

P-1905

INTRUSION DETECTION SYSTEM

By

V. Rajesh

Reg. No. 71204621029

Of

**KUMARAGURU COLLEGE OF TECHNOLOGY
COIMBATORE**

A PROJECT REPORT

Submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements

for the award of the degree

of

MASTER OF COMPUTER APPLICATIONS

JULY, 2007



P-1905

KUMARAGURU COLLEGE OF TECHNOLOGY
COIMBATORE – 641006.

DEPARTMENT OF COMPUTER APPLICATIONS

BONAFIDE CERTIFICATE

Certified that this project report titled **Intrusion Detection System** is the bonafide work of **Mr. V. Rajesh (Reg.no7120462129)** who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.



Project Guide

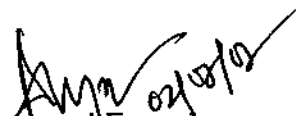


Head of Department

Submitted for the University Examination held on 02-07-2007



Internal Examiner



External Examiner

May 15th 2007

To Whomsoever It May Concern

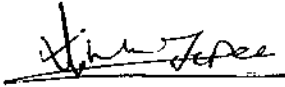
This is to inform you that **V.RAJESH** has successfully completed his project assignment titled **INTRUSION DETECTION SYSTEM** as a part of MCA curriculum.

As a Project Trainee, he started this project on **December 18, 2006** and completed it on **May 15, 2007**.

Please note, as per the company's policies and practices, the company retains ownership of the intellectual property rights concerning work undertaken during projects and disclosure of the source code and any other relevant information or data out of the organization is strictly prohibited.

V.RAJESH designated, as a project trainee will not be delivering the respective source code pertaining to his project.

For Maze Net Solution,



Krupa Japee
Administrator.

ABSTRACT

The project aims at securing the important documents by collecting them into one Intrusion detection system is the process of monitoring the events occurring in a computer system or network and analyzing them for signs of intrusions, defined as attempts to compromise the confidentiality, integrity, availability or to bypass the security mechanisms of a computer or network.

Intrusion detection systems are also classified based on the types of systems they monitor. The two main systems monitored for intrusions are host-based systems and network based systems.

Host-based intrusion detection attempts to detect against attacks on a particular machine. This is typically done through analysis of a computers log files.

Host based IDS typically monitor system, event, and security logs on Windows NT and syslog in Unix environments. When any of these files change, the IDS compare the new log entry with attack signatures to see if there is a match. If so, the system responds with administrator alerts and other calls to action.

Host-based IDS have grown to include other technologies. One popular method for detecting intrusions checks key system files and executables via checksums at regular intervals for unexpected changes. The timeliness of the response is in direct relation to the frequency of the polling interval. Finally, some products listen to port activity and alert administrators when specific ports are accessed. This type of detection brings an elementary level of network-based intrusion detection into the host-based environment.

Host-based IDS monitor user and file access activity, including file accesses, changes to file permissions, attempts to install new executables and/or attempts to access privileged services.

The project modules include Password attack, Scanning attack, Sniffing attack and Spoofing attack. The project is developed using C and Shell scripting in Linux environment.

ACKNOWLEDGEMENT

I wish to express my sincere thanks to **Dr. JOSEPH V. THANIKAL Ph.D.**, Principal, Kumaraguru College of Technology, Coimbatore, for permitting me to undertake this project.

My deepest acknowledgement to **Dr. M. GURURAJAN Ph.D.**, Head of the Department, Computer Applications, Kumaraguru College of Technology, Coimbatore, for his timely help and guidance throughout this project.

I also express my thanks to **Mr. A. MUTHUKUMAR M.Sc., M.C.A, M.Phil.**, Project Coordinator, Assistant professor, Department of Computer Applications, Kumaraguru College of Technology, Coimbatore, for encouraging me to do this project.

I am greatly indebted to my guide **Mrs. P. PARAMESWARI M.C.A, M.Phil.**, Senior Lecturer, Department of Computer Applications, Kumaraguru College of Technology, Coimbatore, for her valuable guidance and encouragement at every stage of this project work

I express my sincere thanks to **Mr. PARUDHI PRAKASH**, Project Leader, MAZE NET SOLUTION, COIMBATORE, for his support and assistance at various level of my project work.

Finally, I owe a lot to my beloved parents and family members and to my department staffs without their help and co-operation the project would not have taken a final shape.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	iii
1	INTRODUCTION	1
	1.1 Project Overview	1
	1.2 Organization profile	2
	1.3 System Configuration	4
	1.3.1 Hardware Configuration	4
	1.3.2 Software Configuration	4
	1.4 Programming Environment	4
2	THEME OF PROJECT	10
	2.1 System Analysis	10
	2.1.1 Literature Review	10
	2.1.2 Methodology	11
	2.1.3 System Review	14
	2.1.3.1 Existing System	14
	2.1.3.2 Proposed System	15
	2.2 System Design	16
	2.2.1 Input Design	16
	2.2.2 Output Design	17
	2.2.4 System Flow Diagram	18
	2.3 Module Development	24
	2.4 Testing and Implementation	25
3	CONCLUSION	29
	3.1 Conclusion	29
	3.2 Future Enhancements	30
	APPENDIXES	31
	REFERENCES	37

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

An Intrusion Detection System (or IDS) generally detects unwanted manipulations to systems. There are a lot of different types of IDS, some of them are described here. The manipulations may take the form of attacks by skilled malicious hackers, or Script kiddies using automated tools.

An IDS is required to detect all types of malicious network traffic and computer usage that can't be detected by a conventional firewall. This includes network attacks against vulnerable services, data driven attacks on applications, host based attacks such as privilege escalation, unauthorized logins and access to sensitive files, and malware (viruses, trojan horses, and worms).

Host-based intrusion-detection is the art of detecting malicious activity within a single computer. A host-based intrusion detection system (HIDS) uses host log information, system activity, and scanners such as virus scanners to determine whether a computer host is being used for illegitimate purposes. HIDS may be local to the protected host, remote (via syslogd, etc), or part of a distributed intrusion detection system.

A common technique is to make checksums of important system files that should not be altered under normal circumstances. Intruders are likely to replace system components with so-called root kits that enable them to remain hidden in the system while performing further probing such as sniffing.

A HIDS will monitor all or part of the dynamic behavior and of the state of a computer system. Much as a NIDS will dynamically inspect network packets, a HIDS might detect which program accesses what resources and assure that (say) a word-processor hasn't suddenly and inexplicably started modifying the system password-database. Similarly a HIDS might look at the state of a system, its stored information, whether in RAM, in the file-system, or elsewhere; and check that the contents of these appear as expected.

