Chapter 1

1. Introduction

1.1. Background of study

Almost all of the people use the internet to carry out essential activities such as bill payment, bank transfer and etc. But attacks towards home network are not uncommon nowadays as everybody is connected to each other through the internet and the attack has been growing more frequent and severe. When an attack does occur, it is essential that a comprehensive and organised analysis is conducted to verify the causes of the attack and the damages of the attack. A thorough and timely investigation and response can serve to minimize network downtime and ensure that critical business systems are maintained in full operation.

The level of connectivity worldwide has provided opportunities for cybercriminals who make a living breaking into networks, as well as amateur hackers who have too much time on their hands. The determined hacker can find a way into your network either by establishing some type of connection and entering your virtual "front door" or by using social engineering tactics to obtain user ID and password information. Whatever the method used, the fact is that an intruder can get into your network and harm your business.

1.2. Problem Statement

The problem statement of this project is:

- An unauthorized access into a home network system

Firewall and anti-virus (AV) won’t enough against any intrusion. Without a good detection system, a computer network will be access by an unauthorized individual may do harm to others by stealing other people’s data not to mention confidential information would be compromise. A Denial of Service (DoS) attacks may also occur

1.3. Objectives

The objectives of this project are:

- To monitor the traffic flow for any malicious activities of a network in real-time.
- To prevent abuse or overload from bandwidth and Denial of Service.
- To develop an intrusion detection system for Windows-based operating system.

The rapid advancement of technology gave us information in an instance. Network connection is vital in personal usage as with this connection we may gain an extra edge in knowledge information. With this advancement come a few problems such as spam, virus and etc. Therefore, a solution is needed to prevent those attacks before it happens.

Intrusion can occur internally or externally. An internal intrusion is an intrusion from within own networking system. They have an access to the networking system. It may be a friend, partner, employee, or even disgruntled client. External intrusion as it sounds is an intrusion from outside of the network system. Also known as attack from the internet.

Network-based intrusion detection places sensors inside a private network, between routers or a switch. This breaks up a network into multiple smaller networks. The sensors test programs at the network level, and the sensors recognize the activity of the program as normal or abnormal, based on existing comparison parameters. The sensor determines if the program is from outside the network, and how to treat it if it is.

Educating household internet user on the benefit of having an intrusion detection system on top of firewall and antivirus.

Chapter 2

2. Literature Review

2.1. Types of Network Intrusion Detection System

Stated by (Kazienko & Dorosz, 2003), an Intrusion Detection System is a defence mechanism, which detects hostile activities in a network. System will be compromise if the intrusion is not detected and possible prevented. One of the major benefits of intrusion detection system is it provides an overview of any unusual unscrupulous activities. According to (Amoroso, 1999), intrusion detection is "a process of identifying and responding to malicious activity targeted at computing and networking resources".

Even though there are firewall and antivirus programs installed to protect their computer from any unwanted access, it can still be vulnerable to any unauthorized user. With the inclusion of network intrusion detection and prevention system, there will be another protection layer against potential hackers.

Intrusion detection and prevention systems are much more secure than common firewall technology. Although considered to be an expansion of the original intrusion detection system, they are actually more a way of controlling who has access to a computer network. They not only control access, but also detect entry to the network, so the two systems are closely linked.

There are 4 types of detection system. One of the systems is network-based detection system where it is mostly used on virtual private servers, remote access servers, and routers by analyzing various network protocols (Sturmer, 2013). Wireless intrusion detection system works much like network-based system only that it applies on wireless networks (Adams).

Access point misuse is one of the illegal activities that are monitored by the system. In host- based system, works on an individual computer. Any changes on the file system, abnormal network traffic and odd application process (Sturmer). Whereas for network behavior analysis, it detects any irregularity in the system and the also the amount of traffic flow of the network (Seehorn).

There are two types of intrusion which is:

- Internal, coming from own networking system. An insider works. People whom you may know.
- External, coming from outside, frequently via the Internet.

The reason why an attack from the inside hurts more is that the insider (attacker) will take advantage of trust and physical access as resources on the local area network of the company are deemed trusted. Practically, we do not firmly restrict their activities because an attempt to control these trusted users too closely will impede the free flow of business. With the increasing numbers of internal intrusion in the industry and tougher regulatory and compliance requirements, organizations are facing tough challenges to protect both their sensitive data against internal threats and meet regulatory and compliance requirements.

Statistics from (Magalhaes, 2003):

- Almost 90% of interconnected networks that uses Intrusion Detection System detected computer security breach in the last 12 months even though there are several firewalls installed.
- Computer Security Institute, 4/7/02 reported that 80% reported financial losses in excess of $455M was caused by intrusion and malicious acts thereafter.
- Millions of jobs have been affected because of intrusion.
- Only 0.1% of companies are spending the appropriate budget on Intrusion Detection System.
- Intrusion Detection System are mostly mistaken as a firewall or its substitute.
- By using Intrusion Detection System will act as an additional barrier on top of an antivirus. Most organizations using antivirus software do not use IDS.

2.2. Network-based Intrusion Prevention System

Network-based Intrusion Prevention System products are usually deployed on the network, which means that the [software] acts like a network firewall. It receives packets, analyses them, decides whether they should be authorized, and allows acceptable packets to pass through.

