Implementation of a Prototype to Detect Unusual Behaviors on a Network

GRADUATE PROJECT REPORT

Submitted to the Faculty of
The School of Engineering & Computing Sciences
Texas A&M University-Corpus Christi
Corpus Christi, TX

In Partial Fulfillment of the Requirements for the Degree of
Master of Science in Computer Science

By

Deepthi Jambula
Fall 2012

Committee Members

Dr. Mario Garcia
Committee Chairperson

Dr. Dulal Kar
Committee Member

Dr. John Fernandez
Committee Member
Security is very crucial in any computing environment such as web software applications, Local Area Networks, etc. because Internet communications are subject to attacks. Lack of adequate protection can cause web security problems. Detecting unusual behaviors is the first step towards predicting the web security issues. This can be achieved by monitoring network web activities, network bandwidths usage, individual user/client session activity and blocking web intrusions. Network monitoring is a crucial task for any organization. These tools have to capture all of the client’s web activities with login time, logout time, and real network bandwidth usage. This project traces the client activity, captures bandwidth, tracks network intrusions, generates denial of service, and blocks any blacklisted websites preconfigured in the server. Along with these features the preconfigured exceedances will be alerted to the server. The project shows a dashboard style network monitoring suite with individual buttons to reset and to extract the reports regarding the above mentioned activities.

Objective of this project is to monitor Local Area Network (LAN) traffic, track client’s activity on LAN, alert network intrusions, track bandwidth usage and block blacklisted websites. In order to accomplish this, LAN server is configured with a proxy server on port 8080 to monitor the network traffic, here the packet information from the clients internet usage is captured using jNetPcap and is published in the server database. The application monitors the database and generates notifications to the server about the client’s unusual behavior.
# TABLE OF CONTENTS

Abstract ........................................................................................................................................ ii

Table of Contents ....................................................................................................................... iv

List of Figures ............................................................................................................................ vi

1. Introduction and Background .................................................................................................1

1.1. Introduction .......................................................................................................................1

1.1.1. JNETPCAP ..................................................................................................................2

1.1.2. Network Monitoring .................................................................................................2

1.1.3. Network Bandwidth Monitoring ..............................................................................2

1.1.4. Proxy Server ..............................................................................................................3

1.1.5. User ID/Password Authentication .............................................................................3

1.1.6. Session Activity .........................................................................................................4

1.1.7. OSI Layers Model ......................................................................................................4

1.1.8. Ports ..........................................................................................................................5

1.1.9. Port Blocker ...............................................................................................................6

1.2. Background ......................................................................................................................6

2. Narrative ..............................................................................................................................10

2.1. Problem Statement .........................................................................................................10

2.2. Scope ..............................................................................................................................10

2.3. Motivation ......................................................................................................................11

2.4. Existing System .............................................................................................................12

2.5. Proposed System ...........................................................................................................12
2.5.1. Server side ................................................................. 14
2.5.2. Client side ................................................................. 15
2.6. System Architecture ...................................................... 16
2.7. Proposed System Design ................................................. 17
3. Functionality of Application ............................................... 18
  3.1. Module Description ....................................................... 18
    3.1.1. Monitoring System ................................................. 19
      3.1.1.1. Network Traffic Routing .................................. 19
      3.1.1.2. Network Monitoring ....................................... 19
      3.1.1.3. Network Bandwidth Monitoring.............................. 19
    3.1.2. Alert Messaging .................................................. 20
    3.1.3. Database Connectivity ........................................... 20
    3.1.4. Blocked Browser .................................................. 20
    3.1.5. Port Blocker ....................................................... 21
  3.2. Object Oriented Analysis and Design (UML) ....................... 21
4. Evaluation and Results .................................................. 26
  4.1. Test case 1 ............................................................. 26
  4.2. Test case 2 ............................................................. 26
  4.3. Test case 3 ............................................................. 27
  4.4. Test case 4 ............................................................. 28
  4.5. Test case 5 ............................................................. 28
  4.6. Test case 6 ............................................................. 29
  4.7. Test case 7 ............................................................. 30
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.8. Test case 8</td>
<td>31</td>
</tr>
<tr>
<td>4.9. Test case 9</td>
<td>32</td>
</tr>
<tr>
<td>4.10. Test case 10</td>
<td>32</td>
</tr>
<tr>
<td>4.11. Test case 11</td>
<td>33</td>
</tr>
<tr>
<td>4.12. Advantages &amp; Disadvantages</td>
<td>34</td>
</tr>
<tr>
<td>5. Conclusion and Future Work</td>
<td>36</td>
</tr>
<tr>
<td>Bibliography and References</td>
<td>38</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

