Implementation of SKONI-Hybrid Intrusion Detection System to Mitigate Network Attacks

***Konyeha, S. & Onibere, E.A.**
Department of Computer Science
University of Benin
Benin City, Nigeria
***E-mail**: susan.konyeha@uniben.edu
***Phone**: +234806826547

ABSTRACT

Intrusion Detection System (IDS) has been of interest to researchers for some time now and can provide advance warning against impending attacks due to its in-depth detection and logging of malicious activities. Most of the current intrusion detection systems have mainly concentrated on detection of intrusions with no mechanism incorporated to respond to such intrusions. Also popular intrusion detection systems detect attacks based on policy or signature and hence are able to detect known attacks only. However a hybrid intrusion detection system detects both known and unknown attacks. In this paper, we report the development of a hybrid intrusion detection system (SKONI-HIDS) that was implemented in the Department of Computer Science, University of Benin, in 2013 using Snort (an open source intrusion detection system) to detect known attacks and a statistical model to detect unknown attacks. SKONI-HIDS was trained for detection of unknown attacks using this statistical model. The information needed for training SKONI-HIDS was collected from a LAN consisting of a server (with Internet access), ten (10) clients and a switch. Clients on this network were assigned IP addresses: 192.168.0.0/24 and 192.168.1.0/24. We tested SKONI-HIDS by using live traffic on our LAN. The results were quite encouraging as SKONI-HIDS was able to detect real attacks performing both signature based and anomaly detection and finally, SKONI-HIDS was also able to prevent attacks from succeeding by automatically reconfiguring firewall rules.

Keywords: SKONI-IDS, hybrid, intrusion, detection, snort, firewall

CISDI Journal Reference Format
Konyeha, S. & Onibere, E.A. (2020): Implementation of Skoni-Hybrid Intrusion Detection Model to Mitigate Network Attacks. Computing, Information Systems & Development Informatics Journal. Vol 11 No 1, Pp 67-84
Available online at www.computing-infosystemsjournal.info

1. INTRODUCTION

An intrusion detection system (IDS) is hardware, software or a combination of both, for monitoring network or system activities to detect malicious signs. In computer security, designing a robust intrusion detection system is one of the most fundamental and important problems. The primary function of system is detecting intrusion and gives alerts when user tries to intrude in a timely manner (Jose et al, 2018). We can define Intrusion Detection Systems (IDS) as software/ hardware designed to monitor network traffic or computer activities and alert administrators of suspicious activities (Fung, 2011). IDS differ in the ways they detect attacks and handle threats. The most common approaches used to detect attacks include misuse or signature based intrusion detection, anomaly intrusion detection, and hybrid intrusion detection (Mgabile et al, 2012).

Signature based intrusion detection systems rely on a set of rules (also known as signatures) for detecting intrusion activity. A signature can be described as a conditional rule, which is tested on an instance of activity, identifying a specific type (Cole et al, 2005). These attack signatures encompass specific traffic or activity that is based on known intrusive activity (Konyeha and Onibere, 2014a). Anomaly detection is concerned with identifying events that appear to be anomalous with respect to normal system behavior. A wide variety of techniques including data mining, statistical modeling and hidden Markov models have been explored as different ways to approach the anomaly detection problem (Modi, et al, 2012).

The focus of this paper is on the development and implementation of a hybrid intrusion detection model named SKONI-HIDS for network attacks mitigation. The design of SKONI-HIDS was presented in Konyeha and Onibere (2020a). The implementation of the hybrid intrusion detection system for network attack mitigation is discussed in this paper, reporting how it is been deployed to monitor network traffic also, the performance of the developed hybrid intrusion detection system SKONI-IDS, is tested using live traffic to detect both known attack (signature based detection) and unknown attack (anomaly based detection)-

2. METHODOLOGY

SKONI-HIDS combined signature based detection using Snort rules (signature based detection) and a statistical method (anomaly detection). We have built a module which can detect unknown attacks and integrated it with Snort (an open source IDS) for signature detection. Shown in figure 1 is the setup of the experiment, which is a LAN consisting of a server with Internet connection and 10 clients (hosts). SKONI-HIDS was configured on one host on the LAN. Static IP address was configured on the server with IP address 192.168.0.1 and subnet mask 255.255.255.0. Static IP addresses were configured manually on the hosts using the following IP address 192.168.0.x (where "x" ranges from 2 to 11) with the subnet mask 255.255.255.0.

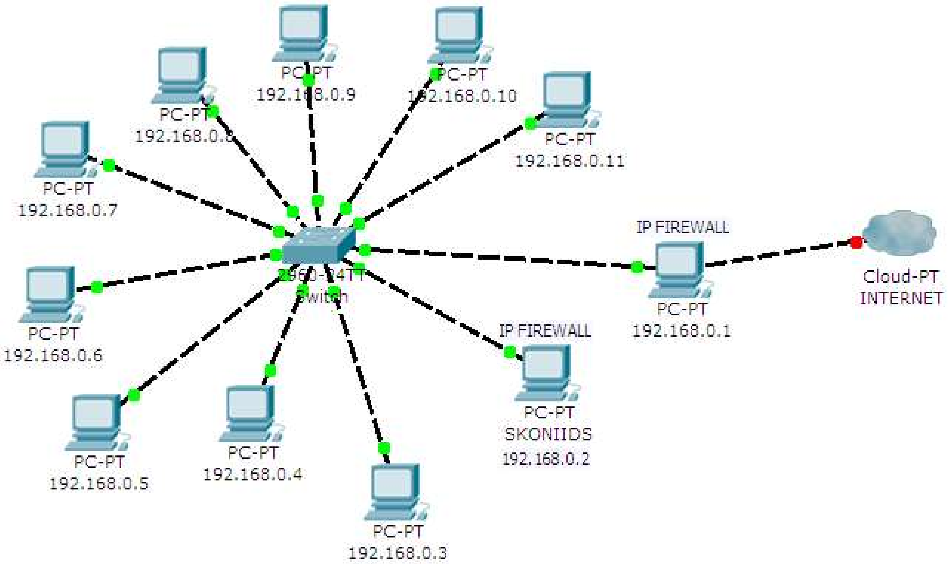


Figure 1: Showing set up of test bed

3. DATA COLLECTION

Several network data traces were collected from the cyber café to use for our experiment. Snort Snort signatures were downloaded from www.bleedingsnort.com into Snort rules folder. We included it into the snort config file and configured the network variables. Use C:\example\anomalydetector>java -jar anomalydetector-3.0.0.SKONI.jar test logfile, in the command prompt. The system begins to process the packets and match it against the rules file. It checks for any offending packets. Once there is a match, the system provides an alarm response.

If no alarm is raised, the data is simply logged into the database. In anomaly detection, training and detection were the two processes carried out. The system was trained with data free of attacks in order to generate the normal profile. In the training mode, the system learns and updates the normal profile. To train SKONI-IDS, we used a statistical formula which computes a new mean and standard deviation whenever an additional packet is captured at the network interface. (Konyeha and Onibere, 2014b).

The packets that enter the network are stored every 30 seconds in a buffer during an analysis cycle and analyzed at the end of the cycle. The anomaly analysis engine checks for anomaly in the network traffic using the sample mean and standard deviation model which is a statistical tool we used for training the anomaly detector in SKONI_IDS.

According to Denning(1987), when using the mean and standard deviation model, a new observation is abnormal if it falls outside a confidence interval that is d standard deviation from the mean for some parameter d: mean + d * stdev. The statistical formula used to update the mean and (estimated) variance of the sequence, for an additional element x_{new}, where, $\overline{x}_n$ denotes the sample mean of the first n samples (x_1, x_2 ., x_1), s_n^2 their sample variance, and σ_n^2 their population variance is stated in equations (1) - (2):

$$\overline{x}_n = \frac{(n-1)\overline{x}_{n-1} + x_n}{n} = \overline{x}_{n-1} + \frac{x_n - \overline{x}_{n-1}}{n} \qquad (1)$$

$$s_n^2 = \frac{(n-2)s_{n-1}^2 + (x_n - \overline{x}_n)(x_n - \overline{x}_{n-1})}{n-1}, n > 1 \qquad (2)$$

$x < \mu + 3(\sigma) + T$ normal traffic $(T \geq 0)$

Where
x_n = Score = Number of packets counted in the cycle (e.g, IP:Port combination)
μ = Mean of score
σ = Standard Deviation
T = Threshold
n = the Cycle no
x = packet count (score)

The anomaly analysis engine uses mean and standard deviation (computed from the variance) rather than variance itself, because the area under the normal distribution curve is obtained using the values of the mean and standard deviation. Figure 2 shows the confidence region for a normal distribution. Hence, about 68.27% of the area under the normal curve is within plus one and minus one standard deviation of the mean. About 95.4% of the area under the normal curve is within plus two and minus two standard deviation of the mean. About 99.7% of the area under the normal curve is within plus three and minus three standard deviation of the mean.

