Architecture for Real Time Monitoring and Modeling of Network Behavior for Enhanced Security

M.Ambika
PG Scholar
Dept of IT
Bannari Amman Institute of Technology,
Sathyamangalam

R.V.Nataraj
Professor
Dept of IT
Bannari Amman Institute of Technology,
Sathyamangalam

ABSTRACT

Network security is a rapidly growing and is the major area of concern for every network. Firewalls are used as a security check point in network environment even then different types of security issues keep on emerging. In order to protect the network from illegal access, the concept of Intrusion Detection System (IDS) and Intrusion Prevention System (IPS) is been developed. An IDS is a system where the events occurring in a network is monitored and analyzed for identifying the sign of possible incidents. These incidents may either be violations or the threats that are about to happen violating the computer security policies or standard security policies. Java programming language is made use for developing the Intrusion Detection System. JPCap package is used along with winpcap for developing the traffic monitoring system. The network packets are captured online i.e., as they come across the interface of the network. The IDS is aimed to provide the preliminary level of detection techniques so as to secure the systems present in the networks.

Keywords

Firewalls, Intrusion Detection System (IDS), Intrusion Prevention System (IPS), Winpcap, Jpcap.

1. INTRODUCTION

A computer network or network is a collection of computers and various other hardware components which are interconnected through communication channels for sharing resources and information. The devices that are connected within the network are referred to as nodes. In this paper Local Area Networks (LANs) are concentrated and the network traffic is being monitored for identifying any case of intrusions.

LANs are deployed within the organizations, educational institutions etc., which mainly focuses on resource sharing and establish effective communication among the nodes that are present within the network. The LANs are established with a wide range of security policies. Even then the highly sophisticated hacking attacks are considered as a threat in LAN. Firewall is the basic level of security measures that was developed in the earlier stages. As the techniques for detecting the intrusion increases, the number of attacks also gets increased. A sample local area network includes a number of nodes connected together for sharing the resources or information. Any type of network is susceptible to malicious attacks.

The initial level of security for the nodes within the network is provided by the firewalls, antivirus software etc., but at times these security measures itself becomes a loop hole for the attackers who try to break into the system. The following figure shows a simple LAN environment:

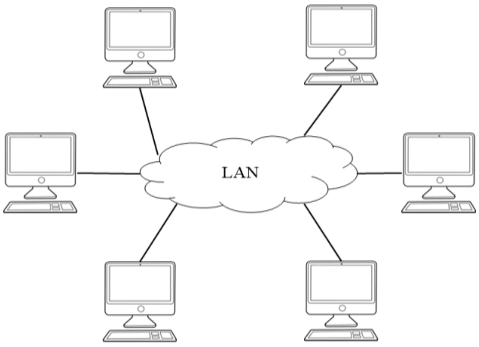


Fig 1: Local Area Network Scenario

LAN mainly focuses on establishing a inter communication environment within any organization and educational Institutions. The information that is communicated within the organizations is of more sensitive type, when compared to that of the educational institutions. Hence forth it is very important for securing the data.

Securing data LAN starts with monitoring the traffic along the network and obtain the pattern of the network traffic. The network packets are the primary information considered by almost all the intrusion detection systems. There are various network monitoring tools available as .exe file. But the proposed monitoring system provides the advantage of monitoring the traffic effectively and makes use of those patterns dynamically.

The rest of the paper is organized as follows: Section 2 gives an overview about various threats that remains as an overhead in LAN environment. Section 3 comes out with various existing methods that are available for monitoring the traffic in a network. Section 4 gives the proposed scheme that is been developed using JPCap java packets. Section 5 concludes the article with the future enhancements.

2. MALICIOUS ATTACKS IN LANS

The lack of proper security measures leads to variety of attacks in a network and paves way for the intruders to take hold of the legitimate information or user accounts. The LANs are vulnerable to any of the following types of attacks if any lag in the security measures.