(Scarfone & Mell, February 2007) Mentions that “Network-based Intrusion Prevention

System typically performs extensive logging of data related to detected events”. The logged data is used to review for any alerts and incidents (Scarfone & Mell). Some of the data that are commonly logged by network-based Intrusion Prevention System are:

- Timestamp
- Packet ID
- Event or action type
- Rating (e.g., priority, severity, impact, confidence)
- Network, transport, and application layer protocols
- Source and destination IP addresses
- Source and destination TCP or UDP ports, or ICMP types and codes
- Number of bytes transmitted over the connection
- Decoded payload data, such as application requests and responses
- State-related information (e.g., authenticated username)

Some network-based Intrusion Prevention System offer limited information gathering capabilities, which means that they can collect information on hosts and the network activity involving those hosts. Examples of information gathering capabilities are as follows:

- Identifying Hosts.
- Identifying Operating Systems.

- Identifying Applications and Network Characteristics.

2.3. Network Intrusion Detection System

Intrusion Detection System monitors all incoming and outgoing network activity and distinguishing weird patterns that show an attempt to break into the network. IDS can serve to confirm secure configuration and operation of other security mechanisms such as firewalls.

(Rozenblum, 2001) Mentions some of the intrusion detection system functions:

- Monitoring and analyzing both user and system activities.
- Analyzing system configurations and vulnerabilities.
- Assessing system and file integrity.
- Ability to recognize patterns typical of attacks by using signature or rules.
- Analysis of any abnormal network activity patterns.
- Tracking for any policy violations.

By identifying your network topology and its incoming points, Intrusion Detection sensors may be installed and configured to report to a central management console. An administrator would review the logs, manage the sensors and update the signatures.

2.4. Phishing

Phishing is a form social engineering whereby the internet is used to gather private information from businesses and individuals in deceptively way (Parno, Kuo, & Perrig). It is similar to fishing as “the phisher puts the lure hoping to fool at least a few of the prey that

encounter the bait” (Rouse M, 2007). Most notable method of phishing is the mock website. A link (email) to the website is sent to the victim which direct the victim to a mock website that looks identical with the real website (Bo, 2010).

In 1996 hackers were stealing American Online (AOL) accounts, which is when the word phishing was founded (Reid, 2009). AOL user’s passwords were being stolen by the hackers (Reid). Reid also mentions that the first mention of phishing was made in 2600 hacker newsgroup in January 1996 on the internet.

2.5. Detecting and Preventing Phishing

With an Intrusion Detection System, it is possible to get the signature/characteristics of known phisher, thereby blocking their IP (Liniger & Vines, 2005). Most Intrusion Detection System vendors provide such information (Liniger & Vines).

It is crucial that an Intrusion Detection and Prevention System is able to protect the user and its organization as phishing attacks become more rampant, focus and sophisticated (Kaspian, 2013). Nowadays, email inspection is not enough as there is increasing of social media adoption (Kaspian).

Chapter 3

3. Project Work

3.1. Project Methodology

Methodology is known as the way a process or procedure used during the delivery of a project.

Research for the project was done by discussion between the developer and the client. The methodology used in this project is iterative development model. All phases in System Development Life Cycle are included in iterative development. Phases in iterative development model are:

- Planning – To plan what is needed to be done to make sure this system can be implemented on time.
- Analysis & Design – To determine the problem and solution.
- Implementation – To take the solution and implement it. Building the system.
- Deployment - Install the system and provide user manual, training and maintenance.
- Testing - Testing is conducted to make sure that each unit meets the user's requirement.
- Evaluation – Does the system follow the standards given?

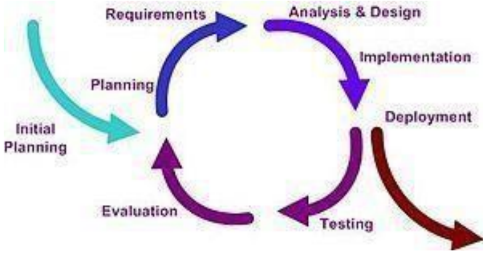


Figure 1: Iteration Development Model

Below are the project activities involved in developing Network Intrusion Detection System:

Planning is conducted during the early stages to determine the purpose of the system and the needs of the potential user.

Design of the system was done after the requirement has been identified by the developer based on the planning stage.

Construction (implementation) of the system was made with code from the design stage.

Testing of the system was conducted in the testing stage to determine if the system meets the requirement gathers during the early part of the development.

Evaluation is the stage whereby testing is conducted on the system based on the requirement.

Deployment will be conducted after the system meets the requirements and if there are no additional requirements.