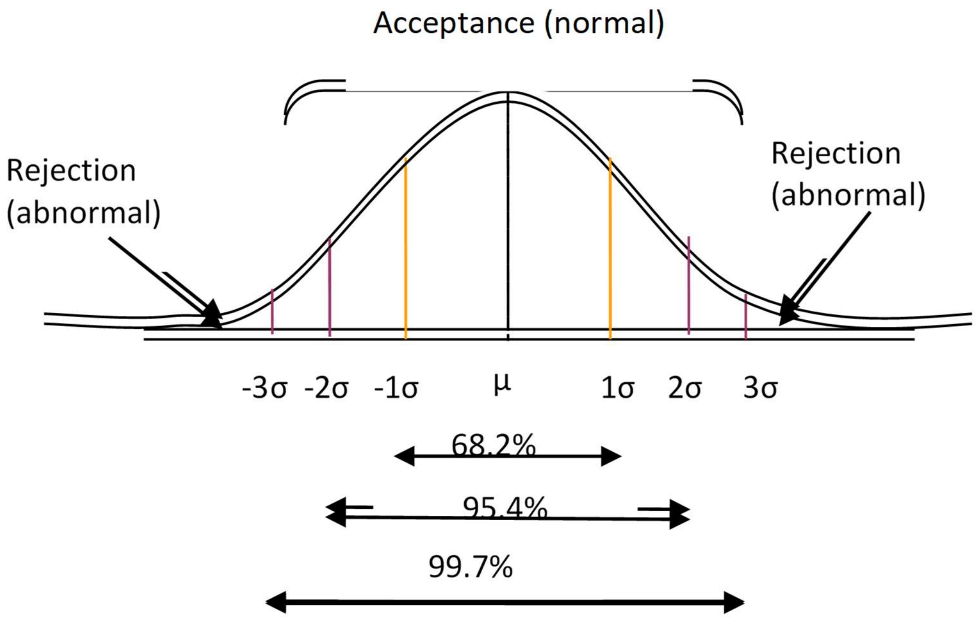


Figure 2: Normal distribution curve (Konyeha and Onibere, 2020a)

We took our baseline as the upper bound for the third standard deviation from mean plus a pre-assigned threshold value (T).

Thus our baseline is given as $x<\mu + 3\sigma + T$

where x = Score = Number of packets counted in the cycle (e.g IP:Port combination)
μ = Mean of packet counted in a cycle
σ = Standard Deviation
T = Threshold.

We defined an upper bound of $x<\mu + 3\sigma + T$ as a normal region, where the proportion of observed packet scores falling in the region is at least 99.7%. Beyond this normal region, the observed packets (IP:PORT combination) are anomalous. The graphic user interface (GUI) and anomaly detection engine components of SKONI-IDS were developed with Java programming language which is a general purpose, concurrent, class-based, object-oriented programming language.

Since JAVA is not native to windows, we used a wrapper for WinPCap called JPCap to capture the packets from the device's network interface card. After packet capture, the system was required to check if it is operating in learning mode or detection mode. When the system is working in learning mode, it would store the mean score and standard deviations (profiles) of these captured packets in an anomaly profile database. However if the system is working in detection mode, it will check the captured packets against the normal profiles of packets already recorded in the database during training. This method of detection owes to the fact that: 99.7% of the data falls under three standard deviation of the mean. The architecture for the hybrid intrusion detection system is shown in figure 3

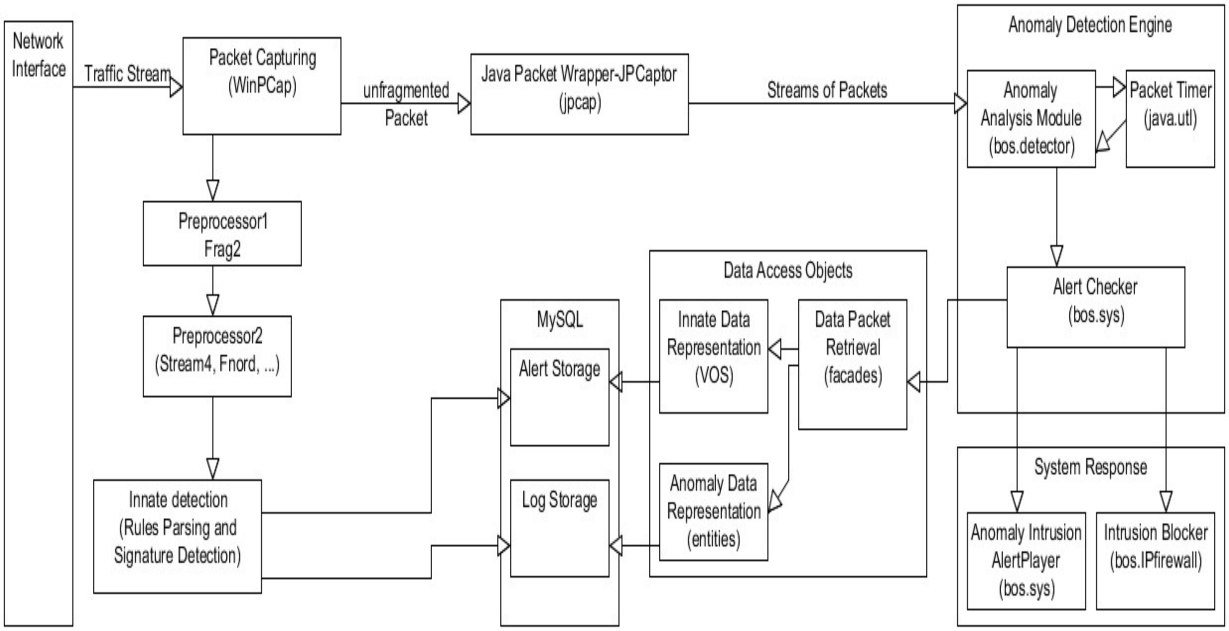

Network Interface
Traffic Stream
Packet Capturing (WinPCap)
unfragmented Packet
Java Packet Wrapper-JPCaptor (jpcap)
Streams of Packets
Anomaly Detection Engine
Anomaly Analysis Module (bos.detector)
Packet Timer (java.utl)
Alert Checker (bos.sys)
Preprocessor1 Frag2
Preprocessor2 (Stream4, Fnord, ...)
Innate detection (Rules Parsing and Signature Detection)
MySQL
Alert Storage
Log Storage
Data Access Objects
Innate Data Representation (VOS)
Data Packet Retrieval (facades)
Anomaly Data Representation (entities)
System Response
Anomaly Intrusion AlertPlayer (bos.sys)
Intrusion Blocker (bos.IPfirewall)

Fig 3: Architecture of SKONI- Hybrid Intrusion Detection System (SKONI-HIDS) for Network Attack Mitigation

Component of The Hybrid Intrusion Detection System

There are Seven (7) standard components and some latent components

Standard components

a. Network traffic Capturing component (WinPcap)
b. Pre-processor engines (Preprocessor1, Preprocessor2...)
c. Signature detection
d. Java Packet wrapper
e. Adaptive Detection Engine (Anomaly detection) (new)
f. Systems Response (new)

The latent components are the data access objects containing:

a. Data Packet Retrieval component (façade)(new)
b. Signature Data Representation (VOS)
    - Alert Storage component
    - Log storage component
c. Anomaly Data Representation (entities)
    - Alert Storage component
    - Log storage component

Description of the components

Network Traffic Capturing Component (WinPcap)
WinPCap is Windows packet capture library that enables interaction with the underlying Network Interface Card. It has the responsibility for grabbing packets directly from the network interface card. It is the capture facility for raw packets provided by the underlying operating system available to other applications.

Pre-Processor Engine Component
The preprocessor engine is designed to examine the packets in order to examine the transmitted packets against intrusion. The stream re-assembly, Back orifice control for Trojan attacks, detection and defeating of polymorphic shell code evasion by the fnord utility, and detection of malicious Address Resolution Protocol (ARP) traffic attempts by ARPspoof utility are handle by the preprocessor engine.

Data Access Objects
This block is the bridge between SKONI-HIDS and the underlying database (MySQL). And it has the following subcomponents: sources her are organized into packages, namely:

- Facades: This package contains classes that are responsible for fetching data from the database.
- Vos: This package contains classes that directly reflects the database structure of snort
- Entities: This package contain classes that reflects the database structure of SKONI-HIDS

Database or memory storage (MYSQL)
MySQL is the database or memory storage. It has two components:

- Alert Storage (Snort_alerts): This is the database that houses the tables where alerts are stored.
- Log Storage (Snort29): This is the database that houses the tables where snort's logs are stored.

Java Packet Wrapper Component (JPCAP)
JPCAP is an open source utility and is licensed under GNU LGPL. This component is responsible for capturing packets from WinPcap. It can capture Ethernet, IPv4, IPv6, ARP/RARP, TCP, UDP, and ICMPv4 packets. Since most organization supports various LAN, JPCAP can prevent MITM in LAN architecture by filtering network packets into various packets components for adaptive detection engine analysis.

Signature Detection Component

- Bos.detector: This package contains the business classes that do the actual anomaly detection
- Bos.sys: This package contains the classes that are responsible for managing snort, MySQL and apache services.
- Java.util: This is a java built in package that contains utility classes. The software uses the Timertask class for timing itself during the analysis of the incoming traffic.