2.1 Eavesdropping

A large percentage of networks employ a clear text format and an unsecured way of communication, which paves way for the attackers who has gained an illegal access to data path in the network to listen or interpret the traffic. Eavesdropper monitors the network and there will not be any sign of presence about these attackers. This is generally a hectic and challenging problem faced by an administrator in any enterprise.

2.2 Data Modification

Once the attacker gets hold of the legitimate data, obviously the next step is to alter that data. The modification of data is done without the knowledge of the sender or the receiver. Even though confidentiality is not required for all communications, it is mandatory to ensure that the messages are not modified.

2.3 Identity Spoofing (IP Address Spoofing)

In most networks IP address of a computer is used to identify the legitimate entity in the network. But in some cases there are chances where an IP address could be falsely assumed, which is referred to as identity spoofing. The attacker gets the opportunity to construct the IP packets through some special programs and these IP packets appear to originate from valid addresses inside the corporate intranet. Once the attacker gain access to the network with the help of a valid IP address, the attacker can reroute the data, and there are possibilities for the data to be deleted.

2.4 Password-Based Attacks

The traditional method of providing access rights to the users is by providing username and password. In case the attacker gets to know the information about the valid user account then the attacker also gets the same level of rights as that of the legitimate user. This attack helps the attacker easily modify the server and network configurations, including the access controls and routing tables.

2.5 Sniffer Attack

The term sniffer refers to the application or a device that has the capability to read, monitor and capture network data exchanges. Sniffer provides a full view of the data inside the packet, if the packets are not encrypted. This application helps the attacker to analyze the network and gain information to eventually cause the network to crash or to become corrupted.

3. LITERATURE SURVEY

The above section clearly states the threats that might happen in LAN communication. Considering these criteria it was necessary to develop an intrusion detection and prevention systems. Following are some of the existing works that provides various intrusion detection and prevention system (IDPS).

The issues of information security are addressed in [1]. The paper also describes the security needs of an organization for protecting their critical information from attacks. Highly trained analysts are required for continuously monitoring the system. Construction of new security strategies involve high effort which are discussed in [2], [9] and [12].

A multilayer approach in intrusion detection and prevention system to monitor a single host is provided in [2]. This approach consists of three layers: File Analyzer, System Resource Analyzer and Connection Analyzer. The advantage of this technique [2] is that it provides both signature based and anomaly based detection and prevention. But on the other hand it requires a large amount of memory for storing system data and network traffic.

A software based solution, Proventia desktop is proposed in [3] which helps in detecting and protecting the system starting from network layer up to application layer from known and unknown attacks. This software provides a great flexibility to set different type of filtering rules. The major drawback is its high rate of false-positives. Highly skilled staff and also a lot of time are required to monitor the IDPS.

In the proposed model in [4], the IDPS is divided is divided into two types, as in-source and out-source. The key business of Managed Security Services Provider (MSSP) [4] is to provide security to an organization against attacks. Most of the time spent by MSSP includes examining new technology to secure the information even better than before.

Snort and source fire are considered as the best intrusion prevention systems for a multinational company in [5]. Snort provides a facility to modify its source code with the help of source fire. The disadvantage of snort is that it uses signature based technique for detecting the intrusion. In case any anomaly behaviour occur then it is not possible for snort to detect that attack.

A mobile agent based IDPS is proposed in [6]. The secure mobile agent is responsible for monitoring the system, processing the logs, detecting the attacks and protecting the host by automated real time response. The major drawback in this technique [6] is that if mobile agent is the target for the attackers then it becomes tedious for any IDPS to prevent from the system being hacked. Hence a separate security infrastructure is necessary for protecting the mobile agent.

David and Paola proposed a technique in [7] that shows the interaction of application with the operating system and also examines how an IDS could be broken without any detection, by using the sequence matching technique. But this technique is not aware of the effort and knowledge that is required for producing such an attack. It is also unaware of the prediction of the working of IDS by the attackers.