3.2. Use Case Diagram

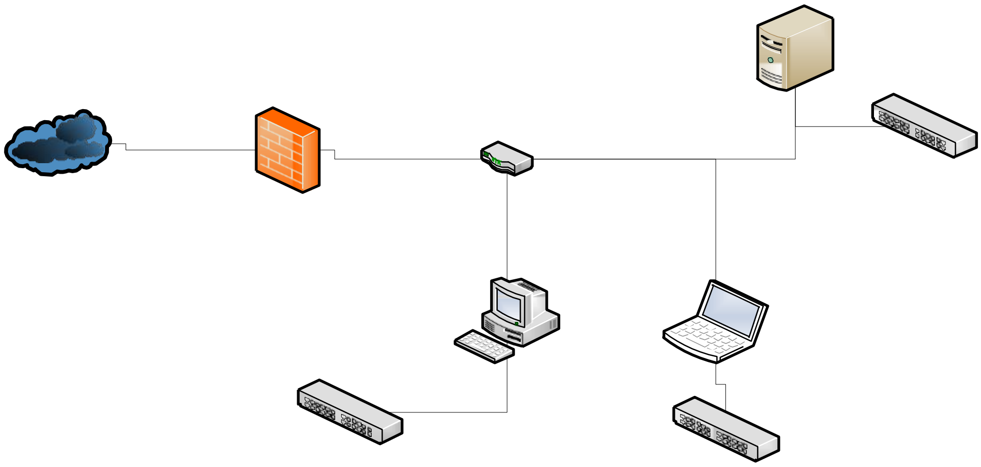


Figure 2: Detailed network diagram of HIDS location

Figure 2 shows an example of deployment of host-based intrusion detection system in a network layout. This system will be installed in each of the host computers in a network system. It will detect any anomaly that is going into the host computers through the given rules.

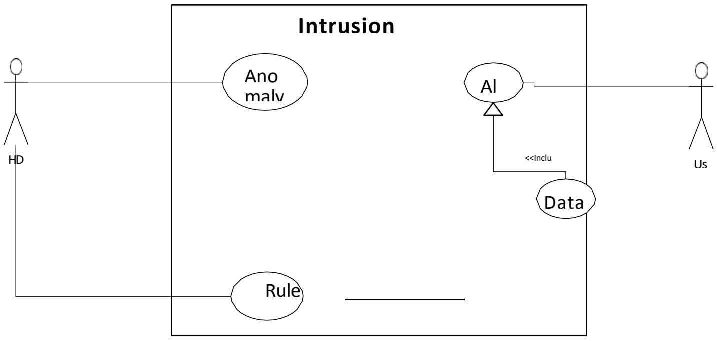


Figure 3: Use Case diagram for Intrusion Detection System

1. Anomaly Detection

If Intrusion Detection System detects any abnormality in the network traffic, then it triggers the alert system

2. Signature recognition

Intrusion Detection System examines the traffic looking for well-known patterns of attack, which are saved in pattern database and triggers the alert system, if a match is found.

3. Alert System

Whenever triggered by anomaly detection or signature recognition, it alerts the system administrator.

4. Result & Discussion

4.1. Expected Feature of the System

From the literature review, it is to be known some of the feature of the system.

The features are:

- Data logging
- Real-time detection
- Installed in host/user computer. No need third party peripherals.

There are two types of detection method:

- Signature/Rule detection. The system detects known phishing by comparing them with pre-configured and pre-determined attack patterns.
- Anomaly detection. The system detects phishing when anomalous traffic is detected.

As of now, the system will be using signature/rule detection but anomaly detection may also be used if there is any free-time.

4.2. Preliminary Findings

There are two types of programs that have been found to be the most suitable for intrusion detection system. The programs are:

- SNORT
- SURICATA

Parameter	Snort	Suricata
Developer	Sourcefire, Inc.	Open Information Security Foundation (OISF)
Availability	1998	2q009
Coded Language	C	C
Operating System	Cross-Platform	Cross-Platform
Threads	Single-Threaded	Multi-Threaded
IPv6 Support	Yes	Yes
Snort (VRT) Rules Support	Yes	Yes
Emerging Threats Rules support	Yes	Yes
Logging Format	Unified2	Unified2
Price	Free	Free

Table 1: Snort and Suricata feature comparison

Most of the tools available for Snort are applicable to Suricata.

Suricata is multi-threaded. Means that it can process faster and balance the load of identifying any intrusions.

Configuration for both programs is quite difficult especially Suricata as there is less documentation compared to Snort.

Snort is one of the most widely used intrusion detection and prevention systems. Snort uses a rule-driven language that combines the benefits of signature, protocol, and anomaly-based inspection methods. Also, it is one of the most well documented intrusion prevention systems.

Snort is the system that will be used for this project.

Even though both of these programs can be used in wide variety of operating systems, it is built to be used with UNIX based system. But as most of the computers uses Windows based system, the needs to tackle this major operating system is greater. Chart below shows the distribution of operating system usage based on Net Applications as of January 2014.