The detection engine is the primary Snort component. It has two major functions: rules parsing and signature detection. The detection engine builds attack signatures by parsing Snort rules. Snort rules are read line by line, and are loaded into an internal data structure. The rules are loaded only when the Snort service is started, meaning that to modify, add, or delete a rule you must refresh the Snort daemon. The detection engine runs traffic through the now loaded rule set in the order that it loads them into memory. You can dictate which rules are run first by prioritizing and then organizing it in the manner you deem fit.

Rules are split into two functional sections:

a. The rule header (rule tree node.
b. The rule option (option tree node).

The rule header contains information about the conditions for applying the signature. The detection engine processes rule headers and rule options differently. The detection engine builds a linked list decision tree. The nodes of the tree test each incoming packet for increasingly precise signature elements. A packet is tested to see whether it is TCP; if so, it is passed to the portion of the tree that has rules for TCP. The packet is then tested to see whether it matches a source address in a rule; if so, it passes down the corresponding rule chains. This process happens until the packet matches an attack signature. if it matches, an attack is detected or else the packet is regarded as a clean packet and is dropped. That is, it is logged to storage.

Anomaly Detection Engine Component

The anomaly detection engine component is responsible for recording all abnormal activities. It uses two different operation modes, training mode and anomaly detection mode. In the training mode, the system records network traffic considered as normal and expected. A profile of the network activity is automatically created and the anomaly detection module stores this profile in a database. Each time the system is executed, the activity profile of the most active clients and servers in the network are loaded from the network profile. As the expected traffic is recorded in the database, it is compared with the real traffic passing through the network. If it detects a deviation in the traffic higher than a certain percentage, it means that something abnormal has occurred and an incidence of abnormality is registered by the system hence the system must compare the received traffic with the activity previously stored in training mode.

System Response

- Alert Player: This component handles the playing of the play-back of the preset alarm when an intrusion is detected.
- Blocker: This component is responsible for blocking the offending packets using the packet's header information.

4. RESULTS AND DISCUSSIONS

4.1 Report of Experiments Conducted

A prototype of SKONI-HIDS was implemented and tested using real time data collected on a network of 10 computers in a cyber café and the IDEVAL 1999 DARPA dataset. We conducted two groups of experiments during the period of this research. For one group of experiments we performed signature based detection and anomaly based detection using live traffic and reconfiguration of the firewall rules.

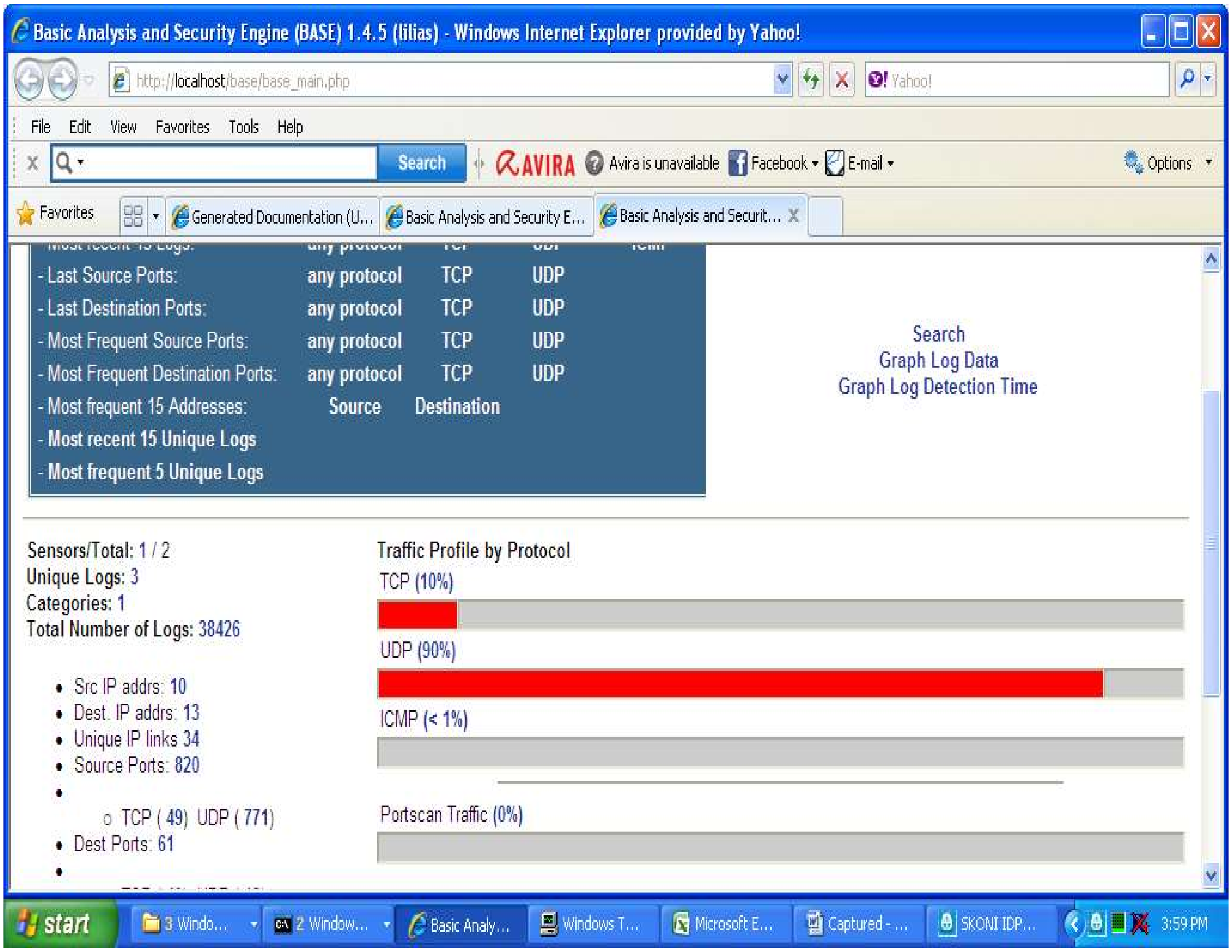
Experiments

Two data traces were collected for this experiment: self-set, consisting of normal traffic on the LAN from IP range 192.168.0.1 to 192.168.0.11 (assigned statically) and mixed traffic (i.e. self set and non-self set), consisting of traffic generated from 192.168.0.1 to 192.168.0.254 (assigned dynamically) representing the whole network (192.168.0.0 /24), such that the traffic from other IPs different from IP range 192.168.0.1 to 192.168.0.11 is non-self set and packets from these IPs will represent intrusive activity. The IDEVAL 1999 DARPA data set was also used to test for validity since it is a recognized standard data set. The dataset was partitioned into two using Wireshark, one set was used to train SKONI_IDS and as such was taken as normal data.

When the mixed data was then encountered by SKONI-HIDS, alerts of anomaly was then obtained since this dataset contained both normal and anomaly information. The experimental results using LAN and DARPA dataset are discussed below.

4.2 Experimental Results using LAN

We collected data over a 7 week period on alternate weeks for self trace (this is normal activity for the LAN). The self set was represented by 38,426 connections. A total of 49 TCP connections were logged, each of which is a data path triplet consisting of 10 source IP address and 13 destination IP addresses. The network protocols were (TCP=49, UDP=771, ICMP<1%, 820 source port and 61 destination port. A self set, consisting of 34 unique IP Links and port combinations were extracted from the set. We screen captured the result of logging this data in the web based interface made possible by BASE as shown in Figure 4.


Basic Analysis and Security Engine (BASE) 1.4.5 (lilias) - Windows Internet Explorer provided by Yahoo!
http://localhost/base/base_main.php
Yahoo!
File
Edit
View
Favorites
Tools
Help
Search
AVIRA
Avira is unavailable
Facebook
E-mail
Options
Favorites
Generated Documentation (U...
Basic Analysis and Security E...
Basic Analysis and Securit...
start
3 Windo...
2 Window...
Basic Analy...
Windows T...
Microsoft E...
Captured - ...
SKONI IDP...
3:59 PM

- Last Source Ports:	any protocol	TCP	UDP
- Last Destination Ports:	any protocol	TCP	UDP
- Most Frequent Source Ports:	any protocol	TCP	UDP
- Most Frequent Destination Ports:	any protocol	TCP	UDP
- Most frequent 15 Addresses:	Source	Destination	
- Most recent 15 Unique Logs			
- Most frequent 5 Unique Logs			