One can think of a HIDS as an agent that monitors whether anything/anyone - internal or external - has circumvented the security policy that the operating system tries to enforce.

The project aims to protect the system from following attacks :

- Scanning attack
- Password attack
- Sniffing attack
- Spoofing attack

1.2 ORGANIZATION PROFILE

Maze Net solution, based at Coimbatore is a private limited company involved in networking solution and training. Maze Net has partnered with Siemens Information system limited to offer technical solution and training.

Siemens is a leading multinational company with a wide expertise in areas like Network Administration, Database Administration ERP consultancy, System integration, Telecom and e-Business Solution.

SISL's strong commitment to training stems from its belief that professional training alone can provide the required skills and knowledge the equip both individuals and corporate to face and survive the competitive challenges of the times.

"Siemens IT Learning"- is a broad based, high-end IT Certification Training, through exclusive Partnership and Wave Technologies UK Ltd - A world leader in developing IT Learning Material and providing certification training and Prometric - The largest global Organization for administering vendor certification examinations in the area of IT Learning.

Siemens IT Learning is targeted towards the young, upwardly mobile executives With IT skills: either IT professionals or those from other functional areas. This initiative Also targets tomorrow's organizations that are sensitive to the need of a workforce that is skilled and requires continuous up gradation, as the only means towards achieving business excellence.

MazeNet also has partnered with Prometric Thomson U.S.A to conduct online vendor neutral certification and Wave technologies U.S.A for technical support in select few areas. MazeNet plays a very crucial role in providing networking solution in and around Coimbatore. It also helps in nurturing network engineers and network administrators.

1.3 SYSTEM CONFIGURATION

1.3.1 Hardware Configuration

Processor	:	Pentium III
Processor Speed	:	1.5 GHZ
Memory (RAM)	:	256MB
Hard Disk	:	20GB

1.3.2 Software Configuration

Operating System	:	Linux 9.0
Language	:	C & Shell Scripting
Kernel	:	2.4.20-8

1.4 PROGRAMMING ENVIRONMENT

The project entitled "INTRUSION DETECTION SYSTEM" is developed with the platform LINUX and the scripting language used is C and shell scripting.

1.4.1 About LINUX

LINUX was born of a humble beginning. This brings us to university of Finland student Linus Torvalds, circa 1991. Torvalds began working on an

operating system of his own. His first prototype, Version 0.01, was born in August of 1991. The first official version of LINUX, Version 0.02, was released on October 6th, 1991. Today, LINUX has evolved into a full-blown operating system that, in many ways, rivals commercial systems with the amount of interest it has garnered. LINUX has now become an actively developing and improving system. While many people do contribute to the development of LINUX, Kernel features are still controlled by Linus Torvalds.

LINUX Features

There are many reasons to use LINUX.

Free

LINUX is licensed under the free software foundation's GNU General Public License (GPL). According to the terms of the License, anything using it must make available the source code used to build the software, and anything using source code from the software must be licensed under the GPL. This perpetuates the availability of the source code.

Open source

This implies that the source code used to create the application is made available at no charge to the General Public. Users' configuration to the source code, which may be, in turn, merged into the distributed binary form of the application, are generally welcome. This is the way LINUX has come to thrive in.

Network

A network is a group of computers and devices connected together in order to communicate and work with each other.

LINUX was built with networking in mind and all distributions include the necessary programs and utilities to attached to and function on a network.

Power

LINUX has the ability to transform your legacy hardware into a powerful, well turned machine. Take that old 486 or Pentium and turn it into a fully multitasking crash free system.

Interactive

LINUX allows user to interact with the system, entering commands that are executed immediately (rather than the mainframe method of queuing commands to be run in a batch).

Multi-user

Linux allows multiple people to access the same computer at the same time, differentiate between them, and understand which process belong to whom.

Multitasking

Linux is capable of carrying out more than one task at the same time.

Multiprocessing System

Linux also supports multiple CPUs on the same computer. System with multiple processors performs faster than single CPU system because the

processors can combine forces to work on one task between multiple processors via a technique called multithreading.

Log files in Linux

Linux keeps detailed records of events within the system. Many programs create these records, known as log files. The system administrator can refer to the log files to determine the status of the system, watch for intruders, or look for data about a particular program or event.

The Purpose of Linux Log files:

On any Linux system, many events go on in the background as users log in and do their work. Daemons are special purpose background processors design to watch for network activity, run other programs, and monitor user actions. The status information collected by daemons is not displayed on the screen. Instead, it is written to log files, which we can then review.

Among other things, log files allow a system administrator to:

- Check for potential security problems, such as repeated login failures or a program that is stopped and restarted without the knowledge of the system administrator.
- Review what was happening on the system in the movements before a major problem occurred.
- Manage the system load by computer statistics based on the log file information.

1.4.2 About C

The programming language C was developed in 1972 by Dennis Ritchie at AT & T Bell Laboratory, Murray Hill, New Jersey.

C proved to be an excellent programming language for writing system programs. The UNIX operating system, C compiler and all UNIX applications software are written in C. The C compiler combines the capability of an assembly language with the features of a high level language, and therefore it is well suited for writing both system software and business packages.

Programs written in C are efficient and fast. This is due to its variety of data type and powerful operators. C is highly portable. This means that C programs written for one computer can be run on another with little or no modifications.

C language is well suited for structured programming, thus requiring the user to think of a problem in terms of function modules or blocks. A proper collection of these modules would make the computer programs. This modular structure makes program debugging, testing and maintenance easier.

1.4.3 About shell script

The shell is the command interpreter, or command-line environment, for Linux. A command interpreter is a program that accepts from the keyboard. Shell script uses that input to launch commands or otherwise control the computer system. There are hundreds of small utility programs designed to run from the command line.

A shell program is a sequence of one or more commands stored in a file. This file is executable, and can be called a shell procedure, the shell script. The shell program has some special built in commands that can be used in a shell script for iteration, conditional execution and defining local variables.

A particularly important feature of a Linux shell is that it gives users the ability to write scripts that the shell can execute. In general terms, a script is a text file that can be interpreted or executed by another program.