The differences in host based and network based intrusion detection and prevention system is been proposed in [8]. Two types of network intrusion detection systems are proposed in this paper: Promiscuous-mode and Network mode. This IDS responds only to the signature based detected attacks which emerged as a major drawback of this method and still human interaction is needed to solve the issue.

A novel string matching technique is proposed in [9] which is an optimization of other matching algorithms. The proposed string matching algorithm breaks the string into small sets of state machines. The subset of string is recognized by the state machine. If there is any presence of the suspicious behaviour the information about the intruder is broadcasted to every module which holds the database for defining the rules and the signature of the intruder is compared with the predefined detected signatures. This algorithm very efficient, ten times faster than the other existing algorithms and also consumes less resources.

Distributed IDS is used to analyze the system according to Mrdović and Zajko [10], where multiple sensors are deployed in selected network segments for observing the network traffic behaviour. The analysis engine used here is snort. The log events are stored in MySQL. The distributed IDS is managed by management console that is responsible for monitoring and configuring the IDS. A greater protection against attacks is provided by this IDS. It is made by possible with the help of multiple computers monitoring and preventing the network from malicious attacks [13]. The implementation part of this system requires large memory and well trained security analysts and also management is difficult.

This paper [11] brings out the two different techniques of IDS: Misuse detection and anomaly detection. Data mining, data fusion and immunological based approaches are used in IDS. Brief information about the existing intrusion detection technology is given in this paper. The challenges and future directions of intrusion detection technology are evaluated. The techniques discussed in [4], [9] & [14] are sufficient for IDPS to detect and respond to anomalies in real time.

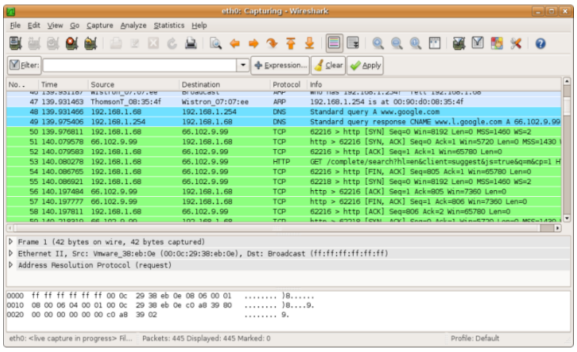
The technique proposed in [12] combines multiple hosts for detecting multiple intrusions and detecting the false-positive rate. The system call events are modeled using Hidden Markov Model which is a speech recognition technique. Decision trees help to model or classify the intrusion type and examine the future challenges. Less false-positive rate is incurred in this technique [12] that increases the performance of the detection. The system can be secured in a better way if these IDS adopt the mechanism of protection that was in discussed in [4] and [8].

Another approach for IDPS that uses peer to peer approach for security is INDRA (Intrusion Detection and Rapid Action) [13]. This network works in a distributed environment by distributing the intruder's information on peer to peer network. If any interrupt is found by INDRA then the security agent cut off the effected packets. It is the most reliable, trusted and efficient IDPS. As discussed in [2] and [3] it requires a large amount of memory to store all the collected information about the intruder.

Architecture to protect HIDS is proposed in [14]. It includes a virtual machine that observes the system behaviour or monitors the system inside a virtual machine. Multiple virtual machines can be made to run simultaneously on same hardware. Cost effectiveness is a great advantage as discussed in [2] and [3].

IDPS and IDS/IPS tools are investigated by Matt and Andrew in [15]. The schema is created using MySQL and the setting of permissions are configured. The changes in specific files are monitored using TRIPWARE software and the logs are checked continuously through SNORT. A large number of different attacks such as viruses, Denial of Services, malware etc., can be detected with the help of SNORT.