Search
Graph Log Data
Graph Log Detection Time

Sensors/Total: 1 / 2
Unique Logs: 3
Categories: 1
Total Number of Logs: 38426

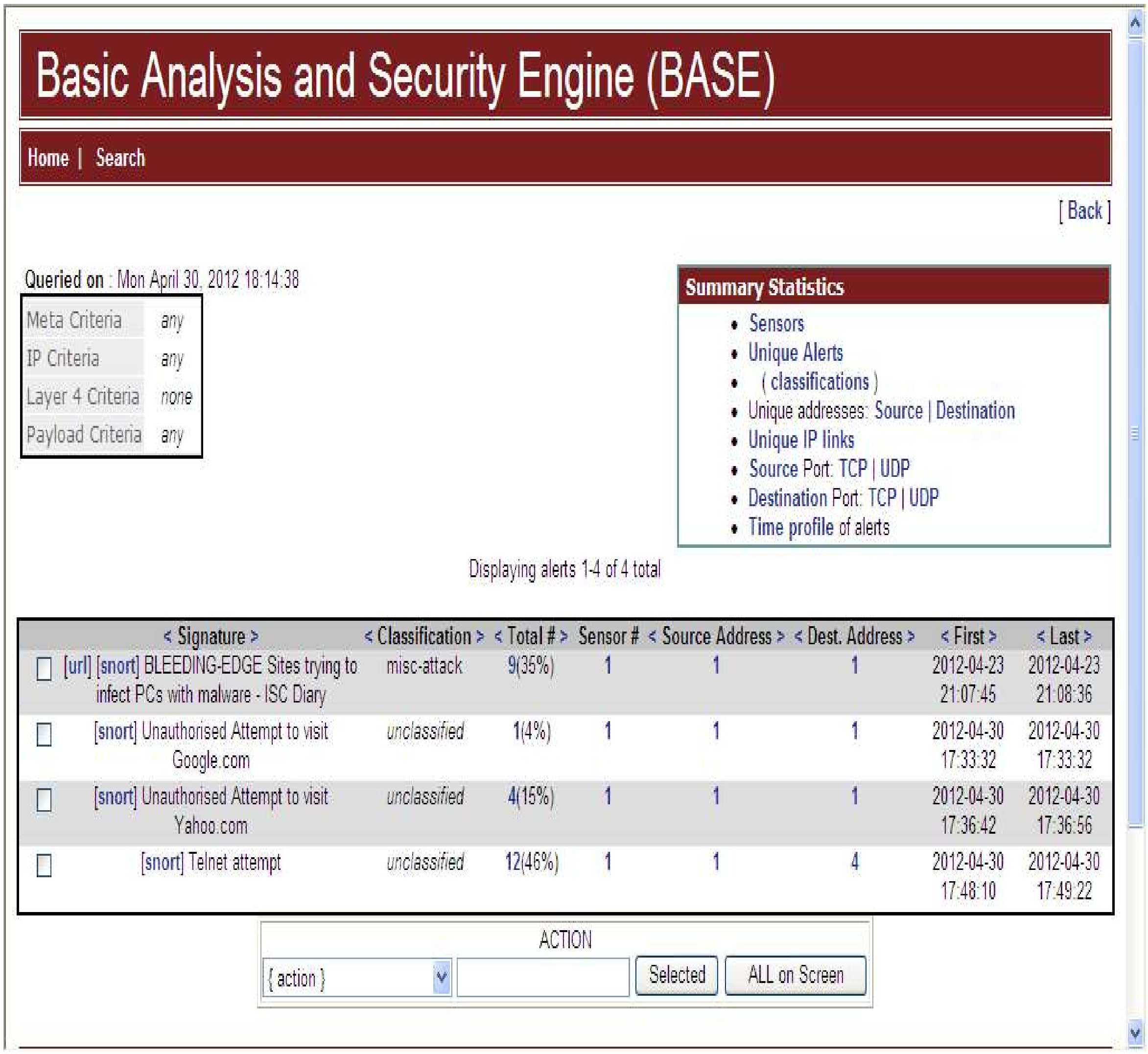
- Src IP addrs: 10
- Dest. IP addrs: 13
- Unique IP links 34
- Source Ports: 820
-
  - TCP (49) UDP (771)
- Dest Ports: 61
-

Traffic Profile by Protocol
TCP (10%)
UDP (90%)
ICMP (< 1%)
Portscan Traffic (0%)

Figure 4: 38,426 logs from our LAN

4.3 Results for signature based detection Experiment using our LAN:

We triggered alerts by performing activities on hosts on our LAN that would violate the rules of the innate detection engine and hence cause alerts to be triggered by SKONI_IDS. We recorded 4 alerts for log displayed in the web based interfaces in Figure 5.



Basic Analysis and Security Engine (BASE)

Home | Search

[Back]

Queried on : Mon April 30, 2012 18:14:38

Meta Criteria	any
IP Criteria	any
Layer 4 Criteria	none
Payload Criteria	any

Summary Statistics

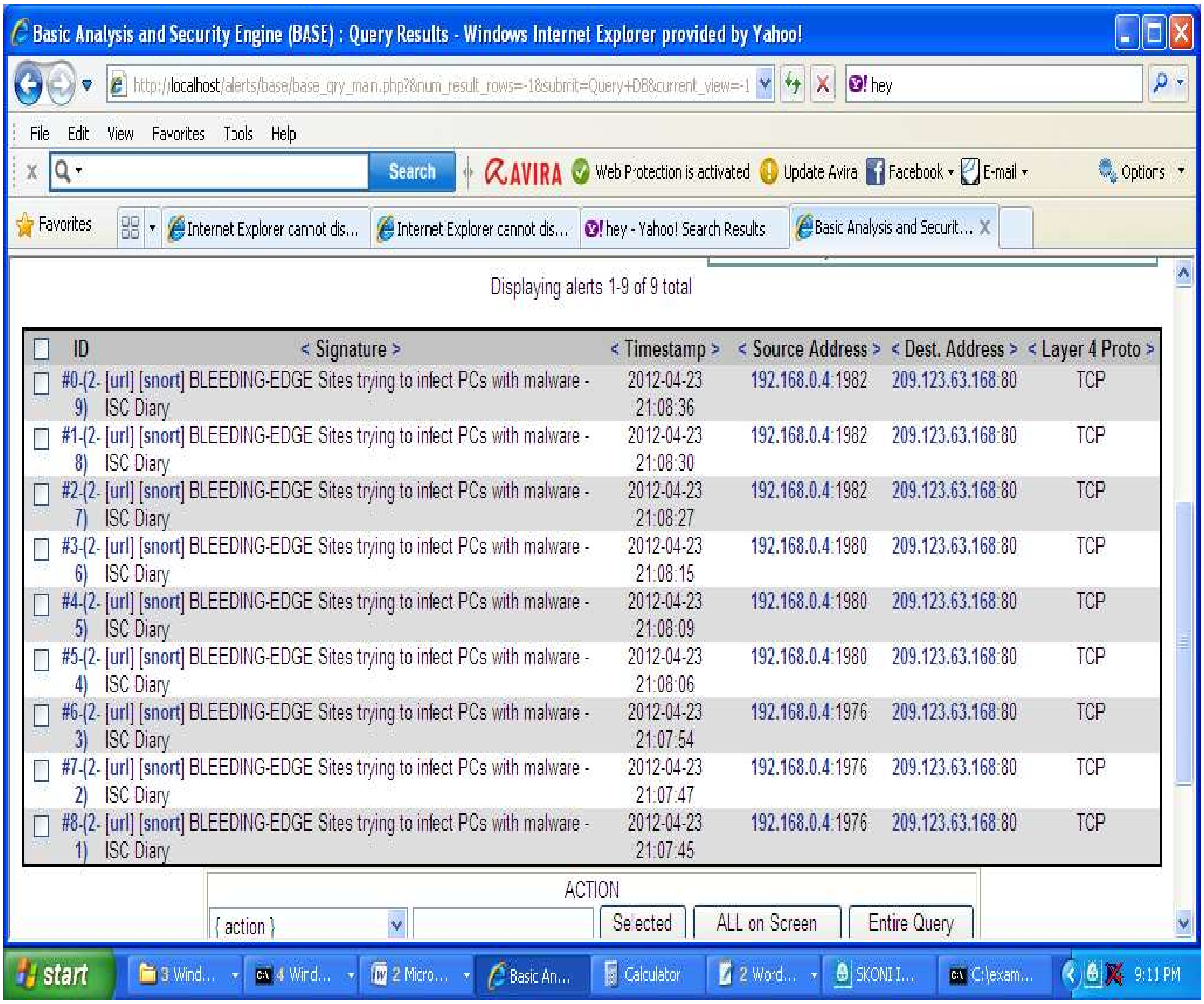
- Sensors
- Unique Alerts
- (classifications)
- Unique addresses: Source | Destination
- Unique IP links
- Source Port: TCP | UDP
- Destination Port: TCP | UDP
- Time profile of alerts

Displaying alerts 1-4 of 4 total

	< Signature >	< Classification >	< Total # >	Sensor #	< Source Address >	< Dest. Address >	< First >	< Last >
□	[url] [snort] BLEEDING-EDGE Sites trying to infect PCs with malware - ISC Diary	misc-attack	9(35%)	1	1	1	2012-04-23 21:07:45	2012-04-23 21:08:36
□	[snort] Unauthorised Attempt to visit Google.com	*unclassified*	1(4%)	1	1	1	2012-04-30 17:33:32	2012-04-30 17:33:32
□	[snort] Unauthorised Attempt to visit Yahoo.com	*unclassified*	4(15%)	1	1	1	2012-04-30 17:36:42	2012-04-30 17:36:56
□	[snort] Telnet attempt	*unclassified*	12(46%)	1	1	4	2012-04-30 17:48:10	2012-04-30 17:49:22

Figure 5: Alerts for performing unauthourised activities on hosts

We were able to detect 9 malware attacks on our LAN for the logfile analysed. The web based interface is shown in Figure 6.