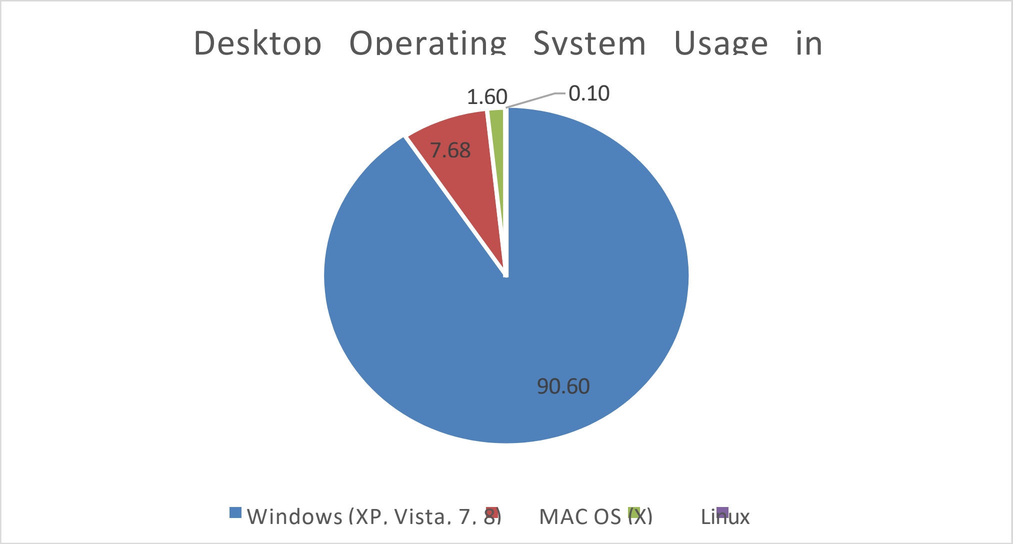


Figure 4: Desktop Operating System Usage Percentage

4.3. Snort

Snort is an open-source network intrusion detection and prevention system. Real-time analysis and packet-logging on IP networks. Analyzing each packet closely to detect any suspicious anomalies (Rouse M., 2005). Snort uses a flexible rules language to describe traffic that it should collect or pass, as well as a detection engine that uses a modular plug-in architecture.

“NSS Group, a European network security testing organization, tested Snort along with intrusion detection and prevention system products from 15 major vendors including Cisco, Computer Associates, and Symantec. According to NSS, Snort, which was the sole open-source freeware product tested, clearly out-performed the proprietary products” mention by Rouse M.

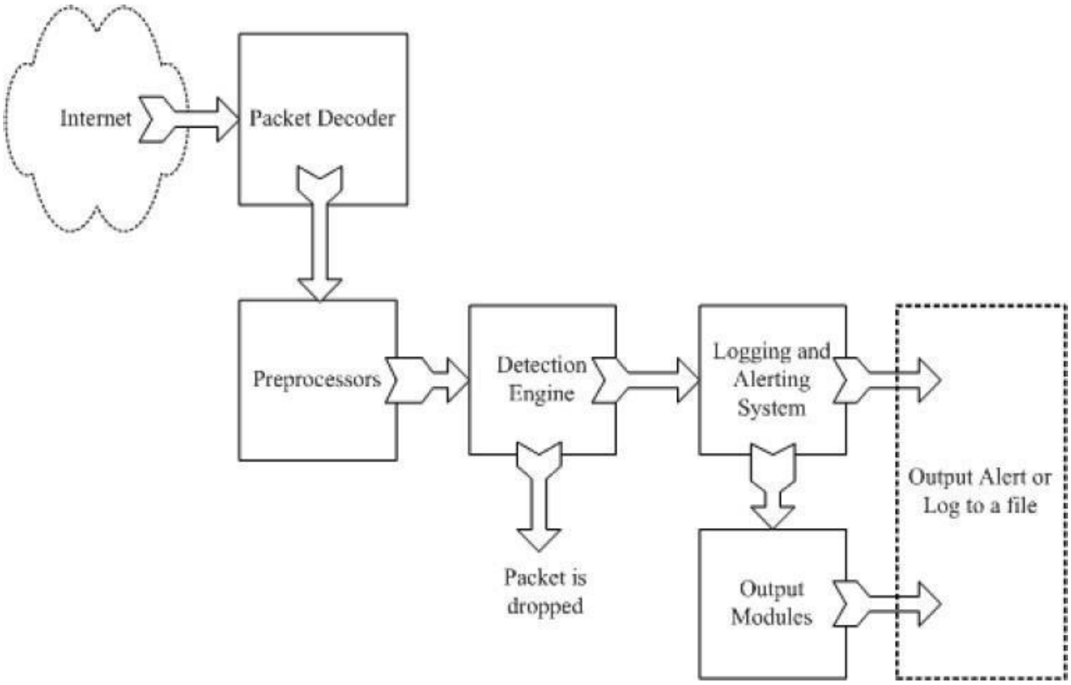


Figure 5: Components of Snort, Intrusion Detection System with snort: Advanced IDS Techniques with Snort, Apache, MongoDB, MySQL, Elasticsearch.

4.4. Rules Engine

Similar to viruses, most intruders have a sort of signature. Better to be understood as a fingerprint. Information about these signatures is used to create rules.

