A BRIEF STUDY AND COMPARISON OF, OPEN SOURCE INTRUSION DETECTION SYSTEM TOOLS

[1]SURYA BHAGAVAN AMBATI, [2]DEEPTI VIDYARTHI

[1,2]Defence Institute of Advanced Technology (DU) Pune –411025
Email: Surya2622@gmail.com, deepti@diat.ac.in

Abstract - As the world becomes more connected to the cyber world, attackers and hackers are becoming increasingly sophisticated to penetrate computer systems and networks. Intrusion Detection System (IDS) plays a vital role in defending a network against intrusion. Many commercial IDSs are available in marketplace but with high cost. At the same time open source IDSs are also available with continuous support and upgradation from large user community. Each of these IDSs adopts a different approaches thus may target different applications. This paper provides a quick review of six Open Source IDS tools so that one can choose the appropriate Open Source IDS tool as per their organization requirements.

Keywords - Intrusion Detection, Open Source IDS, Network Securit, HIDS, NIDS.

I. INTRODUCTION

Every day, intruders are invading countless homes and organisations across the country via virus, worms, Trojans, DoS/DDoS attacks by inserting bits of malicious code. Intrusion detection system tools helps in protecting computer and network from a numerous threats and attacks. Intrusion Detection System (IDS) is useful in monitoring network or host activities for malicious activities or policy violations. Various Open Source IDS tools are available for the users. Working on these tools is based on different approaches, making them suitable for different applications. This paper confers about the methodology, advantages and disadvantages of six Open Source Intrusion Detection tools Snort, Bro, OSSEC, AIDE, Tripwire and Samhain . Hence it become helpful in choosing an Open Source Intrusion Detection System that best suits the organization and it will also help those who want to experiment with intrusion detection tools. This paper is organized as follows: Section II discusses about the basics of Intrusion detection while Section III presents six open source intrusion detection system tools while Section IV discuss and compare open source intrusion detection tools.

II. INTURSION DETECTION

An intrusion occurs when an attacker attempts to gain entry into or disrupt the normal operations of an information system, almost always with the intent to do harm. [1]. IDS is one of the important measures to mitigate computer network/host intrusions. IDSs focus not only on the detection of abnormal activities in computer networks, but also determining whether such activities are malicious or not. There are basically two types of IDS, namely, host-based IDS (HIDS) and network-based IDS (NIDS). HIDS concentrate on the activities in a host without considering the activities in the computer networks. On the other hand, NIDS put its focus on computer networks without examining the hosts' activities.
Intrusion Detection methodologies can be classified as Signature based detection, Anomaly based detection and Stateful Protocol analysis based detection [2]. Signature based detection approach detects only known threats. Whenever there is a novel type of intrusion, the signatures of the IDS has to be updated. Anomaly based detection is the process of comparing definitions of activities which are supposed to be normal against observed events to identify deviations. Stateful Protocol analysis is the process of comparing predetermined profiles of generally accepted definitions of benign protocol activity for each protocol state against observed events to identify deviations. Both inline and passive technologies can be implemented for HIDS and NIDS. An inline IDS is able to prevent further damages on computer network if network intrusions are detected. Conversely, a passive IDS only records the intrusive activities without taking any further action to reduce the damages done by intruders. The broad classification of IDS is shown in fig. 1

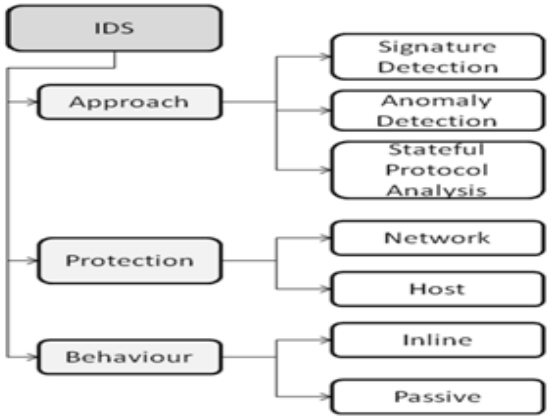

Fig. 1. IDS classification

III. OPEN SOURCE INTRUSION DETECTION TOOLS

There are many open source IDS tools are available in open space, but in this paper our analysis is restricted to two popular NIDS tools Snort and Bro & four HIDS tools OSSEC, Tripwire, AID and Samhain.