Basic Analysis and Security Engine (BASE) : Query Results - Windows Internet Explorer provided by Yahoo!
http://localhost/alerts/base/base_qry_main.php?&num_result_rows=-1&submit=Query+DB&current_view=-1
hey
File Edit View Favorites Tools Help
Search
AVIRA
Web Protection is activated
Update Avira
Facebook
E-mail
Options
Favorites
Internet Explorer cannot dis...
Internet Explorer cannot dis...
hey - Yahoo! Search Results
Basic Analysis and Securit...
start
3 Wind...
4 Wind...
2 Micro...
Basic An...
Calculator
2 Word...
SKONI I...
C:\exam...
9:11 PM

Displaying alerts 1-9 of 9 total

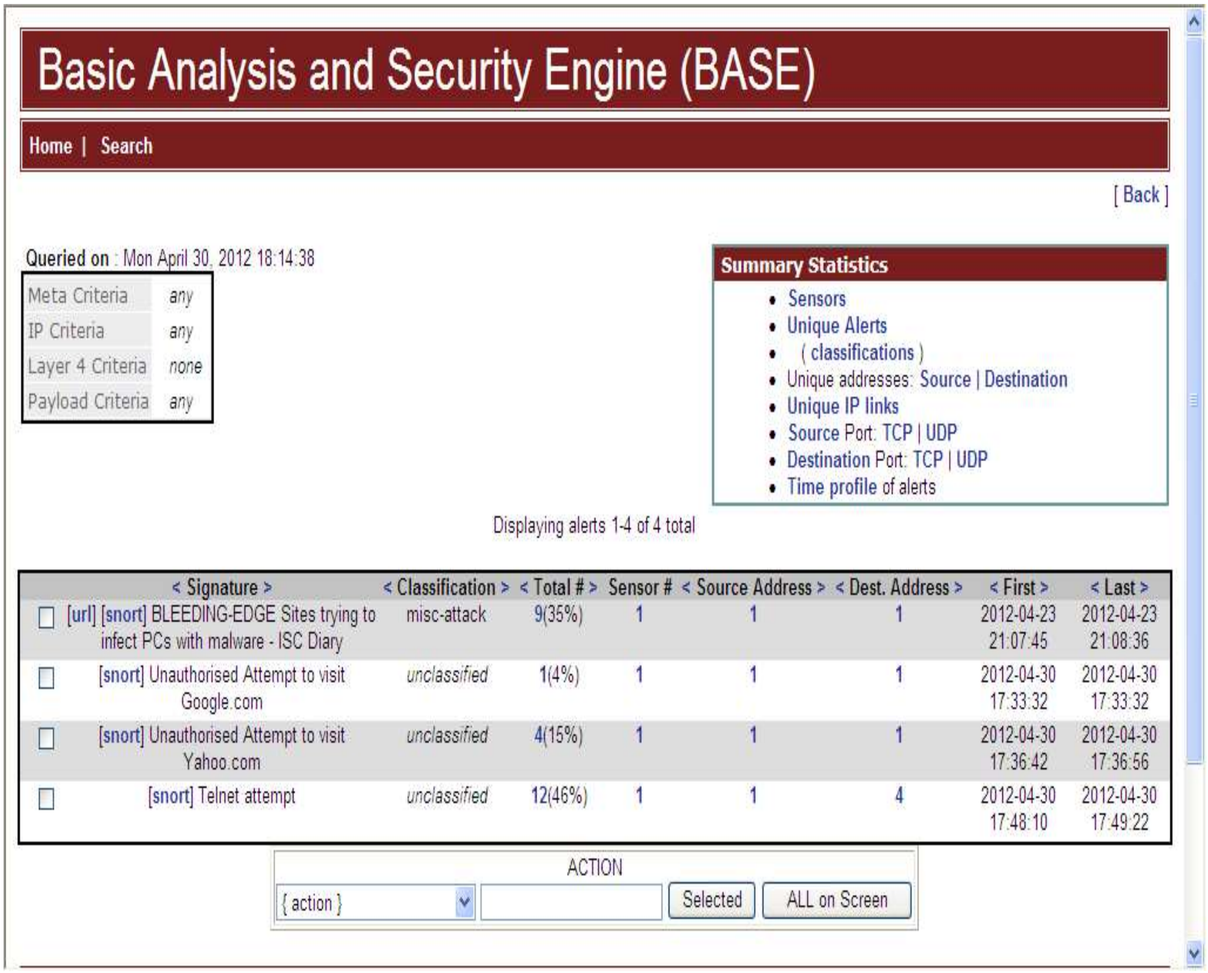
□	ID	< Signature >	< Timestamp >	< Source Address >	< Dest. Address >	< Layer 4 Proto >
□	#0-(2-9)	[url] [snort] BLEEDING-EDGE Sites trying to infect PCs with malware - ISC Diary	2012-04-23 21:08:36	192.168.0.4:1982	209.123.63.168:80	TCP
□	#1-(2-8)	[url] [snort] BLEEDING-EDGE Sites trying to infect PCs with malware - ISC Diary	2012-04-23 21:08:30	192.168.0.4:1982	209.123.63.168:80	TCP
□	#2-(2-7)	[url] [snort] BLEEDING-EDGE Sites trying to infect PCs with malware - ISC Diary	2012-04-23 21:08:27	192.168.0.4:1982	209.123.63.168:80	TCP
□	#3-(2-6)	[url] [snort] BLEEDING-EDGE Sites trying to infect PCs with malware - ISC Diary	2012-04-23 21:08:15	192.168.0.4:1980	209.123.63.168:80	TCP
□	#4-(2-5)	[url] [snort] BLEEDING-EDGE Sites trying to infect PCs with malware - ISC Diary	2012-04-23 21:08:09	192.168.0.4:1980	209.123.63.168:80	TCP
□	#5-(2-4)	[url] [snort] BLEEDING-EDGE Sites trying to infect PCs with malware - ISC Diary	2012-04-23 21:08:06	192.168.0.4:1980	209.123.63.168:80	TCP
□	#6-(2-3)	[url] [snort] BLEEDING-EDGE Sites trying to infect PCs with malware - ISC Diary	2012-04-23 21:07:54	192.168.0.4:1976	209.123.63.168:80	TCP
□	#7-(2-2)	[url] [snort] BLEEDING-EDGE Sites trying to infect PCs with malware - ISC Diary	2012-04-23 21:07:47	192.168.0.4:1976	209.123.63.168:80	TCP
□	#8-(2-1)	[url] [snort] BLEEDING-EDGE Sites trying to infect PCs with malware - ISC Diary	2012-04-23 21:07:45	192.168.0.4:1976	209.123.63.168:80	TCP

Figure 6: Nine (9) alerts generated by Malware attacks

The alerts comprised of these nine different intrusive incidents, which were faithful logs of real incidents that occurred on the network being studied. Most of these attacks consisted of probing of one sort or another, particularly of services with recently reported vulnerabilities. At least one incident involved compromise of an internal computer.

4.4 Results for signature based detection using live traffic

1. We recorded four alerts for violations of miscellaneous signatures.



Queried on : Mon April 30, 2012 18:14:38

Meta Criteria	any
IP Criteria	any
Layer 4 Criteria	none
Payload Criteria	any

Summary Statistics
- Sensors
- Unique Alerts
- (classifications)
- Unique addresses: Source | Destination
- Unique IP links
- Source Port: TCP | UDP
- Destination Port: TCP | UDP
- Time profile of alerts

Displaying alerts 1-4 of 4 total

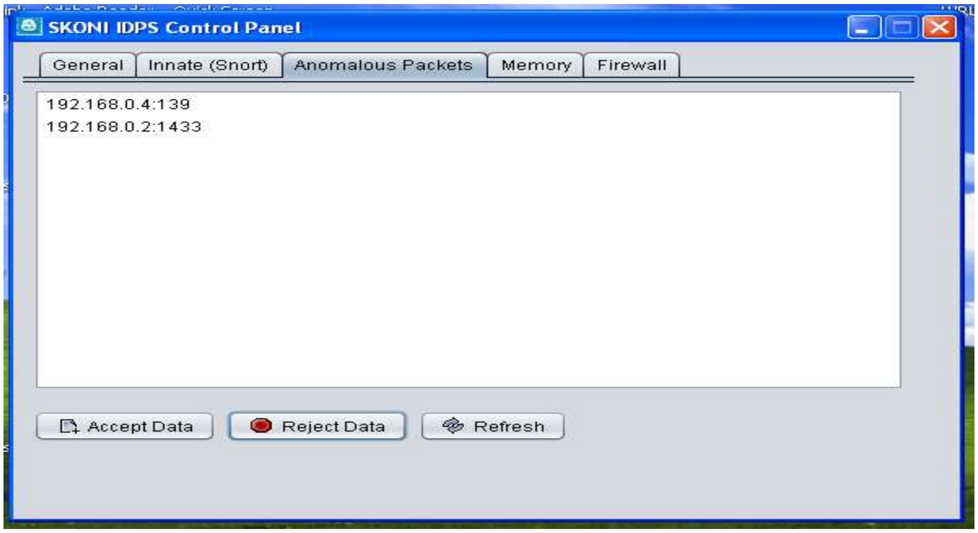
	< Signature >	< Classification >	< Total # >	Sensor #	< Source Address >	< Dest. Address >	< First >	< Last >
□	[url] [snort] BLEEDING-EDGE Sites trying to infect PCs with malware - ISC Diary	misc-attack	9(35%)	1	1	1	2012-04-23 21:07:45	2012-04-23 21:08:36
□	[snort] Unauthorised Attempt to visit Google.com	*unclassified*	1(4%)	1	1	1	2012-04-30 17:33:32	2012-04-30 17:33:32
□	[snort] Unauthorised Attempt to visit Yahoo.com	*unclassified*	4(15%)	1	1	1	2012-04-30 17:36:42	2012-04-30 17:36:56
□	[snort] Telnet attempt	*unclassified*	12(46%)	1	1	4	2012-04-30 17:48:10	2012-04-30 17:49:22

Figure 7: Showing alerts for performing unauthourised activities on hosts

2. We were able to detect nine attacks on our LAN including bogon nets 1, bogon nets 2, bogon nets 3, mysql bot scanning, spambot, SSH, multiple non SMTP server emails, unauthorized attempt to visit a website. This is shown in figure
3. We recorded nine ISC Diary malware attacks.