Apart from the above mentioned tools and techniques there are various other sources available for performing the packet capture action. Wireshark is an efficient tool for capturing and analyzing the network packets. This tool is used for network troubleshooting, for communications protocol development and for education.



```
No. .  Time        Source            Destination      Protocol  Info
 47 139.931463 ThomsonT_08:35:4f  Wistron_07:07:ee  ARP   192.168.1.254 is at 00:90:d0:08:35:4f
 48 139.931466 192.168.1.68       192.168.1.254     DNS   Standard query A www.google.com
 49 139.975406 192.168.1.254      192.168.1.68      DNS   Standard query response CNAME www.l.google.com A 66.102.9.99
 50 139.976811 192.168.1.68       66.102.9.99       TCP   62216 > http [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=2
 51 140.079578 66.102.9.99        192.168.1.68      TCP   http > 62216 [SYN, ACK] Seq=0 Ack=1 Win=5720 Len=0 MSS=1430
 52 140.079583 192.168.1.68       66.102.9.99       TCP   62216 > http [ACK] Seq=1 Ack=1 Win=65780 Len=0
 53 140.080278 192.168.1.68       66.102.9.99       HTTP  GET /complete/search?hl=en&client=suggest&js=true&q=m&cp=1 H
 54 140.086765 192.168.1.68       66.102.9.99       TCP   62216 > http [FIN, ACK] Seq=805 Ack=1 Win=65780 Len=0
 55 140.086921 192.168.1.68       66.102.9.99       TCP   62218 > http [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=2
 56 140.197484 66.102.9.99        192.168.1.68      TCP   http > 62216 [ACK] Seq=1 Ack=805 Win=7360 Len=0
 57 140.197777 66.102.9.99        192.168.1.68      TCP   http > 62216 [FIN, ACK] Seq=1 Ack=806 Win=7360 Len=0
 58 140.197811 192.168.1.68       66.102.9.99       TCP   62216 > http [ACK] Seq=806 Ack=2 Win=65780 Len=0

▷ Frame 1 (42 bytes on wire, 42 bytes captured)
▷ Ethernet II, Src: Vmware_38:eb:0e (00:0c:29:38:eb:0e), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▷ Address Resolution Protocol (request)

0000  ff ff ff ff ff ff 00 0c  29 38 eb 0e 08 06 00 01   ........ )8......
0010  08 00 06 04 00 01 00 0c  29 38 eb 0e c0 a8 39 80   ........ )8....9.
0020  00 00 00 00 00 00 c0 a8  39 02                     ........ 9.

eth0: <live capture in progress> Fil...   Packets: 445 Displayed: 445 Marked: 0   Profile: Default
```

Fig 2: Snapshot of Wireshark tool capturing live packets

Wireshark uses Pcap to capture the packets and hence captures the packets on the types of networks that pcap supports. Data can either be captured from a live network or can be read from a file that contains already captured packets.

4. PROPOSED WORK

An intrusion is someone (hacker or cracker) attempting to break into or misuse the system. The term misuse may include stealing any confidential data or something minor such as misusing the email system for spam. With the rapid development of Internet and the World Wide Web, the intrusion rate also gets increased. Apart from these advantages there is an arms race between the intruders and people who provide security to the systems in networks. The proposed IDS monitoring system is designed in such a way that it runs on the host machines and assists the network administrators in order to detect the intrusion attacks and inform the user of the system to provide security by blocking the malicious users based on their IP addresses.

4.1 System Architecture

The system architecture for monitoring the system network is given below:

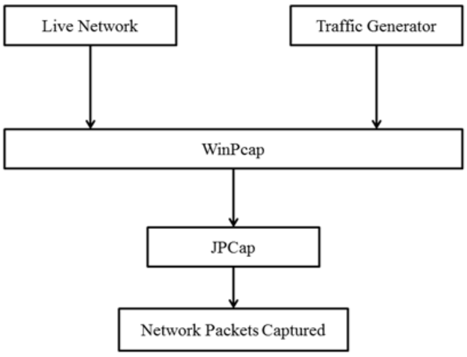


Fig 3: Architecture of Network Monitoring System

The live network packets or the packets or the packets generated by the traffic generator are captured with the help of WinPcap and JPCap. The packets can be processed directly or it can be saved in a file for further processing. JPCap is a java package used for capturing and displaying the packets.