Figure 1: Intrusion Detection System in Network ...........................................9
Figure 2: Proposed System Design .................................................................16
Figure 3: Flowchart of System .......................................................................17
Figure 4: Use-case diagram for administrator ..................................................22
Figure 5: Use-case diagram for client ...............................................................23
Figure 6: Sequence diagram for client .............................................................24
Figure 7: Sequence diagram for administrator ..................................................25
Figure 8: Server Interface ...............................................................................26
Figure 9: Client Interface .................................................................................27
Figure 10: Client Console ...............................................................................27
Figure 11: Server Console ...............................................................................28
Figure 12: Database showing entire web activity ..............................................29
Figure 13: Database showing entire login and logout time of client .................30
Figure 14: System usage of client recorded in database ....................................31
Figure 15: Alert message to server if Bandwidth exceeds ................................32
Figure 16: Browser gets Blocked when client accesses blacklisted website ....32
Figure 17: Client Login Session ...................................................................33
Figure 18: Alert message for blocked ports ....................................................34
1 Introduction and Background

1.1 Introduction

Due to increasing information flow in the Internet, and the need for rapidly and reliable accessing this information by organisations in a timely fashion without much overhead, an optimised network is required. Due to these requirements, the usage of the servers, Local Area Networks, and data storage devices have also increased rapidly in recent years. This rapid growth in technology has changed how organizations and businesses are run in this modern era. In developed countries, paper handling is reduced and usage of computers and their networks are rapidly increasing. Along with this rapid growth in networking and the Internet, threats are also increasing. In order to optimize the network, several network features such as bandwidth, security, firewalls and different communication ports have to be constantly monitored and controlled by the network administrator or an automated server. In most cases, there are several automated monitoring systems which support overall performance of a single unusual behaviour on the LAN. This project focuses on the three unusual behaviours of a client:

1. Excess bandwidth usage in the network
2. User-ID and password mismatch on a synced system
3. Blocking the blacklisted website access by client

And other features are of this project are:

4. Recording client’s session activity using JNETPCAP to capture and log packets in a server database under that particular client
5. Network intrusion detection by monitoring the configured ports
1.1.1 JNETPCAP

Packet capture is accomplished using an open source JNETPCAP library containing a wrapper [6]. This Java function is used to capture and decode all the packets in real time inside the network layer of the OSI model. This function has the capability to capture all the core network protocol packets and also gives the flexibility to add a user defined protocol which is also supported by all Windows extensions.

1.1.2 Network Monitoring

Network monitoring is a tool which gives an overview of the servers, systems, and other devices interconnected to one another with a detailed usage display to the total capacity of each machine [11]. These tools are also called system/network diagnostic tools which give the information about the system performance and availability. These are comprehensive tools which give the information about the overall system performance but fail to track the client’s unusual behaviours such as bandwidth usage, security intrusions, restricted session activity, and other unusual behaviours.

1.1.3 Network Bandwidth Monitoring

Bandwidth monitoring is a tool which can be used to calculate the bandwidth usage of each client [4]. The majority of the bandwidth calculating tools work on capturing the frames at the data link layer in the seven layered OSI model. The data link layer converts the traffic packets to frames requested by the individual clients. The packets are generated by the source/destination of the inbound/outbound traffic through the routers. When the frames exceed a threshold value, then the system
samples as a single exceedances and generates the computed value on the server. The majority of these bandwidth monitoring tools are computed or calculated values.