Two guides in writing rules are:

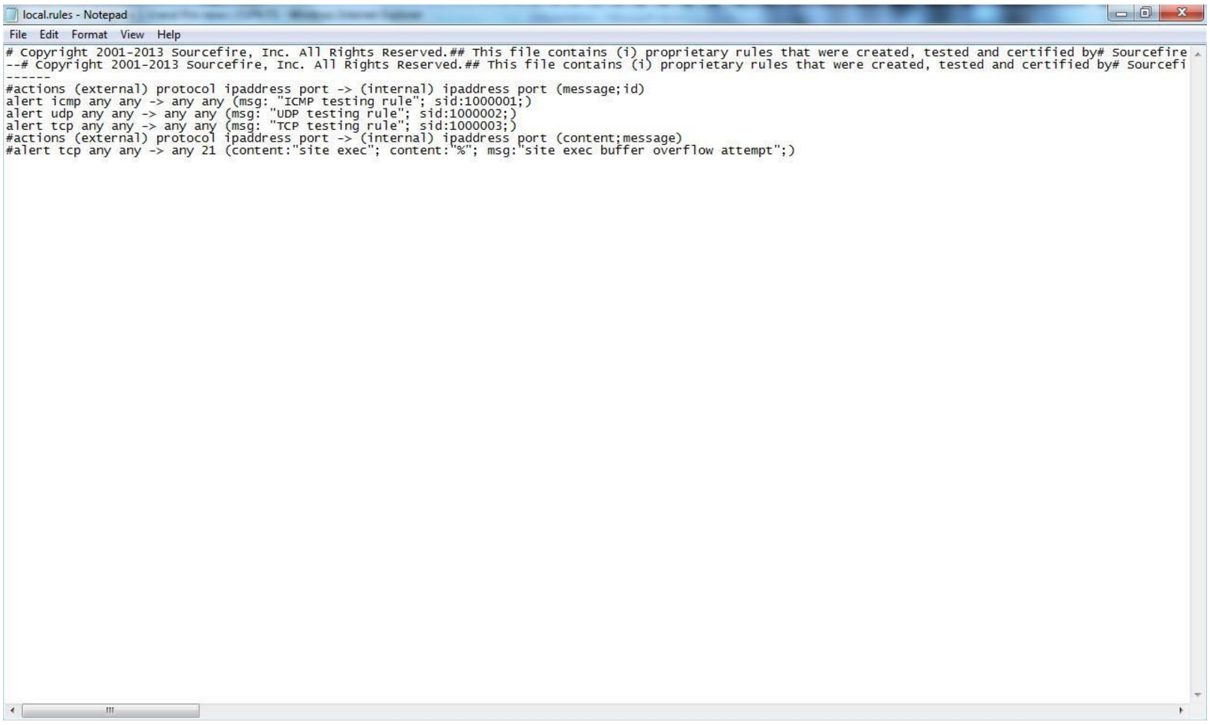
- Rules must be in a single line.
- Rules are divided into two logical sections, the *rule header* and the *rule options*. The rule header contains the rule's action, protocol, source and destination IP addresses and CIDR (Classless Inter-Domain Routing) block, and the source and destination ports information. The rule option section contains alert messages. It also contains information about which parts of the packet you should inspect to determine if you should take the rule action.

4.5. System Development

At this stage, the project enters the development and implementation phase where the software will be developed. The project was started by redefining the literature review and some part of the problem statements, scope of study and methodology. After that the process of software development started.

After a further study, the chosen program that will be developed into intrusion detection system is SNORT. This is because SNORT has a lot of user usage and with that there is more documentation to help first time user in using it. Also, implementing the rules in SNORT is a bit easier.

The main idea is to help home user in developing the intrusion detection system by themselves. There will be a guide on how to install and configure the program. Not to mention developing own rules to detect any abnormality in network usage. This development is to encounter the problem stated before which is to add another boundary to prevent an unwanted access into the network that may cause problems for the users.



```
# Copyright 2001-2013 Sourcefire, Inc. All Rights Reserved.## This file contains (i) proprietary rules that were created, tested and certified by# Sourcefire
--# Copyright 2001-2013 Sourcefire, Inc. All Rights Reserved.## This file contains (i) proprietary rules that were created, tested and certified by# Sourcefi
------
#actions (external) protocol ipaddress port -> (internal) ipaddress port (message;id)
alert icmp any any -> any any (msg: "ICMP testing rule"; sid:1000001;)
alert udp any any -> any any (msg: "UDP testing rule"; sid:1000002;)
alert tcp any any -> any any (msg: "TCP testing rule"; sid:1000003;)
#actions (external) protocol ipaddress port -> (internal) ipaddress port (content;message)
#alert tcp any any -> any 21 (content:"site exec"; content:"%"; msg:"site exec buffer overflow attempt";)
```

Figure 6: Sample rules to test the program

There are 3 actions on what to do in case of match signatures which are:

- Alert– generates an alert and then log the packet.
- Log– logs the packet.
- Pass– drops the packet. It won’t do anything.

Three network protocols that will is usually analyse are TCP, UDP and ICMP. An example rule taken from figure 6 is:

Alert tcp any any -> any any (msg: “TCP testing rule”; sid:1000003;)

This rule describes an alert that is generated when the program matches a network packet with all of the following attributes:

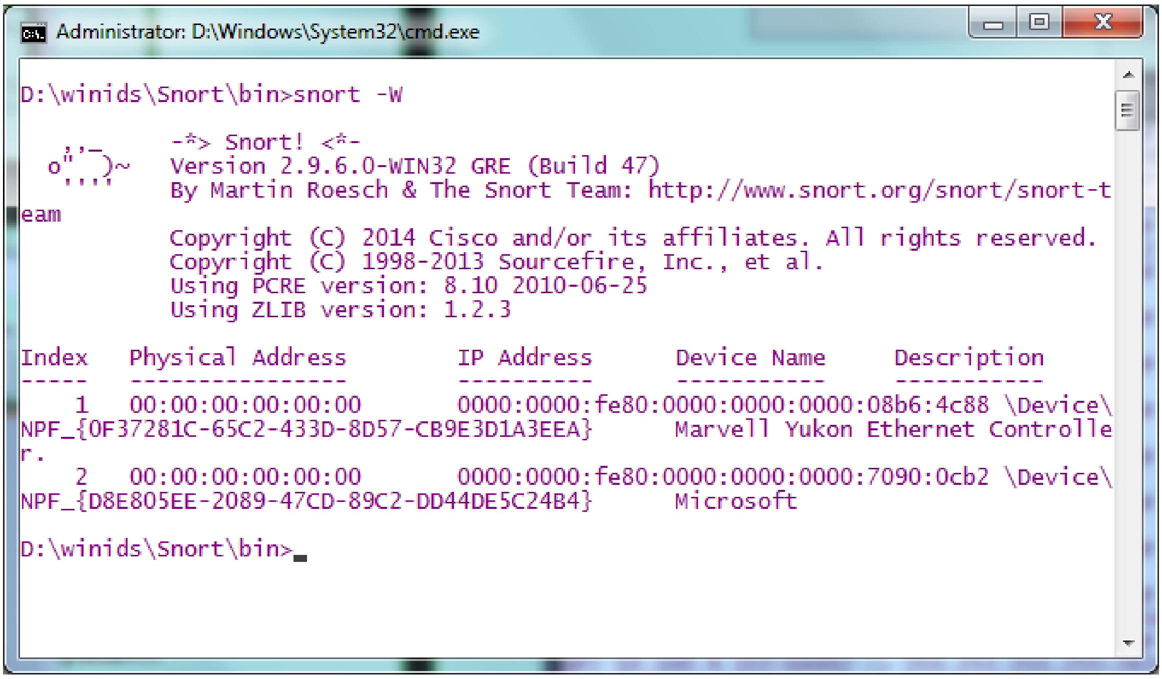
- TCP packet.
- Sourced from any IP address on any port.
- Destined for any IP address on home network on any port.
- With a message “TCP testing rule” and an ID “1000003” to log into the database.

Another example taken from figure 6 is:

alerttcp any any -> any 21 (content:”site exec”; content:”%”; msg:”site exec buffer overflow attempt”;)

This rule describes an alert that is generated when the program matches a network packet with all of the following attributes:

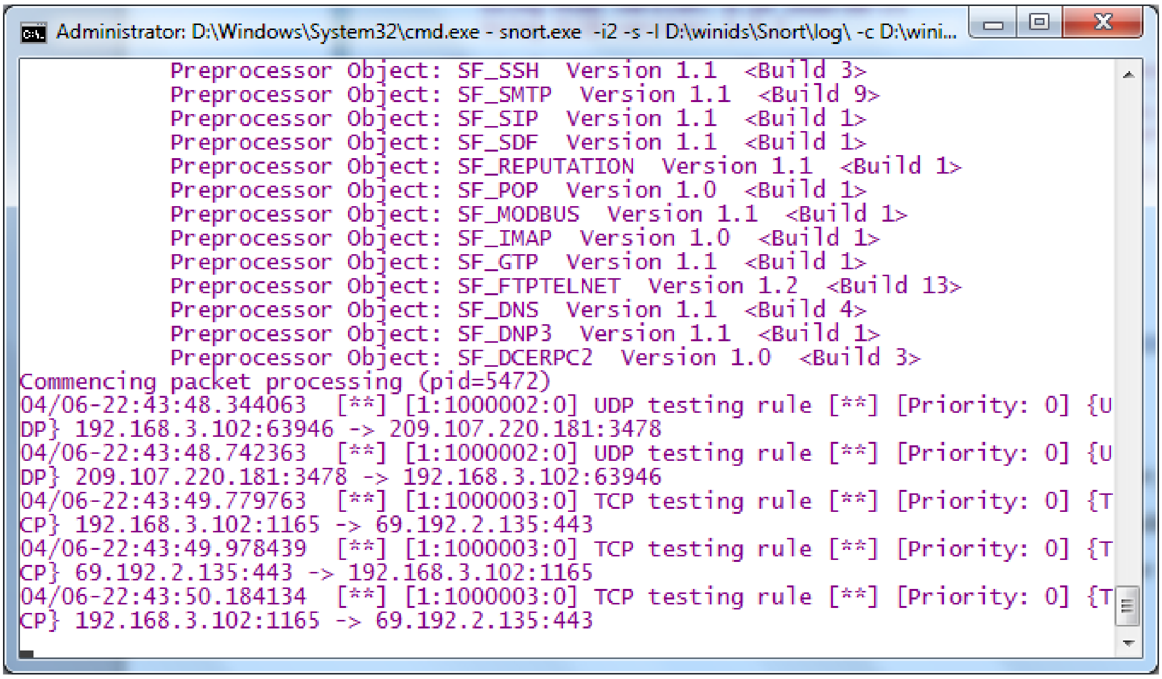
- TCP packet.
- Sourced from any IP address on any port.
- Destined for any IP address on the any home network on port 21.
- The signature contains “site exec” and “%”.
- With a message “site exec buffer overflow” to log into the server.



```
D:\winids\Snort\bin>snort -W

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.6.0-WIN32 GRE (Build 47)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/snort/snort-t
eam
           Copyright (C) 2014 Cisco and/or its affiliates. All rights reserved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using PCRE version: 8.10 2010-06-25
           Using ZLIB version: 1.2.3

Index   Physical Address         IP Address       Device Name      Description
-----   ----------------         ----------       -----------      -----------
    1   00:00:00:00:00:00        0000:0000:fe80:0000:0000:0000:08b6:4c88 \Device\
NPF_{0F37281C-65C2-433D-8D57-CB9E3D1A3EEA}      Marvell Yukon Ethernet Controlle
r.
    2   00:00:00:00:00:00        0000:0000:fe80:0000:0000:0000:7090:0cb2 \Device\
NPF_{D8E805EE-2089-47CD-89C2-DD44DE5C24B4}      Microsoft

D:\winids\Snort\bin>_
```