4.2 Network Packet Capture

Once the packets arrive at the system, they are sniffed by the Sniffer and then various processing techniques are applied to detect if any attack occurs and in that case warning is given to the user. JPCap package is used in the java code developed for capturing the network packets.

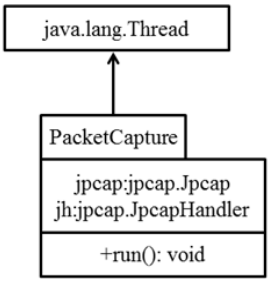
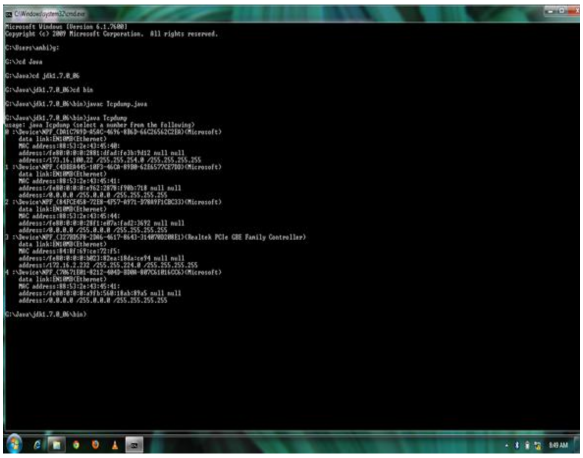


Fig 4: UML diagram showing Packet Capture

The following steps are carried out for capturing the network packets:

Step 1: Obtain the list of Network Interfaces

The first and the foremost task that has to be performed before starting to capture the packets are to identify the list of network interface on the machine. The network interface JPCap provides JpcapCaptor.getDeviceList() method. This method returns an array of network interface objects. Following screenshot displays the network interface details:



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\anbi>g:

G:\>cd Java

G:\Java>cd jdk1.7.0_06

G:\Java\jdk1.7.0_06>cd bin

G:\Java\jdk1.7.0_06\bin>javac Tcpdump.java

G:\Java\jdk1.7.0_06\bin>java Tcpdump
usage: java Tcpdump <select a number from the following>
0 :\Device\NPF_{DA1C7A9D-A5AC-4696-8B6D-66C26562C2EA}(Microsoft)
    data link:EN10MB(Ethernet)
    MAC address:00:53:2e:43:45:40:
    address:/fe80:0:0:0:2881:dfad:fe3b:9d12 null null
    address:/173.16.100.22 /255.255.254.0 /255.255.255.255
1 :\Device\NPF_{4DBEA445-10F3-46CA-89B0-62E6577CE7DD}(Microsoft)
    data link:EN10MB(Ethernet)
    MAC address:00:53:2e:43:45:41:
    address:/fe80:0:0:0:e962:2878:f90b:718 null null
    address:/0.0.0.0 /255.0.0.0 /255.255.255.255
2 :\Device\NPF_{84FCE458-72E8-4F57-A971-D7889F1CBC33}(Microsoft)
    data link:EN10MB(Ethernet)
    MAC address:00:53:2e:43:45:44:
    address:/fe80:0:0:0:28f1:e07a:fad2:3692 null null
    address:/0.0.0.0 /255.0.0.0 /255.255.255.255
3 :\Device\NPF_{3278D5F0-2DA6-4617-8643-31407802008E1}(Realtek PCIe GBE Family Controller)
    data link:EN10MB(Ethernet)
    MAC address:84:8f:69:ce:72:f5:
    address:/fe80:0:0:0:b023:82ea:18da:ce94 null null
    address:/172.16.2.232 /255.255.224.0 /255.255.255.255
4 :\Device\NPF_{70671E01-0212-404D-BD0A-007C61016CC6}(Microsoft)
    data link:EN10MB(Ethernet)
    MAC address:00:53:2e:43:45:41:
    address:/fe80:0:0:0:a9fb:560:18ab:89a5 null null
    address:/0.0.0.0 /255.0.0.0 /255.255.255.255