CHAPTER 2

THEME OF PROJECT

2.1 SYSTEM ANALYSIS

2.1.1 Literature Review

Intrusion detection involves determining that some entity, an intruder, has attempted to gain, or worse, has gained unauthorized access to the system. Casual observation shows that none of the automated detection approaches seek to identify an intruder before that intruder initiates interaction with the system. Of course, system administrators routinely take actions to prevent intrusion. These can include requiring passwords to be submitted before a user can gain any access to the system, fixing known vulnerabilities that an intruder might try to exploit in order to gain unauthorized access, blocking some or all network access, as well as restriction of physical access. Intrusion detection systems are used in addition to such preventative measures.

Some system errors may appear to the intrusion detection system to be intrusions. But, the detection of these errors increases the overall survivability of the system, so unintentional detection will be considered desirable and not precluded.

Intrusion detection systems are usually based on the premise that the operating system, as well as the intrusion detection software, continues to function for at least some period of time after an intrusion so that it can alert administrators and support subsequent remedial action.

Intruders are classified into two groups. External intruders do not have any authorized access to the system they attack. Internal intruders have some authority, but seek to gain additional ability to take action without legitimate authorization. Internal intruders may act either within or outside their bounds of authorization.

Host-based intrusion detection started in the early 1980s before networks were as prevalent, complex and interconnected as they are today. In this simpler environment, it was common practice to review audit logs for suspicious activity. Intrusions were sufficiently rare that after the fact analysis proved adequate to prevent future attacks.

FEATURES OF HOST BASED IDS

- Verifies success or failure of an attack
- Monitors specific system activities
- Detects attacks that network-based systems miss
- Detects attacks that network-based systems miss
- Near-real-time detection and response
- Lower cost of entry
- Requires no additional hardware



P-1905

2.1.2 Methodology

This project is developed under WATER FALL methodology, this methodology has four phases, and all the four phases falls under this project to develop the tool.

The phases are follows.

- Requirement Analysis
- Design
- Coding
- Testing and Implementation

System Scope:

The system group has changed with the responsibility to develop a new system to meet the requirements and design and development of a new information system. The source of these study facts is variety of users at all level through out organization.

Feasibility Assessment

In this stage problem was defined. Criteria for choosing solutions were developed, proposed, possible solutions, estimated costs and benefits of the system and recommended the course of action to be taken.

Requirements Analysis:

During requirement analysis high-level requirements like the capabilities of the system must provide in order to solve a problem. Function requirements, performance requirements for the hardware specified during the initial planning were elaborated and made more specific in order to characterize features and the proposed system will incorporate.

External Design:

External design of any software development involves conceiving, planning out and specifying the externally observable characteristic of the software product. These characteristics include user displays and report formats external data source and data sinks and the functional characteristics.

Internal Design Architectural and Detailed Design:

Internal design involved conceiving, planning out and specifying the internal structure and processing details in order to record the design decisions and to be able to indicate why certain alternations were chosen in preference to others. This phase also includes elaboration of the test plans and provides blue prints of implementation, testing and maintenance activities. The product of internal design is architectural structure specification.

The work products of internal design are architectural structure specification, the details of algorithm and data structure and test plan. In architectural design the conceptual view is refined.

Detailed Design:

Detailed design involved specifying the algorithmic details concern data representations interconnections among data structures and packaging of the software product. This phase emphasizes more on semantic issues and less synthetic details.

Coding:

This phase involves actual programming, ie, transacting detailed design into source code using appropriate programming language.

Debugging:

This stage was related with removing errors from programs and making them completely error free.

2.1.3 System Review

2.1.3.1 Existing System

The majority of commercial IDS are network based. These IDSs detect attacks by capturing and analyzing network packets. Listening on a network segment or switch, one network based IDS can monitor the network traffic affecting multiple hosts that are connected to the network segment, thereby protecting those hosts. Network based IDSs often consists of a set of single purpose sensors or hosts placed at various points in a network. As the sensors are limited to running the IDS, they can be more easily secured against attack.

2.1.3.2 Drawbacks of Existing System

- Network based IDS's may have difficulty processing all packets in a large or busy network and, therefore may fail to recognize an attack launched during period of high traffic. The need to analyze packets quickly also forces vendors to both detect fewer attacks and also detect attacks with as

little computing resource as possible which can reduce detection effectiveness.

- Network based IDS's cannot analyze encrypted information. This problem is increasing as more organization are virtual private networks.
- Most network based IDSs cannot tell whether or not in attack was successful, they can only discern that an attack was initiated. This means that after network-based IDS detects an attack, administrators must manually investigate each attacked host to determine whether it was indeed penetrated.

2.1.3.2 Proposed System

Host based IDS operate on information collected from within an individual computer system. This vantage point allows host based IDS's to analyze activities with great reliability and precision, determining exactly which processes and users are involved in a particular attack on the operating system. Furthermore, unlike network based IDSs, host based IDSs can see the outcome of an attempted attack, as they can directly access and monitor the data files and system processes usually targeted by attacks.

Advantages of Proposed System

- Host based IDS are easier to manage, as information must be configured and managed for every host monitored.

- Host based IDSs, with their ability to monitor events local to a host, can detect attacks that cannot be seen by network based IDS.
- Host based IDSs can often operate in an environment in which network traffic is encrypted, when the host-based information sources are generated before data is encrypted and/or after the data is decrypted at the destination host.
- Host based IDSs are unaffected by switched network.

2.2 SYSTEM DESIGN

2.2.1 Input Design

The input design is the process of converting the description of the input to the computer understandable format. The input design goal is to enter the data into the computer as accurate as possible. The input screens are designed in such a way that has simple intuitive and user-friendly layouts. As for as this project is concerned, the inputs from the users are only the command to monitor the specific events. All other inputs for monitoring suspicious activity are taken automatically by the system from the log files.