Figure 7: Showing the network interface of the user’s computer

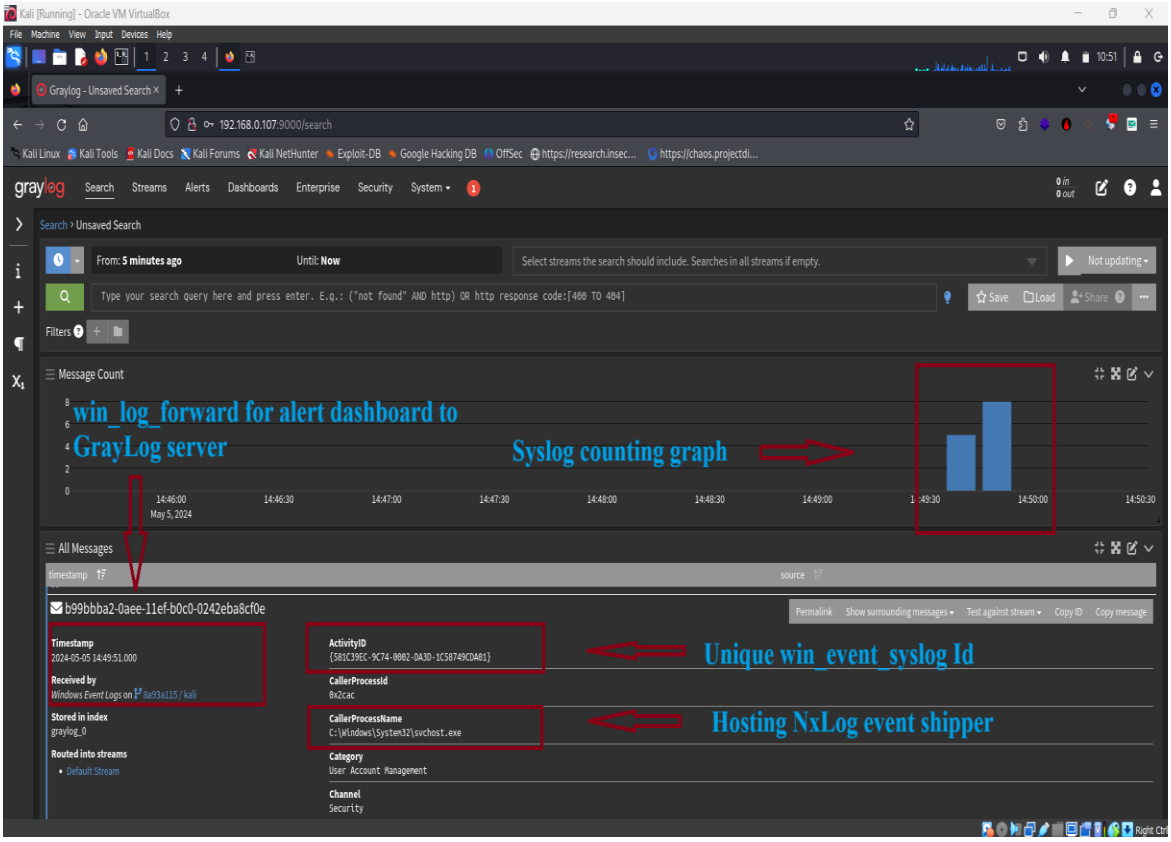
CMD (Command Prompt) is used as the interface for user to execute the program. User needs to run CMD in administrator as the program needs to run under full admin privileges. Figure 7 shows the devices of the network interface that user may choose to use for the program in detection of the network system.