4.5 Results for anomaly detection Experiment using our LAN

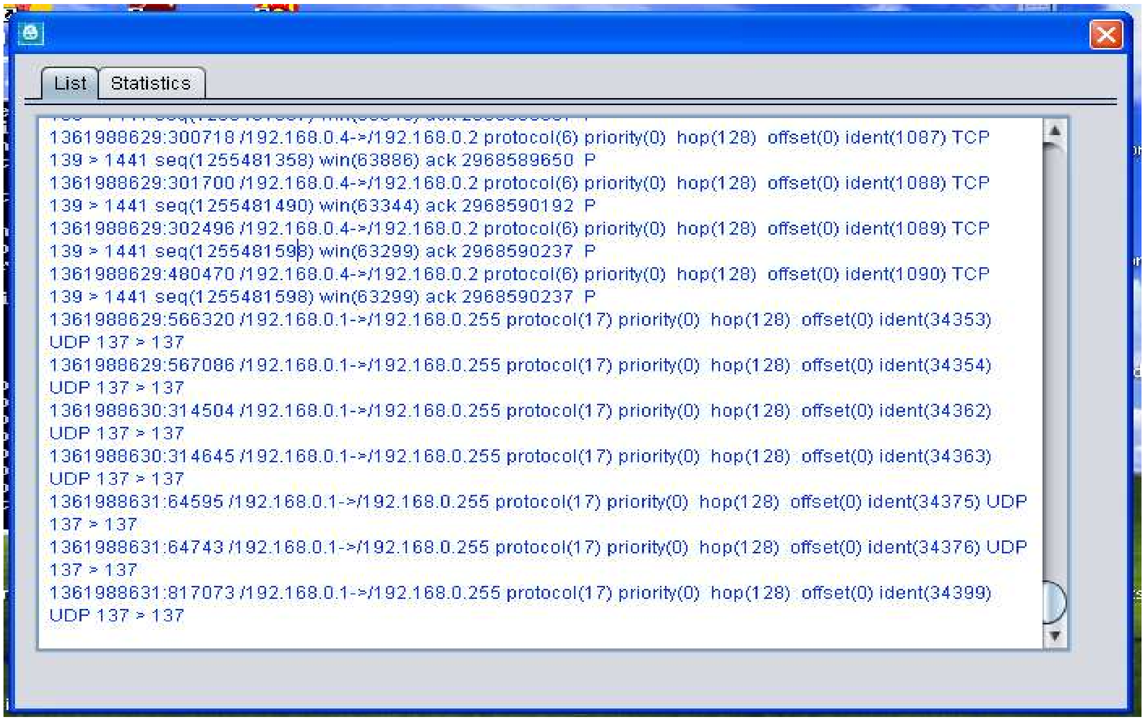
A non-self trace comprised of connections which were not logged by SKONI-HIDS during the data collection cycle and hence not used for training. When a mix of training data and other data were then analyzed by SKONI-HIDS during detection cycle, these connections flagged alerts which was recorded and presented in Figure 8. We observed two of our host computers which received anomalous packets.



```
192.168.0.4:139
192.168.0.2:1433
```

Figure 8: IP address of hosts which received anomalous packets

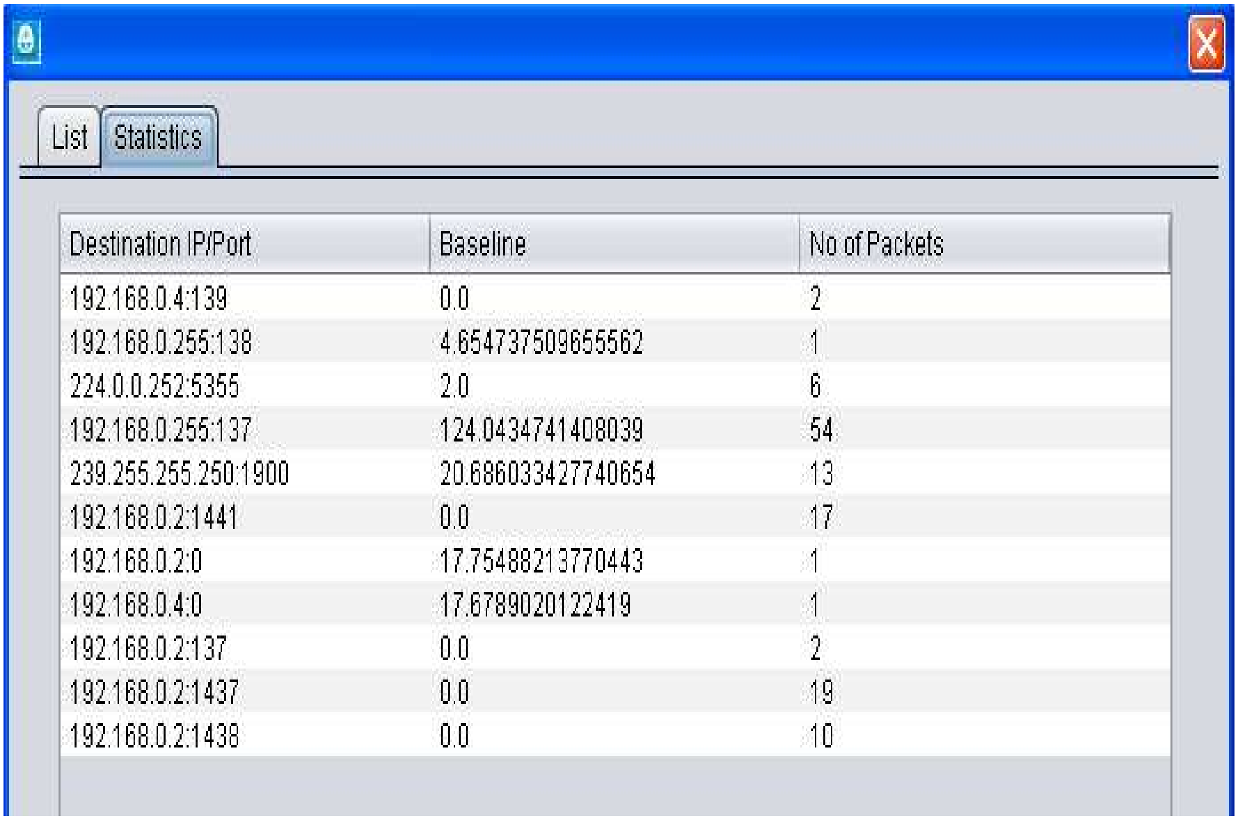
We viewed the list of anomalous packets by clicking on the host's IP and then clicking on list tab. A close investigation will filter out the culprit packets.



```
1361988629:300718 /192.168.0.4->/192.168.0.2 protocol(6) priority(0)  hop(128)  offset(0) ident(1087) TCP
139 > 1441 seq(1255481358) win(63886) ack 2968589650  P
1361988629:301700 /192.168.0.4->/192.168.0.2 protocol(6) priority(0)  hop(128)  offset(0) ident(1088) TCP
139 > 1441 seq(1255481490) win(63344) ack 2968590192  P
1361988629:302496 /192.168.0.4->/192.168.0.2 protocol(6) priority(0)  hop(128)  offset(0) ident(1089) TCP
139 > 1441 seq(1255481598) win(63299) ack 2968590237  P
1361988629:480470 /192.168.0.4->/192.168.0.2 protocol(6) priority(0)  hop(128)  offset(0) ident(1090) TCP
139 > 1441 seq(1255481598) win(63299) ack 2968590237  P
1361988629:566320 /192.168.0.1->/192.168.0.255 protocol(17) priority(0)  hop(128)  offset(0) ident(34353)
UDP 137 > 137
1361988629:567086 /192.168.0.1->/192.168.0.255 protocol(17) priority(0)  hop(128)  offset(0) ident(34354)
UDP 137 > 137
1361988630:314504 /192.168.0.1->/192.168.0.255 protocol(17) priority(0)  hop(128)  offset(0) ident(34362)
UDP 137 > 137
1361988630:314645 /192.168.0.1->/192.168.0.255 protocol(17) priority(0)  hop(128)  offset(0) ident(34363)
UDP 137 > 137
1361988631:64595 /192.168.0.1->/192.168.0.255 protocol(17) priority(0)  hop(128)  offset(0) ident(34375) UDP
137 > 137
1361988631:64743 /192.168.0.1->/192.168.0.255 protocol(17) priority(0)  hop(128)  offset(0) ident(34376) UDP
137 > 137
1361988631:817073 /192.168.0.1->/192.168.0.255 protocol(17) priority(0)  hop(128)  offset(0) ident(34399)
UDP 137 > 137
```

Figure 9: Anomalous packets received by the host.