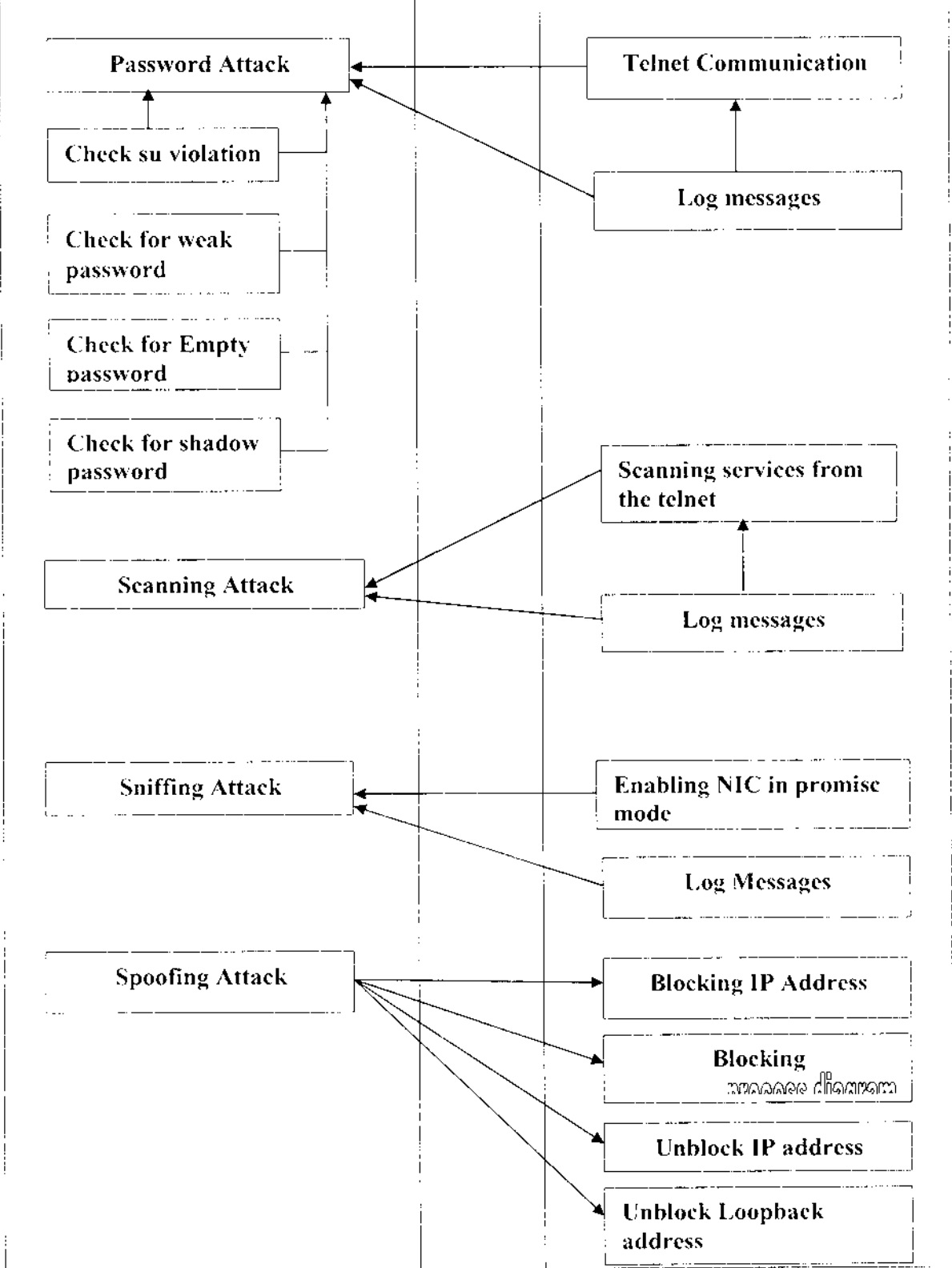
The important log files that feeds input to the log files are as follows :

- `etc/syslog/conf`
- `var/log/messages`
- `etc/shadow`
- `/etc/profile`

2.2.2 Output Design

Output design generally refers to the results and information that are generated by the system. The end-users evaluate the usefulness of the application.

As for as this project is concerned, the output is not to produce a permanent copy of the results for later reference. Rather it is the automatic activities of the system such as automatic lock of user account, automatic lock of services and automatic adjustments of the system to prevent themselves against intrusion.



2.2.5 Data Flow Diagram

Level 0 DFD :

FIG 2.2.5.1 Password Attack

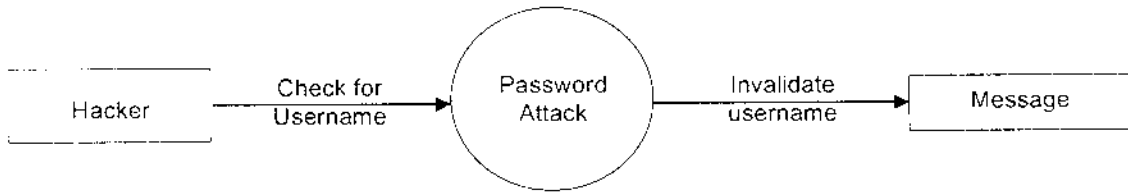
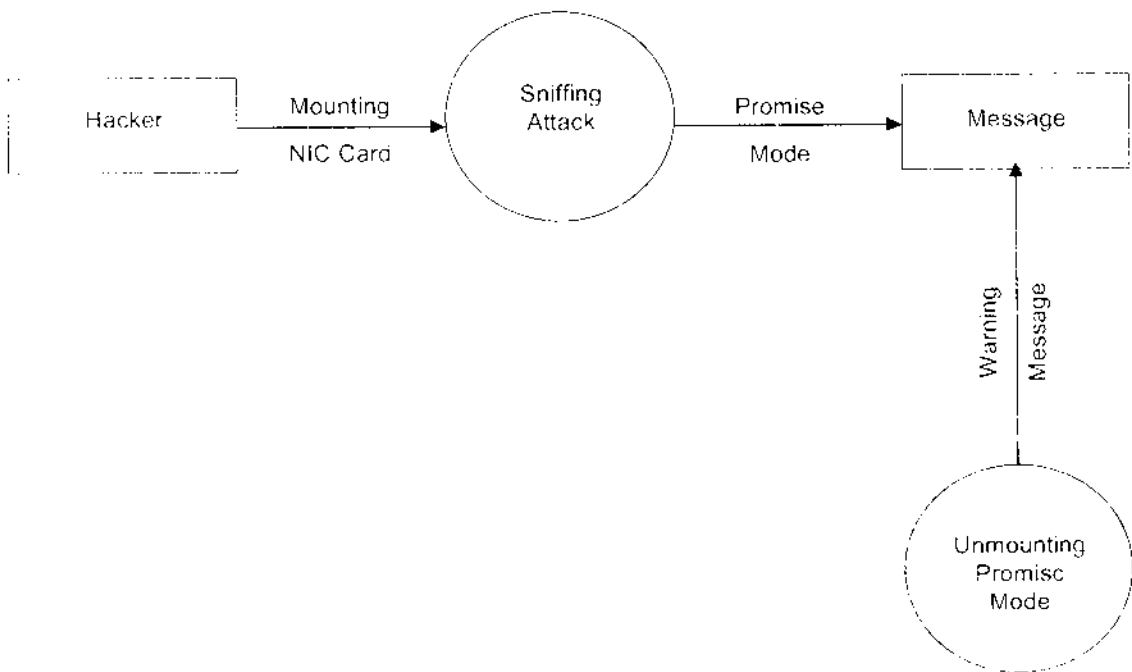