SNORT

Snort is an open source network intrusion prevention and detection system (IDS/IPS) combining the benefits of signature, protocol, and anomaly-based inspection [16]. Snort was originally written by Martin Roesch integrated enterprise versions with purpose built hardware and commercial support services are provided by Sourcefire which was then acquired by Cisco October 7, 2013. [7] Snort can be configured in three different modes namely inline, tap (passive) and inline-test. When Snort is in Inline mode, it acts as an IPS allowing drop rules to trigger, in Passive mode, it acts as a IDS i.e Drop rules are not loaded and in Inline-Test mode simulates the inline mode of snort, allowing evaluation of inline behaviour without affecting traffic. The drop rules will be loaded and will be triggered as a Wdrop (Would Drop) alert. Snort is capable of performing real-time traffic analysis, packet logging, alerting and blocking on IP networks. It performs protocol analysis, content searching, and content matching. Snort can also be used to detect probes or attacks, operating system fingerprinting attempts, common gateway interface (CGI) attacks, buffer overflows, server message block (SMB) probes, and stealth port scans. [16]

Snort Architecture
Snort architecture consists of mainly 7 modules

1) Packet Capture Module: This module gathers packets from network adapter. It is based on the libpcap library for Unix like systems and for windows systems WinPacap is used.

2) Decoder: Decoder fits the captured packets into data structures and identifies link level protocols. Then, it takes the next level, decodes IP, and then TCP or UDP in order to get useful information like ports and addresses. Snort will alert if it finds malformed headers (unusual length TCP options , etc.)

3)Preprocessors : Preprocessors can be treated as filters, which identifies things such as suspicious connection attempts to some TCP/UDP ports or too many TCP SYN packets sent in a short period of time (port scan). Preprocessors function is to take packets potentially dangerous for the detection engine to try to find known patterns. Preprocessors can alert on, classify, or drop a packet before sending it to detection engine

4) Detection Engine: Detection Engine making use of the detection plug-ins, it matches packets against rules loaded into memory during Snort initialization.

*5)*Rules Files:Rules are plain text files which contain a list of rules with a syntax. This syntax includes protocols, addresses, output plug-ins associated and some other things.

6) Detection Plug-ins: These are modules referenced from its definition in the rules files. They are used to identify patterns whenever a rule is evaluated.

7) Output Plug-ins: These are the modules which allow formatting the notifications (alerts, logs) for the user to access them in many ways (console, extern files, databases, etc). [5]

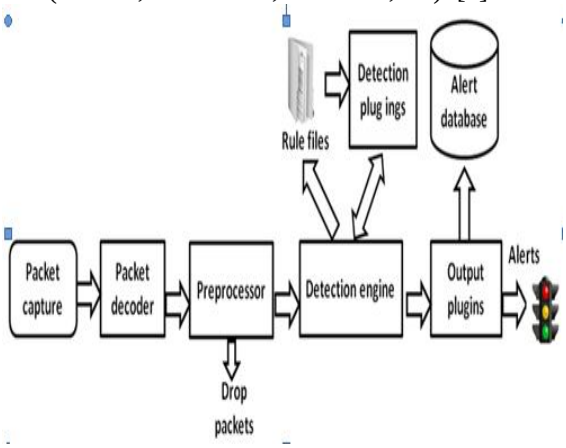

Fig. 2. Architecture of Snort

Some of the advantages of Snort are :

- Snort can easily be deployed on any node of a network, with minimal disruption to operations
- Snort provides well-documented and tested set of signatures. There are 22059 signatures available under Snort registered users on “snortrules-snapshot-2953.tar.gz “as on 26-Sep-2013.
- Portable (Linux, Windows, MacOS X, Solaris, BSD, IRIX, Tru64, HP-UX, etc.).
- Snort can act as IPS by configuring inline mode thus tends to drop packets whenever the rules trigger.