Administrator: D:\Windows\System32\cmd.exe - snort.exe -i2 -s -l D:\winids\Snort\log\ -c D:\wini...

```
           Preprocessor Object: SF_SSH  Version 1.1  <Build 3>
           Preprocessor Object: SF_SMTP  Version 1.1  <Build 9>
           Preprocessor Object: SF_SIP  Version 1.1  <Build 1>
           Preprocessor Object: SF_SDF  Version 1.1  <Build 1>
           Preprocessor Object: SF_REPUTATION  Version 1.1  <Build 1>
           Preprocessor Object: SF_POP  Version 1.0  <Build 1>
           Preprocessor Object: SF_MODBUS  Version 1.1  <Build 1>
           Preprocessor Object: SF_IMAP  Version 1.0  <Build 1>
           Preprocessor Object: SF_GTP  Version 1.1  <Build 1>
           Preprocessor Object: SF_FTPTELNET  Version 1.2  <Build 13>
           Preprocessor Object: SF_DNS  Version 1.1  <Build 4>
           Preprocessor Object: SF_DNP3  Version 1.1  <Build 1>
           Preprocessor Object: SF_DCERPC2  Version 1.0  <Build 3>
Commencing packet processing (pid=5472)
04/06-22:43:48.344063  [**] [1:1000002:0] UDP testing rule [**] [Priority: 0] {U
DP} 192.168.3.102:63946 -> 209.107.220.181:3478
04/06-22:43:48.742363  [**] [1:1000002:0] UDP testing rule [**] [Priority: 0] {U
DP} 209.107.220.181:3478 -> 192.168.3.102:63946
04/06-22:43:49.779763  [**] [1:1000003:0] TCP testing rule [**] [Priority: 0] {T
CP} 192.168.3.102:1165 -> 69.192.2.135:443
04/06-22:43:49.978439  [**] [1:1000003:0] TCP testing rule [**] [Priority: 0] {T
CP} 69.192.2.135:443 -> 192.168.3.102:1165
04/06-22:43:50.184134  [**] [1:1000003:0] TCP testing rule [**] [Priority: 0] {T
CP} 192.168.3.102:1165 -> 69.192.2.135:443
_
```

Figure 8: Network packet matches the sample rule

When a network packets that matches one from the hundreds of rules, it will display the matched rule in the command prompt. Figure 8 is an example that shows the program is detecting that there are network packets that matches the sample rule and displaying it. But one of the drawbacks is when there are lots of matched rule, the previous matched packet can't be seen as the new matched packet is written just below of it.



Kali [Running] - Oracle VM VirtualBox
Graylog - Unsaved Search
192.168.0.107:9000/search
Search
Streams
Alerts
Dashboards
Enterprise
Security
System
Search › Unsaved Search
From: 5 minutes ago
Until: Now
Select streams the search should include. Searches in all streams if empty.
Not updating
Type your search query here and press enter. E.g.: ("not found" AND http) OR http response code:[400 TO 404]
Save
Load
Share
Filters
Message Count
win_log_forward for alert dashboard to GrayLog server
Syslog counting graph
14:46:00
May 5, 2024
14:46:30
14:47:00
14:47:30
14:48:00
14:48:30
14:49:00
14:49:30
14:50:00
14:50:30
All Messages
timestamp
source
Permalink
Show surrounding messages
Test against stream
Copy ID
Copy message
Unique win_event_syslog Id
Hosting NxLog event shipper

b99bbba2-0aee-11ef-b0c0-0242eba8cf0e

Timestamp
2024-05-05 14:49:51.000

Received by
Windows Event Logs on 8a93a115 / kali

Stored in index
graylog_0

Routed into streams
- Default Stream

ActivityID
{581C39EC-9C74-0002-DA3D-1C58749CDA01}

CallerProcessId
0x2cac

CallerProcessName
C:\Windows\System32\svchost.exe

Category
User Account Management

Channel
Security

Figure 9: Log File

With any detection that matches the rules, the program will log the entire detected “anomaly” in a single file. This file will keep increase in size with each and every detection as shown in figure 9. Therefore, it will be very cumbersome for the user to analyses the log. Figure 9 shows the preliminary logging for the program and it will be used only if the user wanted it.

4.6. Future Work

This assessment is only focus on developing intrusion detection system for home-based networking system. In future, a prevention mechanism is to be introduced for Windows-based system, thus improving the network security by combating attacks such as phishing and denial-of-service attack (DoS attack), hacking attempts and anything else that might compromise the network. Also as this system in tailored to individual user (household), it would be very helpful if in the future this system would be implemented in large organisation. It will a major stepping stone if this system can be built not only for detecting but also preventing as can be seen that is widely used in UNIX-based operating system.

Also, to have the rules engine to be updated without the help from the user. Auto-update the rules engine by a weekly basis.

5. Problem Challenges

There are a few obstacles in developing this program. Some of the problems are:

The program is suited best with using UNIX based operating system. Also many of the tools will run on only in UNIX based operating system.

Configuration of the program is tedious and understanding on how the program works needs a long period of time.

As I'm doing this project for Windows-based system, the Intrusion Prevention System cannot be initiated. UNIX is needed to initiate the Intrusion Prevention System. Because of that only the detection part of the system can be use.

To write the rules, I would need to be "attack" before I can write the rules as I need to determine the type of "attack" and its content.

6. Conclusion

This paper was written as an approach to detect any abnormal network activity in a home networking system. It has been shown that Intrusion Detection System is very useful for improving network security by adding another layer of security on top of the firewall and anti-virus programs.

However in reality hackers may take advantage of any possible resources to attack a computer network, and these resources may be unstable and very difficult to categorise. These difficulties can reduce the effectiveness of this approach. Thus, a more secured way of using the internet is needed for the user by only surfing the internet with an updated anti-virus and understanding the risk of surfing the “wrong” sites.