1.1.4 Proxy Server

The proxy server is a server where the client’s requests are evaluated and passed to the appropriate destination. The IP address and the communication ports are configured in this proxy server inside the LAN setting [14]. Because of these settings, the traffic flowing through this port is trapped in this proxy server. There are different types of proxies in the market such as forward proxy, open proxy, and reverse proxy. Forward proxy setup consists of the server inside the LAN which tracks the traffic flow due to the internal client requests to the outside servers. Open proxies are proxies outside the computer network and also outside the destination servers. These proxies track and trap this traffic flow generated by either of the internal or external requests. The third type of proxy is the proxy within the organisation computer network where the traffic flow from the external servers to the client or the recipients is tracked. These servers are typically used to control the communication port. The settings can be accessed by opening the browser’s connections tab under the Internet options. By checking in the proxy server here, the administrator can configure the required port for tracking.

1.1.5 User ID/ Password Authentication

User-ID/ Password authentication is typically used to control access to a particular account, computer, server, etc. Here the designated clients are configured with a unique user-id and password with a possibility of alpha-numerical format. The
functionality in some applications extends further with a minimum requirement and an expiration rule. These automated systems will control intrusions preventing unauthorized access to a particular account.

1.1.6 Session Activity

Session activity of a particular client or a user is his/her log-in time, log-out time, and the web activity. Client log-in time is registered in the server when the client logs-in to the computer using his/her user-ID and password. Similarly, log-out time is the exit time from his/her computer. The websites visited during this time period between the log-in and log-out time is called web-activity.

1.1.7 OSI Layers model

The OSI model is comprised of seven different layers required to make the data flow between the two computers meaningful. Here each layer has its own unique functionality. These seven layers are physical layer, data link layer, network layer, transport layer, session layer, presentation layer and application layer [1]. The physical layer gives the basic system information about the hardware requirements such as power, speed, pins, and other system specifications. The data link layer corrects any data errors in the physical layer and presents them to the network link layer in a presentable format. The network layer puts this data in a particular sequence and separates the packets based on source and destination address. The transport layer transfers the data from the network layers to the upper layers and corrects/keeps track of the data flow. Any missed data will be retransmitted to the upper layers. The session layer controls the session and information queue for local and remote clients. The
presentation layer syntax semantics correlation is done at a higher level in order to provide the application layer with a user understandable format. Finally, the application layer presents the information in an understandable format where the client can interact with the software applications. The network layer is the main focus of this project where the packet information is decoded and sent to the individual recipient.

1.1.8 Ports

A port is an interface between the computers and other peripheral devices. Each computer has more than 65,000 ports [7]. The first 1,024 ports are blocked ports which are predefined for certain software applications and communication protocols. Every protocol has its predefined ports used for communication between the client and the external servers. For example, port 80 is used for HTTP access for the client and the Internet. Similarly, port 8080 is used for configuring a proxy server, where the complete LAN traffic will pass through this particular port. In recent years, several idle ports or open ports have been attacked by outside users/viruses causing huge down time of the networks and loss of valuable information. There are several different types of ports such as software ports and hardware ports. Mainly, this project focuses on software ports which are invisible to the client and are prone to attacks. There are several software ports such as TCP/IP, UDP, FTP, HTTP, etc. Every port has its own packet format, length, and structure.
1.1.9 Port Blocker

A port blocker is software used to block or deny service by filtering the data content or by the reputation of the source and destination transaction [2].

1.2 Background

Several researchers have worked on these parameters at different levels. Ubik, Anoniades and Osbelo [13] used a packet capture method to monitor the traffic going through the router connected to the organization server. Their method works by continuously monitoring the inbound and outbound traffic through a router without a predefined bandwidth allocation to a particular host or client. They also proposed the bandwidth allocation on an OSI model. This method helps in identifying the overall bandwidth utilization in the seven layered architecture without the need for information about the data flow requirements.

Basu, Cheng and Yinzhe [3] show that the bandwidth allocation for a multimedia service is required in order to provide quality of service (QoS). Here the network actively monitors the available bandwidth and thereby controls the speed of the packet delivery. If the bandwidth is small, the delivery of the packet will be slow and vice versa. Since this involves multimedia data transfers, the data has to be centered from multiple servers. This method is based on the statistical confidence interval of different bandwidths between the client and the distributed servers. Here the distribution servers may be connected to the client on different ports. The primary advantage of this method is to monitor the bandwidths on different com ports and effectively deliver the packet without a noticeable lag time. This algorithm attains an optimal bandwidth level without much overhead computation. In order to achieve this, the algorithm puts the packets from
different multimedia servers in one matrix. A threshold value is determined for that particular quality of service (QoS). In order to deliver the quality of service, the packet has to be delivered to the destination without losing the packets. This can be a complex task since each multimedia packet is in a different format and development of a common threshold value for different packet formats required intense complex computation.