FIG 2.2.5.2 Scanning Attack



FIG 2.2.5.3 Sniffing Attack



Level 1 DFD :

FIG 2.2.5.4 Password Attack

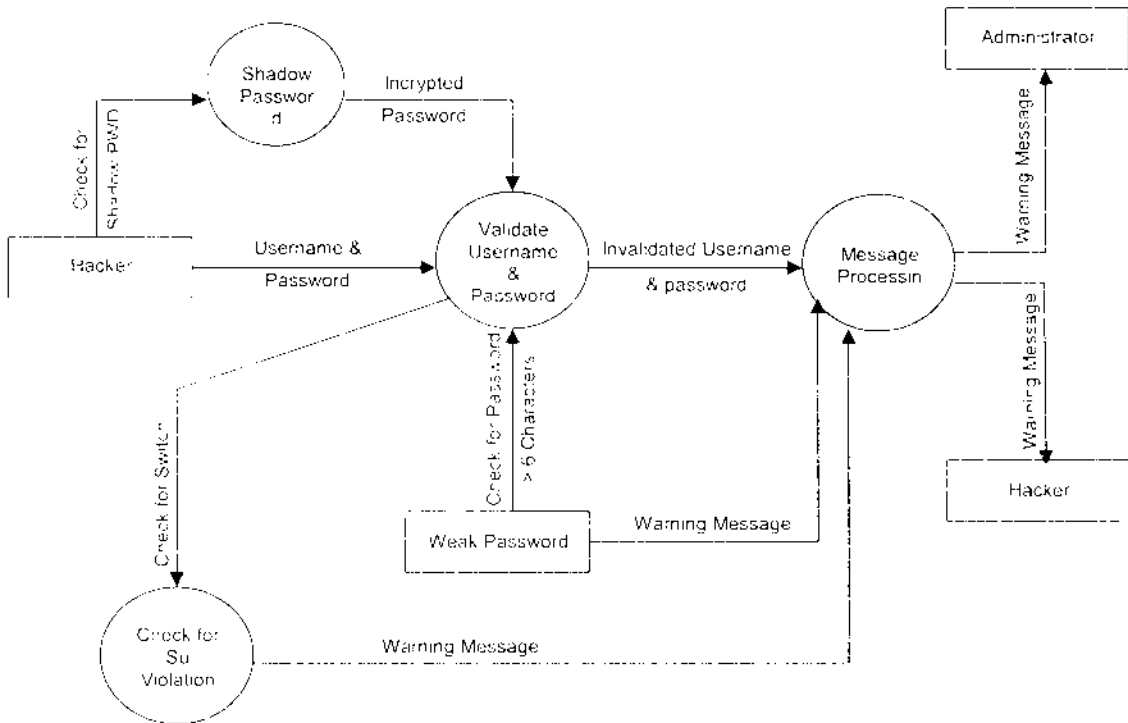


FIG 2.2.5.5 Scanning Attack

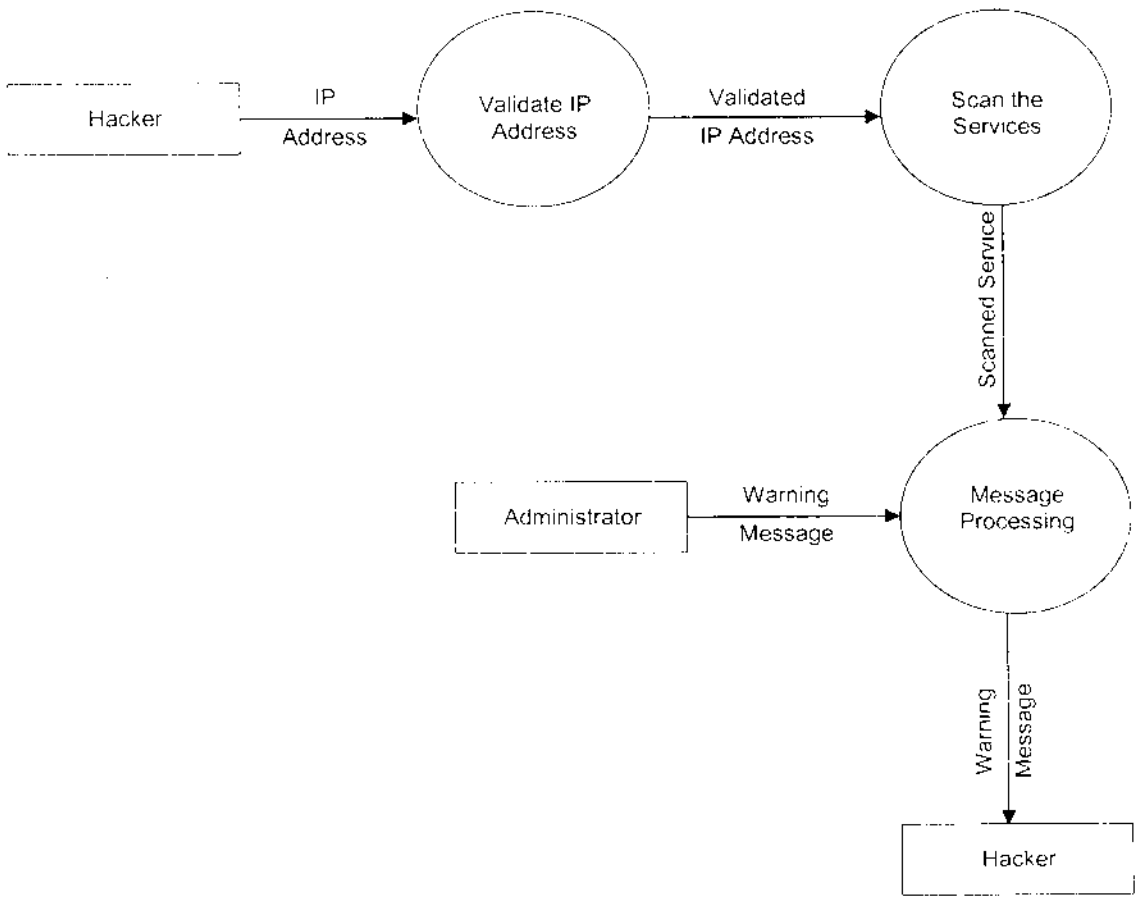
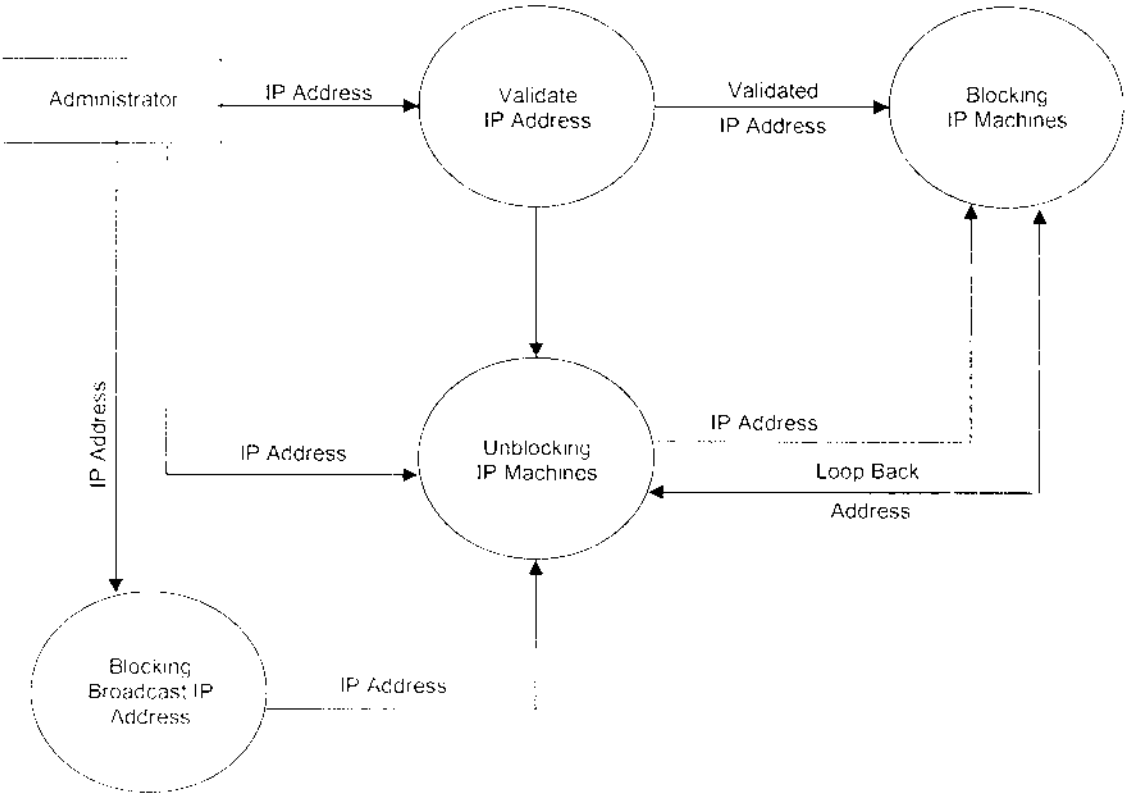


FIG 2.2.5.6 Sniffing Attack



2.3 MODULE DEVELOPMENT

The core modules of this system are as follows:

1. Scanning attacks
2. Password Attack
3. Sniffing Attack
4. Spoofing Attack

1. Scanning attacks

A Scanning attack occurs when an attacker probes a target network or system by sending different kinds of packets. Using the responses received from the target, the attacker can learn many of the system's characteristics and vulnerabilities. Some of them are :

- The topology of a target network.
- The active hosts on the network.
- The server software they are running.
- The software version numbers for all detected software.

2.Password Attack

Password attacks involves Authorized User attack and Public User attack. The authorized user attacks are those that start with a legitimate user account on the target system. Most authorized user attacks involve some sort of privilege escalation.

Public user attacks are those launched without any user account or privileged access to the target system. Public user attacks are launched remotely

through a network connection using only the public access granted by the target. One typical attack strategy calls for an attacker to use a public user attack to gain initial access to a system. Then once on the system, the attacker uses authorized user attacks to take complete control of the target.

3. Sniffing Attack

Sniffing means reading the contents of the Network Interface card in the promisc mode of the Linux environment. This attack is done in the promisc mode only. If any hackers tried to view the contents of the Network Interface Card the software will identify the hackers by giving a warning message to the hackers. Our software identifies the hacker and gives a warning message when the hacker tries to enter the promisc mode and all the hacker details will be appended into the recorded file present in the software.

4. Spoofing Attack

Spoofing means the Internet packet with the local address. When the hackers tried to hack the Internet packet that has the local address the software will identify the hackers by giving a warning message to the hackers and also the hacker details will be appended into the recorded file in the software.

2.4 TESTING AND IMPLEMENTATION