G:\Java\jdk1.7.0_06\bin>
```

Fig 5: List of Network Interfaces

The network interface object contains information about the corresponding network interface, like name, description, IP and MAC addresses, datalink name and description.

Step 2: Opening a Network Interface
After obtaining the list of network interfaces, one particular network interface is chosen for capturing the packets. This process can be performed by using JpcapCaptor.openDevice() method. This method returns an instance of JpcapCaptor. Further it is possible to call several other methods of the class JpcapCaptor for capturing packets from network interface.

Step 3: Capture Packets from the Network Interface
As the JpcapCaptor instance has been obtained, the next immediate step is to capture the packets from the network interface. The steps 1 and 2 explained above are considered as the basic requirements that are to be set up before capturing the packets from the network interface. Two major methods can be applied for capturing packets using JpcapCaptor instance. First is the callback method and the second is to capture the packets one-by-one. Following diagram explains the steps involved in packet capture:

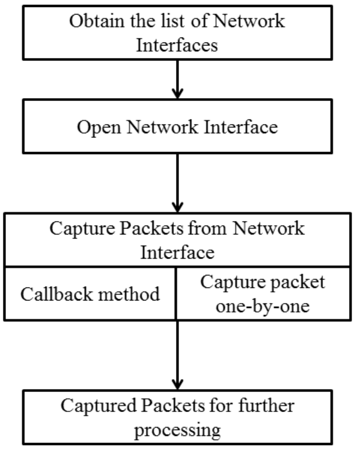
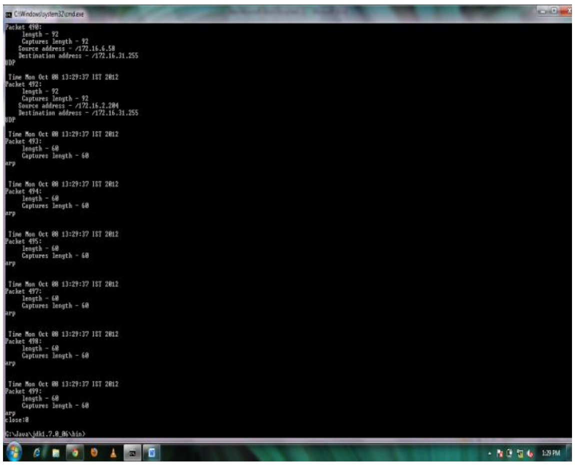

Obtain the list of Network Interfaces
Open Network Interface
Capture Packets from Network Interface
Callback method
Capture packet one-by-one
Captured Packets for further processing

Fig 6: Steps involved in Packet Capture

Callback method is similar to recursive methods where a particular method is called recursively for capturing the packets. For many applications callback method may not sound better. In that case the packets are captured one-by-one unlike the callback method where the packets are captured continuously.



```
C:\Windows\system32\cmd.exe
Packet 490:
     length - 92
     Captures length - 92
    Source address - /172.16.6.58
    Destination address - /172.16.31.255
UDP

 Time Mon Oct 08 13:29:37 IST 2012
Packet 492:
     length - 92
     Captures length - 92
    Source address - /172.16.2.204
    Destination address - /172.16.31.255
UDP

 Time Mon Oct 08 13:29:37 IST 2012
Packet 493:
     length - 60
     Captures length - 60
arp


 Time Mon Oct 08 13:29:37 IST 2012
Packet 494:
     length - 60
     Captures length - 60
arp


 Time Mon Oct 08 13:29:37 IST 2012
Packet 495:
     length - 60
     Captures length - 60
arp


 Time Mon Oct 08 13:29:37 IST 2012
Packet 497:
     length - 60
     Captures length - 60
arp


 Time Mon Oct 08 13:29:37 IST 2012
Packet 498:
     length - 60
     Captures length - 60
arp


 Time Mon Oct 08 13:29:37 IST 2012
Packet 499:
     length - 60
     Captures length - 60
arp
close:0