Recent developments in the network optimization field have motivated many researchers to detect unusual client behavior on the network by real monitoring of the communication port. Gang, Hui [15] did extensive work on the real time traffic analysis of the network. They used cross queuing methodology where in the packet length and the repeated increase in the request for the packets above the threshold have been identified and implemented on the real time network. This method not only identifies the real time bandwidth allocations, but also identifies the network security intrusions.

Some other researchers have developed adaptive tools for monitoring the bandwidth used on the communication port [16]. Here the author used methods to identify the maximum daily usage of the bandwidth and, if it is below the threshold, then no change is applied to the bandwidth selection. If the daily maximum bandwidth increases beyond the threshold value, then the method requests for a predefined step increase in the bandwidth. This method helps to a certain extent to have a high speed response even if the usage increases, but due to the limitations of the available bandwidth provided by suppliers the increase in bandwidth will result in and exponential increase in cost.

Several recent studies show that the researchers have developed complex algorithms to monitor bandwidth usage in different aspects by computing values from
inferred data of packet to frame conversion in the data-link layer. In this project, instead of computing a value, it measures the bandwidth usage directly by analysing the inbound and outbound packets as well as by monitoring the data transfer in the network layer. This achieves improved web server performance without much overhead.

Shiyong, Chengrong and Guo [17] focused on bandwidth optimization on a multi-server, multi-gateway ring type internet connection, here the source and destination gateway bandwidths are constantly monitored for sending or receiving the data when there are minimum packet queues at either end. The destination gate calculates the available bandwidth for a new transaction while keeping the existing data exchange flow running in the queue and subtracts this used capacity from the total bandwidth capacity. This information is communicated to the source for releasing the data as soon as the required bandwidth is available to receive the data at the destination gateway.

Tian, Meijuan and Zhang [8] proposed a model to train the network to detect the network intrusions based on the least square method between the nearest network neighbours. By using this method networks can adapt to future intrusions by recording the attacked profiles between the nodes of a computer network.

Kuai Xu, Zhi-Li Zhang [10] proposed techniques to identify the traffic dynamics caused by cyber-attacks and different application software’s deployed for creating these dynamics. They have created a methodology to understand the traffic dynamics caused by various scenarios using data mining and entropy principles. They used a cluster of points from the traffic flow which is separated based on their characteristics and behaviour; and later presented in a understandable format for the client to differentiate
and capture the intrusions and thereby training the network to adapt and prevent future intrusions.

Tseng and Chen [5] studied the email scams extensively, they have identified that the content based filtering and reputation based filtering are insufficient to the modern world for detection of spam in the email and social networks. They have developed a model to extract and train the spam features to the email networks. Their work was based on support vector machine model (SVM), where in the system is trained to catch the changing nature of the communication by incremental scheme.

Tan and Collie [12] worked on detecting network intrusions by comparing the signatures of the existing software applications with the new applications. This way, the system can detect the network intrusions after the attack has already taken place. This methodology will easily identify any remote terminal applications running on the client server and any modifications to the network services.

Komviriayut, Sangkatsanee, Wattanapongsakorn and Charmsripinyo [9] proposed decision tree and rule based approaches for detecting network intrusions. Like shown in figure [1], these authors presented that even with the existence of the firewalls, the systems cannot be protected completely due to the access of certain open ports. These ports have to generate denial of service based on the set of decision tree rules.
Figure 1: Intrusion detection system in network environment [9]
2 Narrative

2.1 Problem Statement

Detecting client’s unusual behavior is a challenging task using network monitoring tools. The existing network monitoring tools give an overview of the servers, systems, and other devices interconnected to one another with a detailed display of percentage of the data content each machine is holding when compared to the total capacity of each machine. These tools are also called system/network diagnostic tools which give information about the system performance and availability. These comprehensive tools fail to track the client’s unusual behaviours such as bandwidth usage, security intrusions, restricted session activity, and other unusual behaviours.