Testing is a set of activities that can be planned in advance and conducted systematically. After finishing the development of any computer based system the next complicated time consuming process is system testing. During the time of testing only the development company can know that, how far the user requirements have been met out, and so on.

Testing Methodologies

Following are the some of the testing methods applied to this effective project:

Unit Testing

Individual components are tested to ensure that they operate correctly. Each component is tested independently, without other system components. With respect to this project, the individual functions are treated as component and were tested.

Validation Testing

This is the important factor in input design. The input data is the main source of the system, so proper validation for input data is needed in both field

level and form level; these are accomplished in our software by including appropriate validation procedures. Data validation ensures that every value that the user enters into the application is accurate. Two data validation methods done in this project are

- Page Level Validation takes place after a user has filled in all fields in the form.
- Dynamic Level Validation takes place as each field on a form is filled in.
- The function or performance characteristics confirming to the specifications are accepted.
- Deviation from the specification is found and a deficiency list is created.

Output Testing

After performing the validation testing, the next step is output testing of the proposed system since no system would be termed as useful until it does produce the required output in the specified format.

Implementation

System implementation is the process of bringing the developed system into operational use and turning it over to the user. This is the final stage of the project where the theoretical design is turned into working design.

The System implementation phase consists of the following steps:

- Testing the developed software with sample data
- Creating files of the system with actual data
- Errors can be identified
- Making necessary changes to rectify errors

Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and its constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

Implementation is the process of converting a new system design into operation. It is the phase that focuses on user training, site preparation and file conversion for installing a candidate system. The important factor that should be considered here is that the conversion should not disrupt the functioning of the organization.