We also view the statistics of the packets by clicking on the host of interest and click on statistics tab. This action will give the information of the normal profile or 'baseline' of packets which have been encountered by SKONI-HIDS and the number of packets counted as shown in Figure 10.



Destination IP/Port	Baseline	No of Packets
192.168.0.4:139	0.0	2
192.168.0.255:138	4.654737509655562	1
224.0.0.252:5355	2.0	6
192.168.0.255:137	124.0434741408039	54
239.255.255.250:1900	20.686033427740654	13
192.168.0.2:1441	0.0	17
192.168.0.2:0	17.75488213770443	1
192.168.0.4:0	17.6789020122419	1
192.168.0.2:137	0.0	2
192.168.0.2:1437	0.0	19
192.168.0.2:1438	0.0	10

Figure 10: Normal profile and count of the packets

A graph plotted indicating the baseline alongside the count of packets at a glance shows the anomalous packets detected.

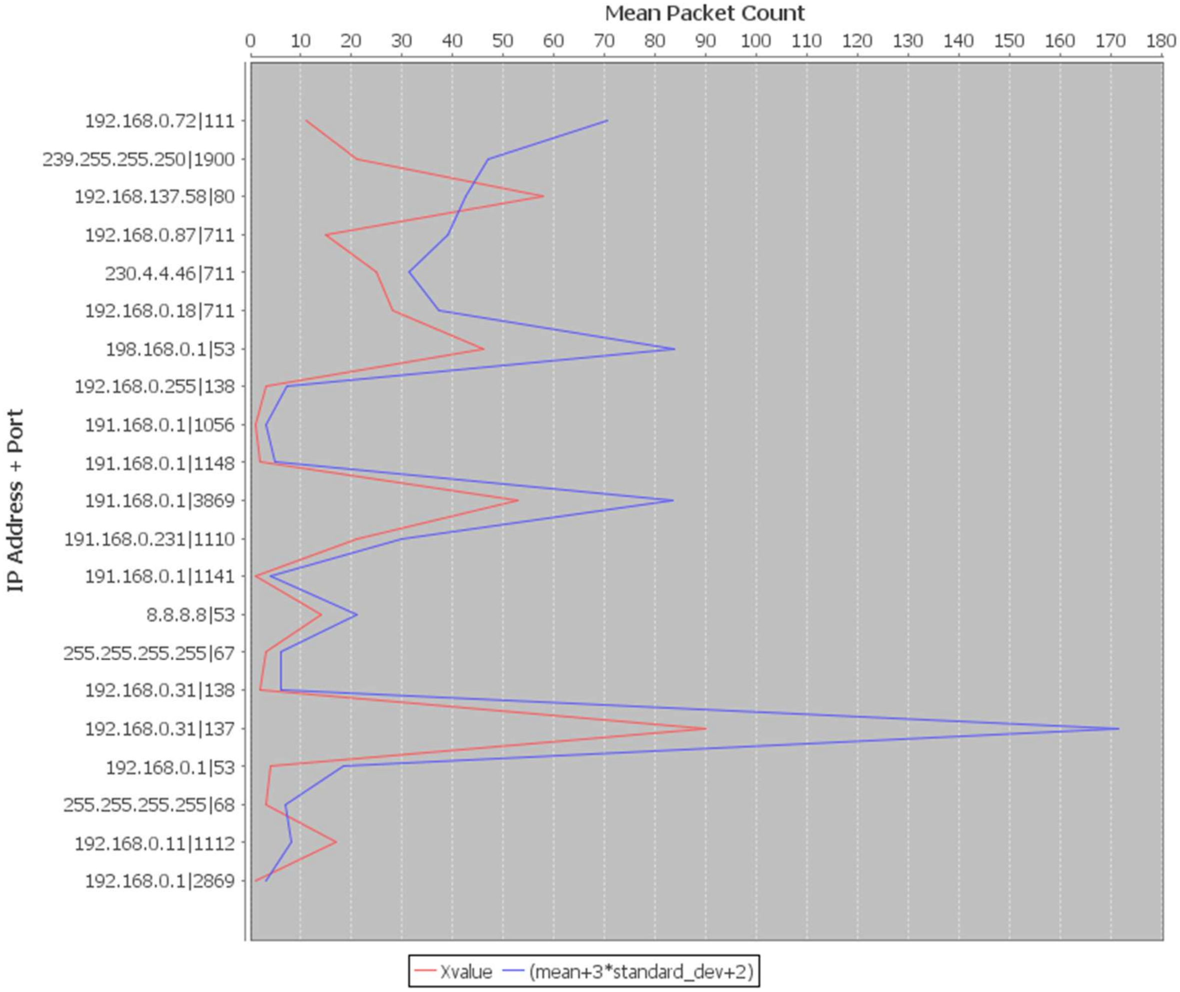
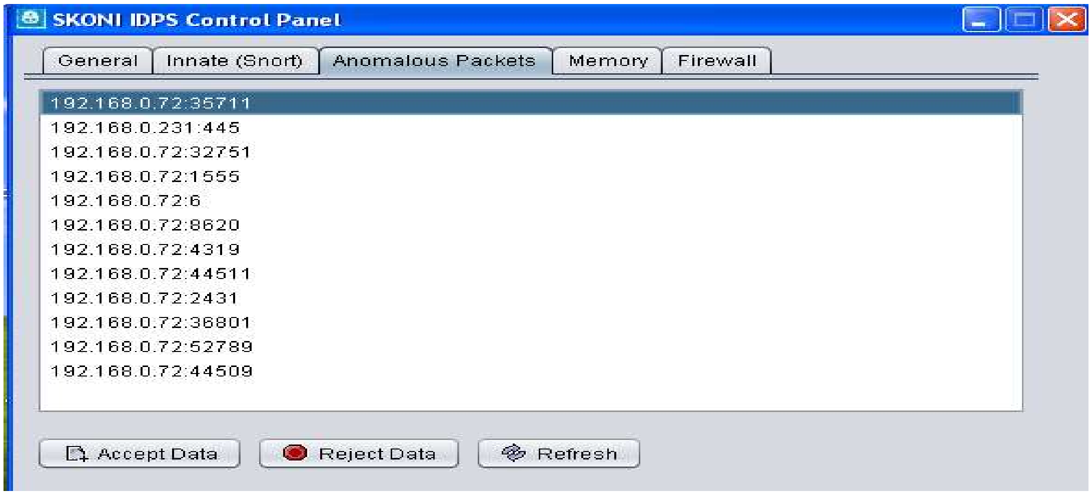


Figure 11: Anomalous packets from 192.168.137.58:80 and 192.168.0.11:1112

4.6 Results for anomaly based detection using live traffic

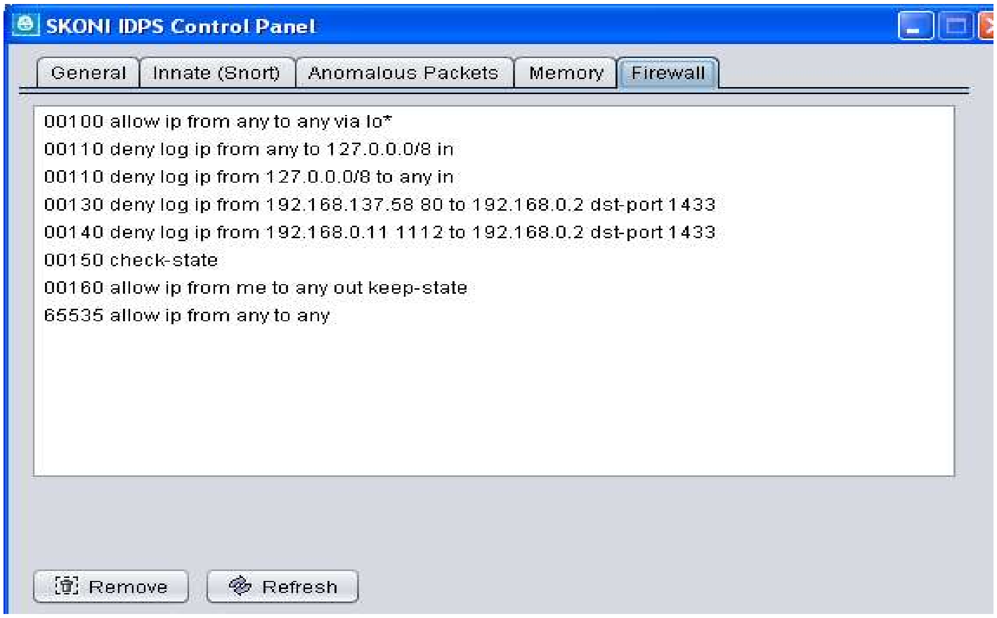
We recorded several instances of anomalous packets from 192.168.0.231:445, 192.168.0.72:32751, 192.168.0.72:1555, 192.168.0.72:6, 192.168.0.72:8620, 192.168.0.72:4319, 192.168.0.71:44511, 192.168.0.72:2431, 192.168.0.72:36801, 192.168.0.72:52789, and 192.168.0.72:44509. This is interpreted as an intense portscan attack.



```
192.168.0.72:35711
192.168.0.231:445
192.168.0.72:32751
192.168.0.72:1555
192.168.0.72:6
192.168.0.72:8620
192.168.0.72:4319
192.168.0.72:44511
192.168.0.72:2431
192.168.0.72:36801
192.168.0.72:52789
192.168.0.72:44509
```