2.2 Scope

The scope of this project is as follows:

1) Monitor client’s bandwidth usage in the network and alert the server if bandwidth usage exceeds the threshold value.

2) Monitor client’s unusual behaviours by intrusion detection and notification to server when an unauthorised third party login attempts to break a synced client and server.

3) Block the blacklisted websites and record client’s session activity.

4) Develop a reset button on the server side to reset client’s session activity, and develop a bandwidth retrieval button to extract client wise bandwidth usage report.
5) Alert the network of intrusions caused by external sources on configured ports.

2.3 Motivation

Due to the increasing information flow in the Internet, and the need for rapidly and reliably accessing this information by organisations in a timely fashion without much network overhead, an optimised network is required. Due to these requirements, the usage of the servers, local area networks, and data storage devices have also increased rapidly in recent years. This rapid growth in technology has changed how the organizations and businesses are run in this modern era. Along with this rapid growth in networking and Internet world, threats are also increasing. In order to optimize the network, several network features such as bandwidth, security, firewalls, and different communication ports have to be constantly monitored and controlled by the network administrator or an automated server.

Many tools are emerging in the market targeting unusual behaviours in the networks, but all these tools have to be used independently to detect a particular unusual behaviour. The idea of creating a new dashboard style prototype leads to the motivation to work towards developing this project. Combining all of these features in one single suit is a challenging task.

In most cases, there are several automated monitoring systems which give an overall performance of a single unusual behaviour in the LAN. This project focuses on the three unusual behaviours of a client such as:
1. Excess bandwidth usage in the network
2. User-ID and password mismatch on a synced system
3. Network intrusion detection by monitoring the configured ports
   
   And other features are of this project are:
4. Blocking the blacklisted website access by client
5. Record client’s session activity using JNETPCAP to capture and log packets in a server database under that particular client.

2.4 Existing System

Detecting a client’s unusual behavior is a challenging task in network monitoring tools. The existing network monitoring tools give an overview of the servers, systems, and other devices interconnected to one another with a detailed display of percentage of the data content each machine is holding when compared to the total capacity of each machine. These tools are also called system/network diagnostic tools which give information about the system performance and availability. These comprehensive tools fail to track the client unusual behaviours such as bandwidth usage, security intrusions, restricted session activity, and other unusual behaviours.

2.5 Proposed System

The first part of this project includes tracking the unusual behaviour caused by the bandwidth exceedances. Here client’s bandwidth usage is measured and a notification is generated to the server if the usage exceeds the client’s predefined threshold value configured in the server database. When a client exceeds predefined bandwidth configured to his/her account, the system will generate an alert to the server about the
excess usage of bandwidth by that particular client. The bandwidth between a particular
client and server is a measured value in the network layer, where the total number of
packets received and transmitted by a particular client through a predefined port are
captured and counted in the network layer. A predefined threshold value is set in the
database inside the server. This number can be altered only by the administrator. If the
client’s daily bandwidth usage exceeds this threshold value, then an alert will be
generated to the server conveying the excess usage of bandwidth by that client. The
threshold value can be manually changed with the increasing usage of the bandwidth.

The second part of this project includes tracking the unusual behaviour caused by
the network intrusion. Here two independent types of security intrusions are being
focused under the security intrusion detection part of this project. First, a client’s
computer access is controlled by a dedicated user ID and password for a dedicated synced
system. If another client tries to login to this synced computer, then the system will
generate an alert to the server or administrator about a possible intrusion attempt at that
particular client computer.

The final unusual behaviour tracked by this project includes blocking the
predefined blacklisted websites in the server and capturing the session activity. The
session activity includes client’s login/logout time and web activity. Here the web
activity is captured in the form of packets from the inbound and outbound traffic seen on
the communication port. This information is logged in a database and can be retrieved by
the administrator if required at a later date to help analyse and troubleshoot errors caused
during that particular session activity.
Also in this project, the server is configured with a reset button to reset independent client’s session activity logged under the client’s database. A bandwidth retrieval button configured in the server will retrieve the bandwidth used by an individual client.

The system generates an alert when a preconfigured port is attacked by an external source. This will allow organizations to protect the internal server data from external intrusions.