CHAPTER 3

CONCLUSION

3.1 CONCLUSION

This project, IDS, prevents the hacking throughout the network and maintains the traffic efficiently among network servers. This vendor-neutral guide to the concepts and terminology of hacking offers practical guidance to planning and implementing the technology in most environments. This benefit includes extreme availability, redundancy/resiliency/replication, and security throughout the network and prevents the hacking

- A lot of details are required to build good IDS.
- Difficult to build perfect IDS.
- IDS are an incremental work - possibly never ending.
- Simple tools can sometimes detect complicated attacks and vice versa.
- Protection seems more difficult than intrusion

3.2 FURTHER ENHANCEMENT

- Network- and host-based intrusion detection can be integrated into a single system.
- Shared management console with a consistent interface for product configuration, policy management and single-event display for notifications from both host and network components.
- Integrated event database
- Integrated on-line help for incident response.
- Unified and consistent installation procedures.

APPENDIXES

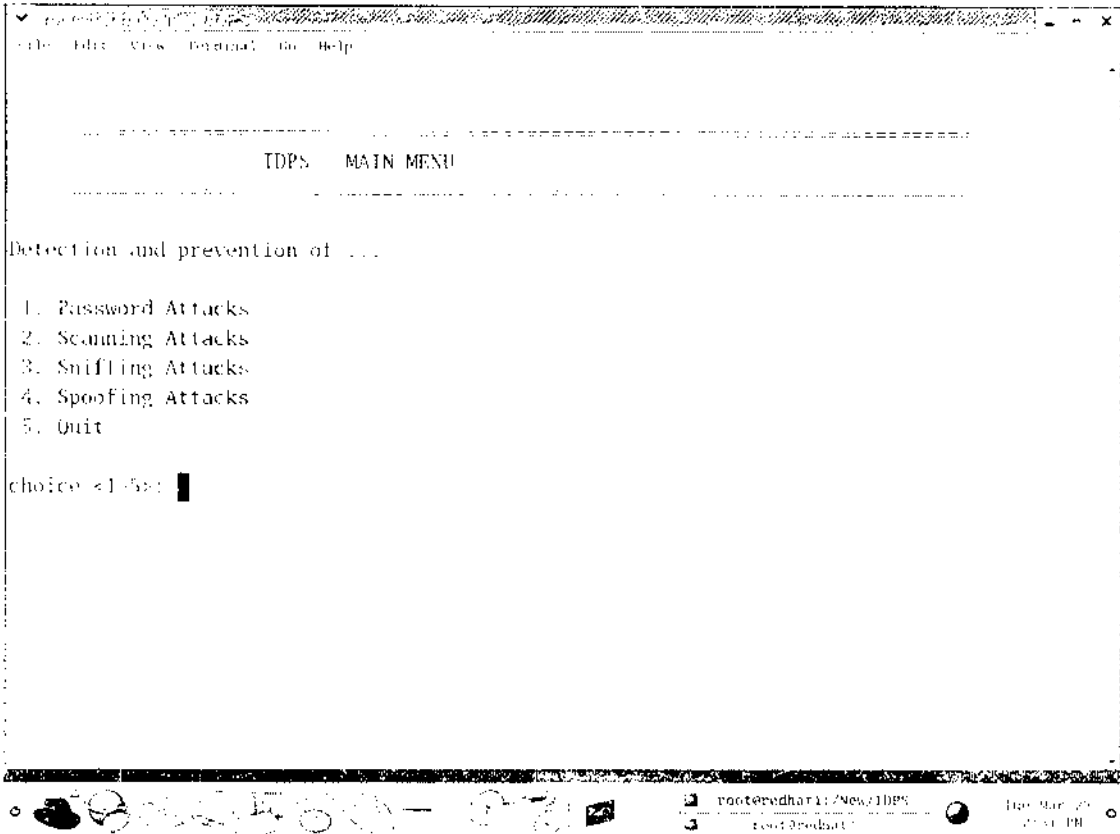


FIG A.11 MAIN MENU

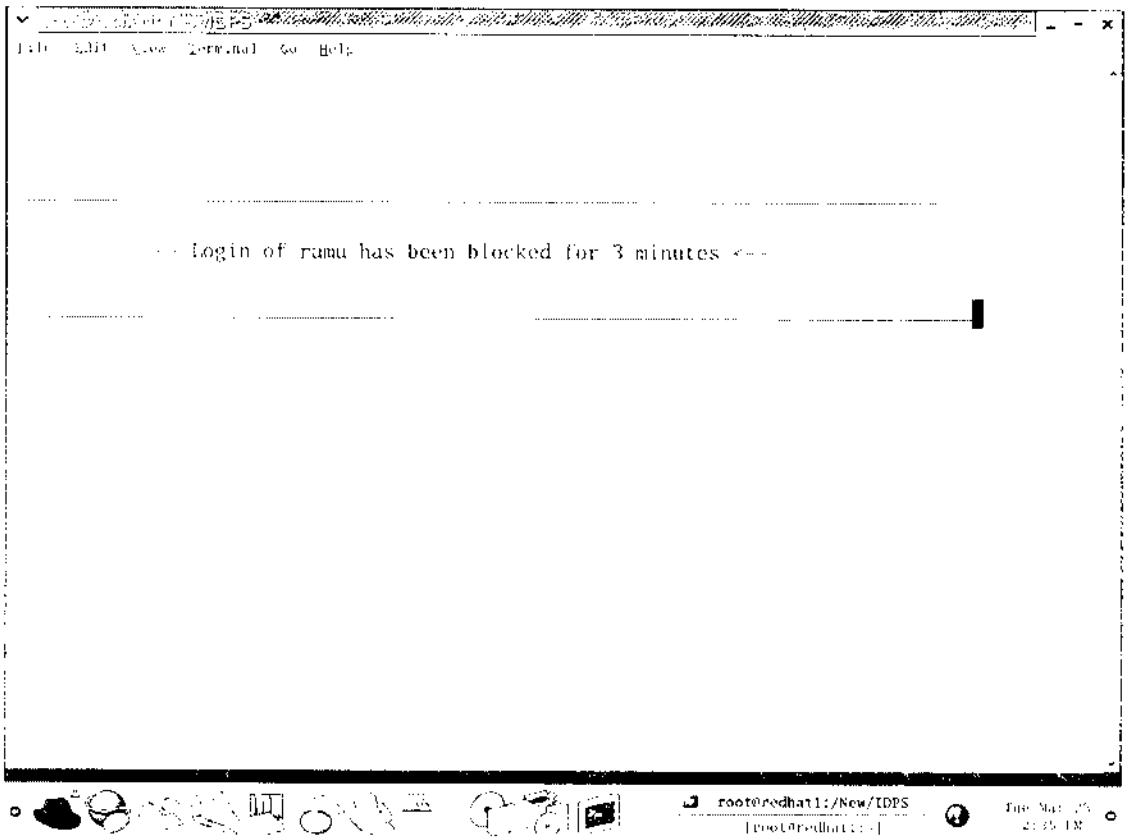


FIG A.1.2 LOGIN BLOCKING MESSAGE SCREEN

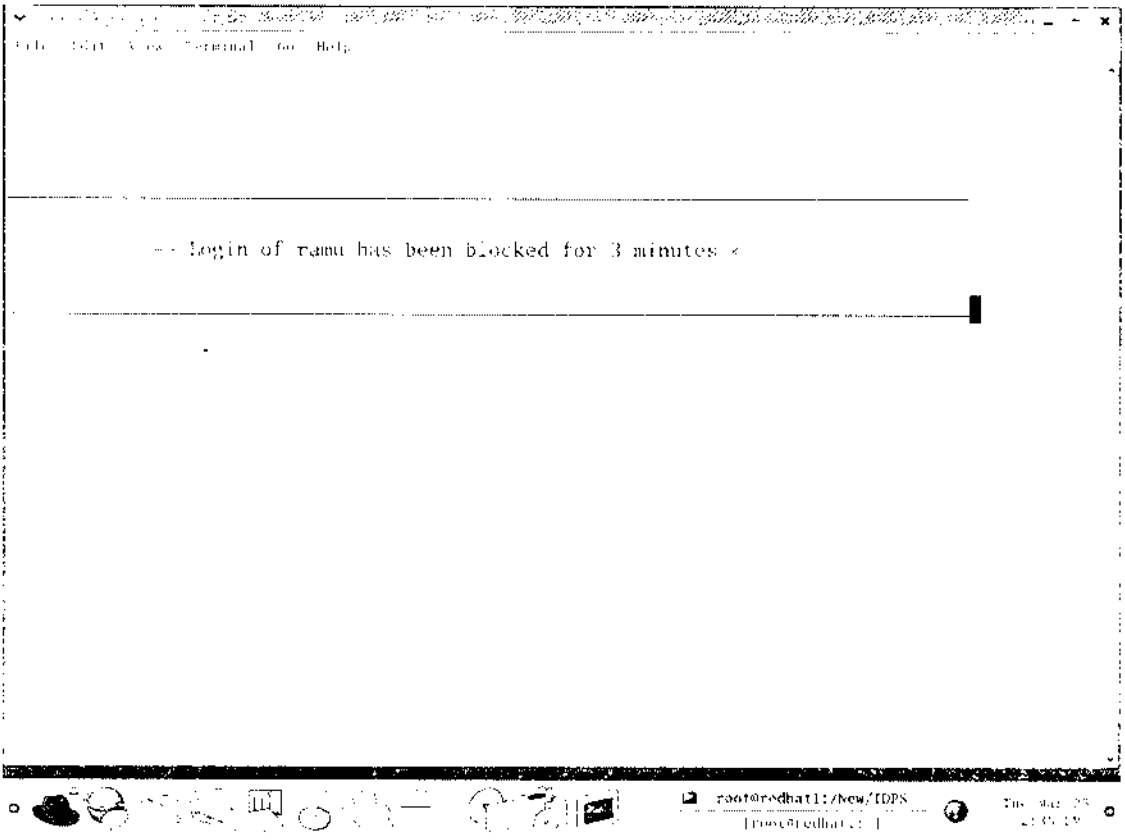


FIG A.13 MAIN MENU


```
File Edit View Terminal Go Help
Chain INPUT (policy ACCEPT)
target    prot opt source                destination
ACCEPT    all  --  anywhere              anywhere
DROP      all  --  anywhere              anywhere
ACCEPT    all  --  anywhere              anywhere
DROP      all  --  anywhere              anywhere
ACCEPT    all  --  anywhere              anywhere
DROP      all  --  anywhere              anywhere

Chain FORWARD (policy ACCEPT)
target    prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target    prot opt source                destination