Figure 12: Instance of Anomalous packets

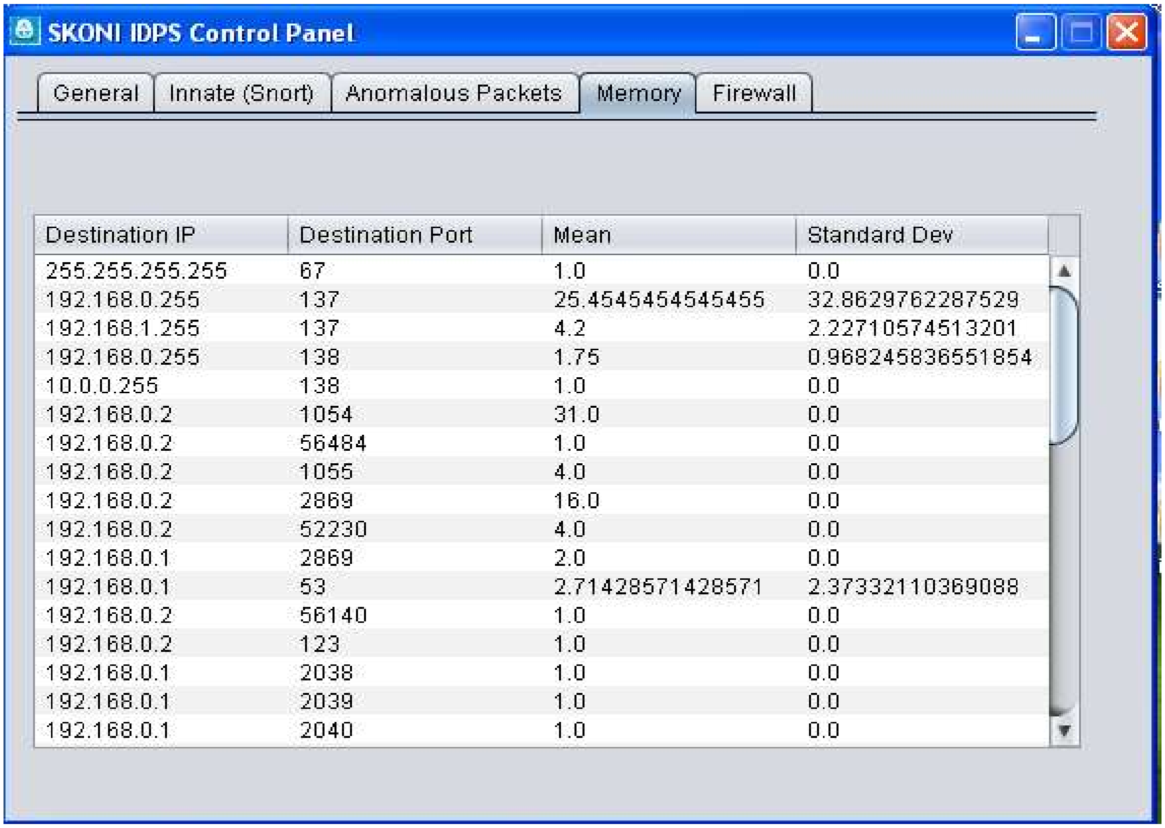
SKONI-HIDS automatically adds anomalous packets detected to the firewall deny rule list. The firewall list can be seen for anomalous packets 192.168.137.58:80 and 192.168.0.11:1112 in figure 13.



```
00100 allow ip from any to any via lo*
00110 deny log ip from any to 127.0.0.0/8 in
00110 deny log ip from 127.0.0.0/8 to any in
00130 deny log ip from 192.168.137.58 80 to 192.168.0.2 dst-port 1433
00140 deny log ip from 192.168.0.11 1112 to 192.168.0.2 dst-port 1433
00150 check-state
00160 allow ip from me to any out keep-state
65535 allow ip from any to any
```

Figure 13: Packets for 192.168.137.58:80 and 192.168.0.11:1112 blacklisted using firewall rules

The baseline (or normal profile) is obtained using the sum of the mean of packet count and three times the standard deviation from the mean and the threshold value used. The values for the mean of packet count and the standard deviation from the mean obtained are recorded in anomaly database memory shown in figure 14.



Destination IP	Destination Port	Mean	Standard Dev
255.255.255.255	67	1.0	0.0
192.168.0.255	137	25.4545454545455	32.8629762287529
192.168.1.255	137	4.2	2.22710574513201
192.168.0.255	138	1.75	0.968245836551854
10.0.0.255	138	1.0	0.0
192.168.0.2	1054	31.0	0.0
192.168.0.2	56484	1.0	0.0
192.168.0.2	1055	4.0	0.0
192.168.0.2	2869	16.0	0.0
192.168.0.2	52230	4.0	0.0
192.168.0.1	2869	2.0	0.0
192.168.0.1	53	2.71428571428571	2.37332110369088
192.168.0.2	56140	1.0	0.0
192.168.0.2	123	1.0	0.0
192.168.0.1	2038	1.0	0.0
192.168.0.1	2039	1.0	0.0
192.168.0.1	2040	1.0	0.0

Figure 14: Mean packets count and the standard deviation from the mean in SKONI-HIDS memory

5. CONCLUSION

SKONI-IDS hybrid IDS presented in this work, was developed and implemented in the Department of Computer Science, University of Benin. In the course of this research work we developed and implemented a hybrid Intrusion Detection System (SKONI-HIDS) for network attacks mitigation. SKONI-HIDS is able to combine signature based detection (using snort rules) and anomaly detection (using statistical analysis) network intrusion detection to form a hybrid network intrusion detection system (SKONI-HIDS). We incorporated an IP firewall to provide active response against intrusion attempts on the network, this enables it to mitigate attacks and also to act as an intrusion prevention system.

Acknowledgements: We would like to acknowledge the management and staff of Kontech Network Research Labs for all their support and the contributions of the Postgraduate Board of Department of Computer Science as well, during the development and deployment of SKONI-HIDS.

Note: The name SKONI is an acronym derived from the names of the authors of this research work, **S**usan **KONI**bere.

REFERENCES

1. Shijoe Jose1, D. Malathi1, Bharath Reddy1 and Dorathi Jayaseeli1 Published under licence by IOP Publishing Ltd Journal of Physics: Conference Series, Volume 1000, National Conference on Mathematical Techniques and its Applications (NCMTA 18) 5–6 January 2018, Kattankulathur, India
2. Murtaza Syed Shariyar, Khreich Wael, Hamou-Lhadj Abdelwahab *and* Gagnon Stephane (*2015) A trace abstraction approach for host-based anomaly detection Computational Intelligence for Security and Defense Applications (CISDA) (2)1-8*
3. *Bukac V., Tucek P and Deutsch M (2012) Advances and Challenges in Standalone Host-Based Intrusion Detection Systems Trust, Privacy and Security in Digital Business. TrustBus ed S. Fischer-Hübner, S. Katsikas and G. Quirchmayr*
4. *Jyothsna V. and Rama Prasad V. V. (2011) A Review of Anomaly based Intrusion Detection Systems International Journal of Computer Applications*
5. Fung Carol (2011). *Collaborative Intrusion Detection Networks and Insider Attacks*. University of Waterloo, Waterloo, ON, Canada. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, volume: 2, number: 1, pp. 63-74
6. Mgabile Tinny, Msiza S. Ishmael, and Dube Erick (2012). *Anomaly Based Intrusion Detection for a Biometric Identification System using Neural Networks* International Conference on Artificial Intelligence and Image Processing (ICAIIP'2012)
7. S. Konyeha and E. A. Onibere (2014a). An anomaly based statistical intrusion Detection Model. Journal of the Nigerian Association of Mathematical Physics.Vol.27, No.1. Pp485-494
8. Susan Konyeha and Onibere E.A (2020) Design of SKONI Hybrid Intrusion Detection Model (SKONI-HIDM). Computing Information System, Development infromatics and allied Research Journal CISDI Vol. 8 No. 4 May, 2020
9. Cole, E., Krutz, R. and Conley, J.W. (2005). *Network Security Bible*, Wiley Publishing Inc.
10. Denning D. E. (1987). *An Intrusion-Detection Model*, IEEE Transaction on Software Engineering, Vol. SE-13, No.2 (Feb), 222-232
11. Modi, C., Patel, D., Patel, H., Borisaniya, B., Patel, A. and Rajarajan, M. (2012). *A survey of intrusion detection techniques in Cloud*. Journal of Network and Computer Applications.
12. S. Konyeha and E. A. Onibere (2014b). Mitigating Network Attacks using SNORT IDS. Journal of the Nigerian Association of Mathematical Physics. Vol.27, No 1.Pp 479-484