Therefore, this project empowers the administrator to track, control, and deploy three unusual client behaviours real time on any organization’s network. With this prototype/dashboard style suite, the administrator has the capability to extract client’s activity information from the database by clicking the dedicated button for a particular client. These features allow the administrator not only to track client’s unusual behaviours, but also to optimize the port traffic by bandwidth measurement in the network. This allows the server/administrator to keep track of each transaction taking place on the network and helps to take necessary actions to increase the bandwidth allocation to the particular client based on his/her scope of work and to detect frequent security intrusions caused in a particular client’s session activity. When these methodologies are taught to the organizations and their clients, the awareness increases, which minimizes the misuse of organisational resources due to human behaviour.

### 2.5.1 Server Side

On the server side, there would be list of all the systems available for monitoring client web activities. The administrator can view the regular activities of the clients. The administrator can reset the activities of the client. For any unusual behavior in network bandwidth, the administrator would get an alert message prompt
stating the system name and the exceeded bandwidth activity. In the server, some blacklisted websites are predefined. Client visits to such websites will be blocked by the system. If a client visits any pre-configured security ports, an alert message will be generated to the server/administrator.

2.5.2 Client Side

As the client starts the system, the monitoring activity starts. In the monitoring activity; the login time, logout time, web activity, and network bandwidth will be monitored. These activities will be recorded in the database. For any unusual usage of network bandwidth, the administrator would get a notification message. If a client visits any blacklisted websites, the browser gets blocked.
2.6 System Architecture

Figure 2 shows the system architecture of this application for network monitoring, network bandwidth monitoring by user behaviour, blocking blacklisted websites, and port blocker.

Figure 2: Proposed System Architecture
2.7 Proposed System Design

Figure 3: Flow Chart of System
3 Functionality of Application

This application is comprised of 3 phases of architecture. They are:

- The primary phase is comprised of the user interface. In this phase the traffic is supervised. When dealing with traffic, the message is shown in the interface.

- The second phase is the monitoring system. It monitors traffic and network bandwidth. If monitored network bandwidth exceeds threshold value, an alert message will be generated and sent to the administrator/server.

- The database is the third phase. This is comprised of the client data or the signature of the client that would be proportionate to the supervised traffic. The supervised traffic and monitored network bandwidth will be preserved.

3.1 Module Description

The following is the detailed description of each module involved in this project

1. Monitoring system
   a. Network Traffic Routing
   b. Network Monitoring
   c. Network Bandwidth Monitoring

2. Alert Messaging

3. Database Connectivity

4. Blocked Browser
5. Port blocker

3.1.1 Monitoring System

3.1.1.1 Network Traffic Routing

In this application the proxy server is used to direct the network traffic via server. Thus, the network traffic of all the systems which are linked through LAN should cross through the server. In order to identify the anomalous behavior in the network set-up, it is necessary to supervise the exit and entry of traffic in the network. Here it is obligatory to supervise the traffic in the LAN.

3.1.1.2 Network Monitoring

The network will be monitored by the server continuously and these monitoring activities will be recorded into the database specifying web activity, login time, and logout time. By doing this, the recorded activity will be reviewed by the administrator if there are any issues related to the client, to view working status, etc.

3.1.1.3 Network Bandwidth Monitoring

The bandwidth of network activities will be monitored by the server. The usage of bandwidth by the client will be tracked by the server from login to logout, and then total bandwidth will be stored in the database. If usage of bandwidth exceeds the threshold, an alert message will be generated and sent to the server. When the client visits a particular website, the bandwidth will be calculated based on packets involved in accessing that website.
3.1.2 Alert Messaging

Whenever network bandwidth exceeds the threshold, an attentive notification will be generated. Whenever a client tries to access blocked ports, and also when a client tries to enter the wrong password or user id, an alert message will be generated to the server. At this point, keep away from the prompt of fake attentive. Furthermore, the screen resolution as well as attentive notification window location has been taken into account to keep away from overlaying of the prompts.