Even though snort is most popular open source IDS/IPS it has the following disadvantages :

- Information overload, rules database is very large. For example http packet in Snort having

more than 1000 signatures, thus enormous processing is required to match the packets.
- Monitoring packets in large network is an expensive task.
- Fails to detect fragmented packets at high speed networks (> 5Gbps) [3]

BRO.

Bro was originally written by Vern Paxson at Lawrence Berkeley National Lab and the International Computer Science Institute. Bro is a passive, open-source and unix based Network Intrusion Detection System (NIDS) that monitors network traffic looking for suspicious activity. Bro detects intrusions by first parsing network traffic to extract is application-level semantics and then executing event oriented analyzers that compare the activity with patterns deemed troublesome. Bro has gained its reputation due to its Stateful Protocol Analysis capabilities.[8] Bro has its own specialized policy language and if Bro detects something of interest, it can be instructed to either generate a log entry, alert the operator in real-time, execute an operating system command (e.g., to terminate a connection or block a malicious host on-the-fly). In addition, Bro's detailed log files can be particularly useful for forensics. Bro is aimed to target high-speed (Gbps), high- volume intrusion detection. Making use of packet-filtering techniques, Bro is able to achieve the necessary performance while running on commercially available PC hardware, and thus can serve as a cost-effective means of monitoring a site's Internet connection.

Bro-ids Architecture

Bro IDS architecture consists of mainly 5 modules.

1) Packet Capture: Bro captures traffic using libpcap. Packets filtered by Bro-IDS are based on ports and bits in IP or TCP headers. For examples, 13th bit of TCP header indicates whether it is set with SYN, FIN, RST or nothing. This information is important to keep the status of TCP connections states.

2) Event Engine: This layer performs several integrity checks to assure that the packet headers are well-formed. For example, it verifies the IP header checksum is correct. At this point Bro reassembles IP fragments so that network layer analyzer can accent to complete IP datagrams. It sends events to the Policy layer.

3) Signature Engine: Signature Engine inspects the packet stream, and generates an event each time a signature is matched. Those events can then be analyzed by a policy script.

4) Policy Layer: The policy script interpreter executes scripts written in a specialized Bro language. These scripts specify event handlers the happenings received for the Event

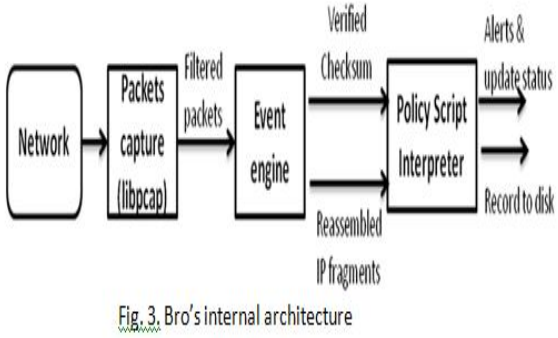

Fig. 3. Bro's internal architecture

Fig. 3. Bro's internal architecture

Advantages of Bro Network Intrusion Detection System includes following:

- Bro reassembles the packet stream prior to reaching the event engine. Reassembling at this level implies that Bro can detect, not only attacks hidden by natural TCP segmentation, but also an important type of subterfuge attacks.

- Bro-ids is capable to perform application level deep packet inspection. Bro-ids analysis file contents exchanged over application-layer protocols including MD5/SHA1 computation for fingerprinting

- Bro is capable in doing Tunnel detection and analysis (including Ayiya, Teredo, GTPv1). Bro decapsulates the tunnels and then proceeds to analyze their content as if no tunnel was in place.

- Improved forensic capabilities with the support of Time Machine, a high-performance packet bulk recorder with a Bro interface.