ACCEPT    all  --  anywhere              anywhere
DROP      all  --  localhost            anywhere
ACCEPT    all  --  anywhere              anywhere
DROP      all  --  localhost            anywhere
ACCEPT    all  --  anywhere              anywhere
DROP      all  --  localhost            anywhere

Do you want to block any IP address ...[y/n]:y
Enter the IP Address to be blocked ....: 192.168.200.15
```

FIG A.1.4 IP ADDRESS BLOCKING SCREEN

```

Chain INPUT (policy ACCEPT)
target     prot opt source                destination
ACCEPT    all  --  anywhere              anywhere
DROP      all  --  anywhere              anywhere
ACCEPT    all  --  anywhere              anywhere
DROP      all  --  anywhere              anywhere
ACCEPT    all  --  anywhere              anywhere
DROP      all  --  anywhere              anywhere
DROP      all  --  192.168.200.15      anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
ACCEPT    all  --  anywhere              anywhere
DROP      all  --  localhost            anywhere
ACCEPT    all  --  anywhere              anywhere
DROP      all  --  localhost            anywhere
ACCEPT    all  --  anywhere              anywhere
DROP      all  --  localhost            anywhere
DROP      all  --  192.168.200.15      anywhere

The specified IP address will be blocked in the future
Do you want to continue [y/n]: █

```

FIG A.15 BLOCKING CONFIRMATION SCREEN

REFERENCES

BOOK REFERENCES

1. Linux in easy step by David Nash a Dream Tech Publications.
2. Linux Application Development by Michael K.Johnson and Erik W.Troan, a Red Hat Publications.
3. Shell Programming” written by Yashwan Kanitkar, Tata McGraw-Hill, Publishing Company Limited, New Delhi.
4. An Intrusion Detection Model.” by D.Denning, Second Edition, Pearson Education Asia, ISBN: 981-4035-20-3.

WEB REFERENCES

1. <http://www.linuxforu.com>
2. <http://www.freeref.com/books/security/ids>