3.1.3 Database connectivity

The database part of the project is maintained by this module. It preserves the figures in relation to every client. The data configuration is similar to a hash table. The memory is assigned vigorously at the same time as the fresh clients are joined in to the network. Moreover, their data is instinctively saved in the database. Whichever data is saved in the database is totally behavior oriented. Recurrently, the database is revised in order to keep up with the revised user id and password of the client. In the database; the network monitoring activities, network bandwidth monitoring activities, and blocked ports will be stored.

3.1.4 Blocked Browser

When a client visits any blacklisted websites, then the system scans the port through which this access is being requested. The system then compares this request by converting the web address into strings and by comparing with the blacklisted website lists. Therefore, the system blocks the access if it matches the website listed on the list. Otherwise, the client can visit the entered website.
3.1.5 Port Blocker

The system generates an alert when a preconfigured port is attacked by internal and external sources. This will allow organizations to protect the internal server data from intrusions.

3.2 Object Oriented Analysis and Design (UML)

This application follows the UML design guidelines in the software development process. The use-case diagrams are analyzed and sequence diagrams are designed for behavior of the system.

Figure 4 shows the use-case diagram for the administrator. The administrator starts the server program, views all the system details, monitors the client’s activities, and can reset their activities to start recording from beginning and store fresh data to the database. If their activities do not match with the recorded activity, i.e. the login/logout details and the website visited, it means it’s irregular behavior and an alert message will be generated at the server about the irregular behavior.
Figure 4: Use-case diagram for Administrator

Figure 5 shows the use-case diagram for a client. As a client starts the application, the client can view the last login details and the application will report to the server about the current login details. The details include the time the client logged in and all his/her activities (i.e. list of all the website addresses which he/she viewed). After all the activities are done, the client sends the log-out time and details, then the client program stops.
Figure 5: Use-case diagram for Client

Figure 6 shows the sequence diagram for a client. Initially the client program starts and the user/client executes the program by sending the log-in details to the server. The client sends the list of all the websites being visited. This information from the client to the server will be sent in the form of packets and be captured by the server. Then once all the activities are reported, the client will send the log-out details to the server.
Figure 6 shows the sequence diagram for the client. The client/server will monitor all of the user/client systems for any new activities and retrieve any required data. Initially, the server will record all of the activities and store it in a fixed table. Once the client does some unusual activities, then it will be alerted and a notification will be sent to the server.

Figure 7 shows the sequence diagram for the administrator. The administrator will start the server program. The administrator/server will monitor all of the user/client systems for any new activities and retrieve any required data. Initially, the server will record all of the activities and store it in a fixed table. Once the client does some unusual activities, then it will be alerted and a notification will be sent to the server.
Sequence Diagram: Administrator

Figure 7: Sequence Diagram for Administrator
4 Evaluation and Results

4.1 Test case 1

Test case 1 is comprised of server interface and shows the list of clients connected to server, as well as the system bandwidth usage button, and can reset the user behaviour at any point in time.

![Server Interface](image)

Figure 8: Server Interface

4.2 Test case 2

Test case 2 shows the small Graphical User Interface (GUI) of the client side program. When a client starts its part of program, it shows the last log-in time. On a particular day, a client might login any number of times, but the very first log-in time will be recorded for that day. Similarly, the last log-out time of the day is considered to be recorded in the
database. The client may log-out from the application any time. Figure 9 shows the client’s last login time with system name.

![Client Interface](image)

**Figure 9: Client Interface**

### 4.3 Test Case 3

Test case 3 shows the client console. The data will be sent from client to server in the form of packets. The same thing is shown in Figure 10 where the packet capturing is started and all the activities of the client will be captured in the form of packets and will be monitored by the server.

![Client Console](image)

**Figure 10: Client Console**
4.4 Test case 4

Test case 4 is comprised of the server console where the monitoring of client activities takes place. It monitors each and every packet involved in accessing web activity and inserts this activity into the database.