Some of the drawbacks of Bro-IDS are as follows :

- Bro requires a UNIX platform. Bro-ids support only Linux, FreeBSD, and Mac OS

- Bro-ids only reports information to log files and do not have a graphical user interface (GUI) supported by Bro Project. (Brownian is a web interface for viewing and interacting with Bro IDS logs provided by GitHub Enterprises).

OSSEC

OSSEC is an Open Source Host-based Intrusion Detection System that performs log analysis, file integrity checking, Windows registry monitoring,

unix-based rootkit detection, real-time alerting and active response. It runs on most operating systems, including Linux, MacOS, Solaris, OpenBSD, FreeBSD, HP-UX, AIX and Windows

The OSSEC project was founded by Daniel Cid, it was made public in 2004. In 2008, 'Third Brigade' acquired the OSSEC project, which was then acquired by 'Trend Micro' in 2009 continuing OSSEC as an open source and free [15].

The OSSEC HIDS can be installed as a stand-alone tool to monitor one host or can be deployed in a multi-host scenario, one installation being the server and the others as agents. The server and agents communicate securely using encryption. OSSEC also has intrusion prevention features, being able to react to specific events or set of events by using commands and active responses. Communication occurs on UDP port 1514 and messages are compressed using zlib and are encrypted using the symmetric key Blowfish algorithm.

OSSEC consists of a main application, a Windows agent, and a web interface. Main Application is required for distributed network or stand-alone installations. It is supported by Linux, Solaris, BSD, and Mac environments. Windows Agent is provided for Microsoft Windows environments. The main application needs to be installed and configured for server mode to support the Windows Agent. Web Interface provides a graphical user interface.

OSSEC Architecture

OSSEC is composed of multiple sections. It has a central manager for monitoring and receiving information from agents, syslog, databases and from agentless devices. It stores the file integrity checking databases, the logs, events and system auditing entries. OSSEC Agent is a small program or collection of programs installed on the systems which are need to be monitor. The agent will collect information in real time and forward it to the manager for analysis and correlation.

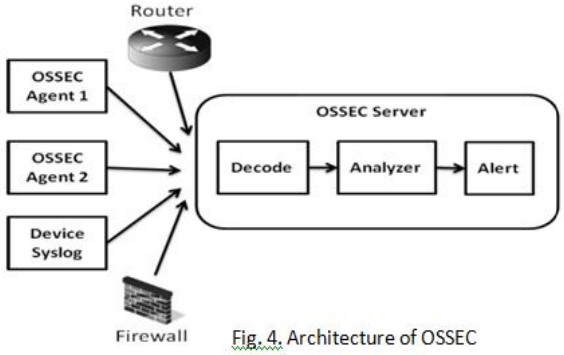

Fig. 4. Architecture of OSSEC

- Analyse logs from multiple devices and formats. The devices can be Agents, Syslog devices, Routers, Switches, Printers, etc.,
- An active response system. This means OSSEC will not only monitor, but also respond to threats (ex. black list naughty IP addresses)

Some of the disadvantages of OSSEC:

- Difficulty in upgrades between versions. OSSEC comes with default rules and they are overwritten on every upgrade.
- Coordinating pre-shared keys can be problematic. In OSSEC architecture Client and server communicate through encrypted channel using blowfish algorithm. Here pre- sharing keys before the communication establishment is a challenging issue.

TRIPWIRE.

Open Source Tripwire is a Host Based Intrusion Detection System for monitoring and alerting on specific file change(s) on a range of systems. The project is based on code originally contributed by Tripwire, Inc. in 2000. The first version of Tripwire was written by Gene Kim and Dr. Eugene Spafford at Purdue University in 1992 and released to the open source community [18] Tripwire monitors Linux system to detect and report any unauthorized changes to the files and directories. Once a baseline is created, tripwire monitors and detects, which file is added/modified, what are the changes and access/modified timestamp details. Cryptographic hashes are employed to detect changes in a file without storing the entire contents of the file in the database. While useful for detecting intrusions after the event, it can also serve many other purposes, such as integrity assurance, change management, and policy compliance.