G:\Java\jdk1.7.0_06\bin>
```

Fig 7: Network Packets Captured

5. CONCLUSION AND FUTURE WORK

This paper proposes a monitoring for dynamically capturing the packets in a network. The IDS is designed to provide the basic detection techniques so as to secure the systems present in the networks. IDS help the Network Administrator to track down intruders on the Internet whose very purpose is to bring your network to a breach point and make it vulnerable to attacks. Security organizations will have to adopt such a strongest model or mechanism which provides strongest protection against threats to ensure that the system will remain secure. As the first step in intrusion detection this proposed system provides an efficient way of capturing packets in the network. The code is completely written in java and tested Windows XP and Windows 7 operating system.

The present system just displays the log information but doesn't employ any techniques to analyze the information present in the log records and extract knowledge. The system can be extended by incorporating Data Mining techniques to analyze the information in the log records which may help in efficient decision making.

6. REFERENCES

[1] Ahmed Patel, Qais Qassim, Christopher Wills. "A survey of intrusion detection and prevention systems", Information Management & Computer Security Journal, Vol.3, Page: 64-69, 2010.

[2] Oludele Awodele, Sunday Idowu, Omotola Anjorin, and Vincent J. Joshua, "A Multi-Layered Approach to the Design of Intelligent Intrusion Detection and Prevention System (IIDPS)", Babcock University, Volume 6, Page: 20-26, 2009.

[3] Mario Guimaraes, Meg Murray. "Overview of Intrusion Detection and Intrusion Prevention", Information security curriculum development Conference by ACM Page: 39-44, 2008.

[4] Muhammad Awais Shibli, Sead Muftic. "Intrusion Detection and Prevention System using Secure Mobile Agents", IEEE International Conference on Security & Cryptography, Page: 58-67, 2008.

[5] David Wagner, Paolo Soto. "Mimicry Attacks on Host Based Intrusion Detection Systems", 9th ACM Conference on Computer and Communications Security, Page: 27-35, 2002.

[6] Harley Kozushko. “Intrusion Detection: Host-Based and Network-Based Intrusion Detection Systems”, Babcock University, Volume 6, Page: 20-26, 2003

[7] Lin Tan, Timothy Sherwood. “A High Throughput String Matching Architecture for Intrusion Detection and Prevention”, Proceedings of the 32nd Annual International Symposium on Computer Architecture (ISCA 2005).

[8] S. Mrdovic, E. Zajko. “Secured Intrusion Detection System Infrastructure”, University of Sarajevo/Faculty of Electrical Engineering, Sarajevo, Bosnia and Herzegovina, ICAT, Page: 49-57, 2005.

[9] Yeubin Bai, Hidetsune Kobayashi. “Intrusion Detection Systems: technology and Development”, 17th International Conference of Advanced Information Networking and Applications, (AINA 2003).

[10] M. Laureano, C. Maziero1, E. Jamhour. Protecting “Host-Based Intrusion Detectors through Virtual Machines”, The International Journal of Computer and Telecommunications Networking, Vol.9, Page: 24-31, 2007.networks”. IEEE Transactions on Mobile Computing 2010;9(July):913–26.

[11] “Host Intrusion Prevention Systems and Beyond”, SANS Institute (2008).

[12] “Intrusion Detection and Prevention In-sourced or Out-sourced”, SANS Institute (2008).

[13] Matt Carlson and Andrew Scharlott. “Intrusion detection and prevention systems”, (2006).

[14] Sang-Jun Han and Sung-Bae Cho. “Combining Multiple Host-Based Detectors Using Decision Tree”, Australian Joint Artificial Intelligence Conference, (AUSAI 2003).

[15] Ramaprabhu Janakiraman, Marcel Waldvogel, Qi Zhang. Indra: “A peer-to-peer approach to network intrusion detection and prevention”, Enabling Technologies: Infrastructure for Collaborative Enterprises, WET ICE 2003.