![Server Console](image11.png)

Figure 11: Server Console

4.5 Test case 5

Test case 5 shows the entire web activity of the client stored in the database by the server. It records each and every web activity with dates.
4.6 Test case 6

Test case 6 is comprised of the login time and logout time with dates. This application allows multiple login time and logout time of clients.
Figure 13: Database showing entire login and logout time of client

4.7 Test case 7

Test case 7 shows the network bandwidth used by a client. It also shows usage with time and date. If usage doesn’t exceed, it shows false and if it does exceed, it shows true and an alert will be sent to the server.
Figure 14: system usage of client recorded in database

4.8 Test case 8

Test case 8 shows the alert message when client’s bandwidth usage exceeds the threshold value. Here it is configured to 0MB. If the client exceeds this threshold value, the alert will be generated showing unusual behaviour.
Figure 15: Alert message to server if bandwidth exceeds

4.9 Test case 9

Figure 16 shows the blocked site when the client accesses a predefined blacklisted website. The client gets blocked and no further access will take place.

Figure 16: Browser gets Blocked when client accesses blacklisted website

4.10 Test case 10

Test case 10 is comprised of the login activity of the client. The client should give his/her assigned user id and password for login. If the client fails to give this information, or
another user may login with other user id or password, the alert will be generated to the server about this unusual behaviour.

![Client login session](image)

**Figure 17: Client login session**

4.11 Test case 11

Test case 11 is comprised of alert messages from both source port and destination port. Since in the database blocked type is set as 2 i.e., data from destination and source will be captured and give alert to administrator/server.
4.12 Advantages & Disadvantages:

Some of the key advantages of this dashboard style suite are as follows:

1. This tool enables the administrator to measure the bandwidth usage in the network as well as by individual clients. This information can be used to optimize the bandwidth requirements and can enhance complete network performance, speed, and availability.

2. This single tool can capture multiple unusual client behaviours in real-time. This helps in minimizing the bandwidth overhead caused by the usage of multiple software applications.

3. No hardware is required to implement this tool in an organization’s network.

4. Low cost of implementation.
5. Administrator friendly dash board style for multiple unusual behaviours in one suite.

6. Ease of accessing the session activity logged in the server database.

7. Ease of generating bandwidth usage report from the server database.

Some disadvantages of this tool are listed below:

1. This tool can only detect and notify of unusual behaviours such as bandwidth exceedances and system intrusions, but will not block the exceedances.

2. This can measure the bandwidth usage by each client, but cannot optimise the usage.

3. Internal data transfer activity will not be captured in the client’s database inside the server.
5 Conclusion and Future Work

Due to the increasing information flow in the Internet and the need for rapidly and reliable accessing this information by organizations in a timely fashion without much network overhead, an optimised network is required. Network bandwidth optimization is a challenging task due to the rapid data flow, traffic speed, and the underlying network intrusion threats. Therefore, this project focuses on network monitoring aspects and some of its unusual behaviors in real-time. In this project a proxy server is used to capture all of the data passing through the network and a Java program is developed to capture the packet flow in the network layer. Based on the packet flow and a predefined threshold value, the system computes the bandwidth usage of the client. The server is configured to log client’s session activity and also alert the server when a particular client has exceeds his/her bandwidth limit. Security intrusion detection is the second part of this project, and this part has two independent scenarios: unauthorized third party login attempts to a synced client and server and multiple failed client login attempts will generate an alert to the server about a possible network intrusion attempt at the identified client location. The third aspect of this project is to block the blacklisted websites and to record the session activity continuously.

Therefore, this project can be used as a comprehensive tool with buttons for administrator to identify the session activity and bandwidth exceedances. This tool can be used as a network management panel with configured buttons at the server side to track unusual behaviors. With this application, an organization’s network security will be enhanced since all the irregularities will be captured and analyzed thoroughly. This
eliminates the need for multiple software tools to detect the unusual behaviors in the network and thereby helps reducing the cost.

**Future Work**

As part of future work, to enhance the features, one can focus on capturing the web certificates information in place of predefined blacklisted websites and block the access by filtering these web certificates when compared to the predefined category certificates. Another future aspect would be, to create the amalgamation with registry setting to obtain the log-in time; and at the same time system log-in time and log-out time. This keeps software entry in the registry with the intention that there is no need for opening and closing the program particularly by the client. The other network unusual behaviors such as illegal software downloads, license theft identification, blocking the data backups, etc. can be incorporated into this developed dashboard panel for advanced network management. Internal website password requirement, etc. can also be added. Therefore, this tool can be used as a server management panel for tracking other unusual behaviors that arise in future.
Bibliography and References


