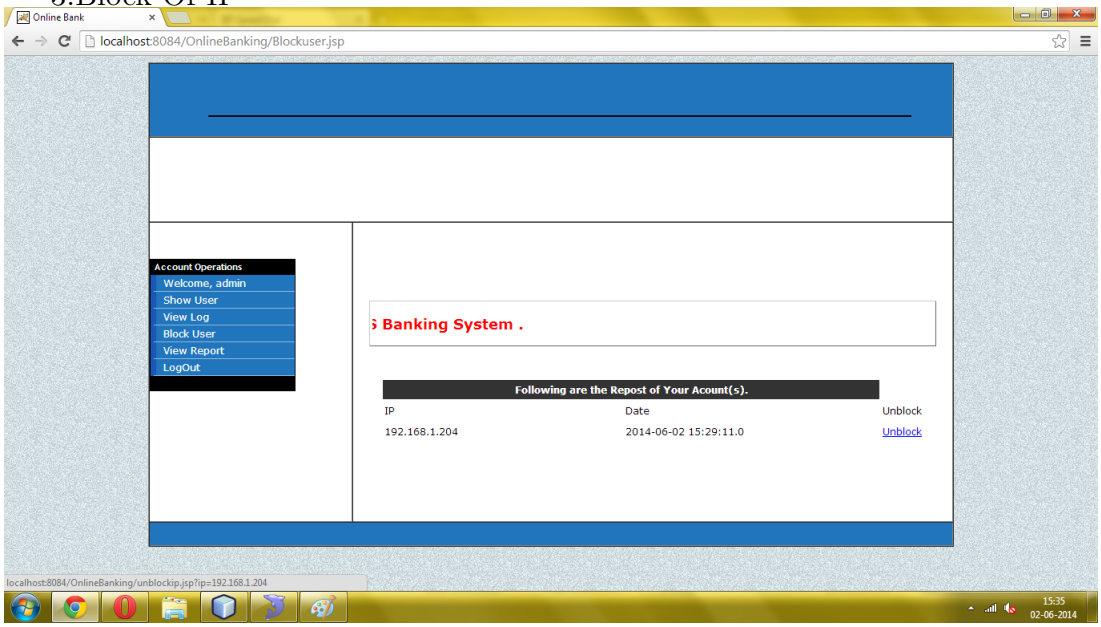

Figure 9.3: Block IP.

11 CONCLUSION AND FUTURE SCOPE

Conclusion:
Prevention of security breaches completely using the existing security technologies is unrealistic. As a result, intrusion detection is an important component in network security. IDS offers the potential advantages of reducing the man power needed in monitoring, increasing detection efficiency, providing data that would otherwise not be available, helping the information security community learn about new vulnerabilities and providing legal evidence. In this system, we propose a new intrusion detection approach, called FC-ANN, based on ANN and fuzzy clustering. Through fuzzy clustering technique, the heterogeneous training set is divided into several homogenous subsets. Thus complexity of each sub training set is reduced and consequently the detection performance is increased. The experimental results using the dataset demonstrates the effectiveness of our new approach especially for low-frequent attacks, i.e., R2L and U2R attacks in terms of detection precision and detection stability. In future research, how to determine the appropriate number of clustering remains an open problem. Moreover, other data mining techniques, such as support vector machine, evolutionary computing, outlier detection, may be introduced into IDS. Comparisons of various data mining techniques will provide clues for constructing more effective hybrid ANN for detection intrusions.

Future Scope:
1.This system can be used as core Information Extraction system for all types of information systems having large databases.
2.This system can be extended to system which will take input in any language.
3.This system can also be extended using for business Security.
4.Better algorithms can be developed to increase efficiency and quality of results.

REFERENCES

References

[1] Gang Wang a,b,*, Jinxing Hao b, Jian Mab, Lihua Huanga, A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering, Expert Systems with Applications xxx (2010) xxxxxx

[2] G. Goth, Fast-moving zombies: Botnets stay a step ahead of the fixes, IEEE Internet Computing, vol. 11, pp. 79, 2007.

[3] G. Goth, Fast-moving zombies: Botnets stay a step ahead of the fixes, IEEE Internet Computing, vol. 11, pp. 79, 2007...

[4] Vincent Shi-Ming Huang , Robert Huang, Ming Chiang, A DDoS Mitigation System with Multi-Stage Detection and Text-Based Turing Testing in Cloud Computing, 2013 27th International Conference on Advanced Information Networking and Applications Workshops.

[5] Hassan M. Najadat, Mohammed Al-Maolegi, Bassam Arkok, An Improved Apriori Algorithm for Association RulesJune 2013...

[6] Manoranjan Pradhan, Sateesh Kumar Pradhan, Sudhir Kumar Sahu, Anomaly Detection Using Different Artificial Neural Network Training FunctionsApril 2012...

[7] German Florez, Susan M. Bridges, and Rayford B. Vaughn, An Improved Algorithm for Fuzzy Data Mining for Intrusion Detection...

[8] Professor Anita Wasilewska Lecture Notes, APRIORI Algorithm...

[9] Safaa O. Al-mamory, Evaluation of Different Data Mining Algorithms with KDD CUP 99 Data Set...

APPENDIX

Appendix A: Glossary

List of Abbreviations

Sr. No.	Abbreviation	Meaning
1	IDS	Intrusion Detection System
2	Dos	Denial of Service
3	KDD	knowledge Discover Data Mining
4	ANN	Artificial Neural Networks
5	TS	Testing Set
6	TR	Traning Set