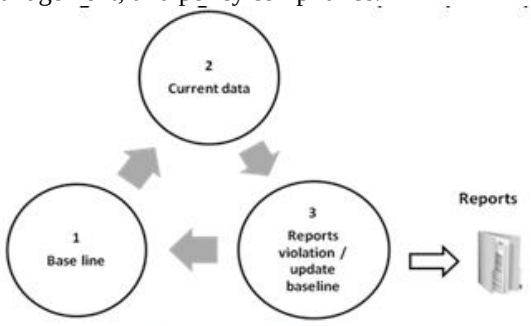

Fig. 5 Framework of Tripwire

Advantage of Tripwire:

- Advantage of tripwire is that it encrypts its database and config file.

Some of the disadvantages of Tripwire:

- Tripwire does not generate real-time alerts upon an intrusion.

- Tripwire will not detect any bugs that were already exists in the system. Tripwire should be installed right after the OS installation, and before system connected to a network for better performance.
- Open source Tripwire is suitable for monitoring a small number of Linux servers, where centralized control and reporting is not essential.

AIDE.

AIDE is a host-based IDS, which scans the file system and logs the attributes of important files, directories, and devices. Each time it runs, it compares its findings against the previous, "known good" data, and alerts you if something has changes. AIDE was originally written by Rami Lehti and Pablo Virolainen in 1999. Between 2003 and 2010 it was maintained by Richard van den Berg.. In October 2010 Hannes von Haugwitz took over the project. [15]

IDE supports multiple hash algorithms with which it can generate checksums for each file. AIDE takes a "snapshot" of the state of the system, register hashes, modification times, and other data regarding the files defined by the administrator. This "snapshot" is used to build a database that is saved and may be stored on an external device for safekeeping. AIDE is used on many Unix-like systems for file integrity checking and rootkit detection.

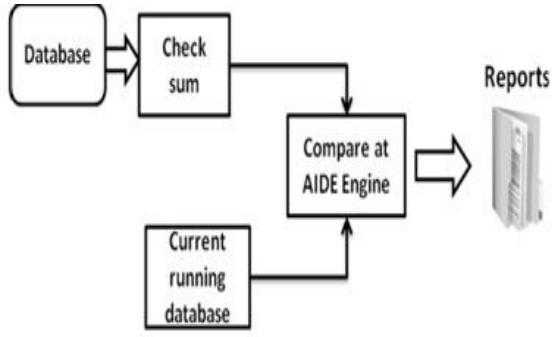

Fig. 6. Framework of AIDE

Advantages of AIDE:

- Constructs a database of directories specified in configuration file to verify the integrity of the files
- Creates a cryptographic checksum of each file and supports several message digest algorithms By default, the list includes MD5, SHA-1, SHA-256, SHA-512, WHIRLPOOL RMD-160, Tiger, HAVAL, and CRC-32
- Automatically generate a daily report. The report is mailed to root and is in /var/mail/root

Disadvantages of AIDE:

- AIDE report changes after the incident. It does not prevent file from being altered.
- AIDE does not encrypt and sign the baseline database by default. Hence one has to sign the database for greater security or the newly created database should be moved to a secure location such as read-only media, otherwise an attacker could read and modify the configuration file.

SAMHAIN.

The Samhain host-based intrusion detection system (HIDS) provides file integrity checking and log file monitoring/ analysis, rootkit detection, port monitoring, detection of rogue SUID executables, and hidden processes. Samhain been designed to monitor multiple hosts with potentially different operating systems, providing centralized logging and maintenance, although it can also be used as standalone application on a single host. Samhain can run on platforms like Unix, Linux, Cygwin/Windows (crywin/ windows supports Samhain monitoring agent only as on Oct 2013) [19]. Samhain uses cryptographic checksums of files to detect modifications. Samhain can run continuously as a daemon (background process), and any stop/restart process will leave a recognizable mark. Thus it is capable to find rogue SUID executables anywhere on disk as long as the daemon is running. Samhain can also monitor which ports are open on the local host, and compare against a list of allowed or required port/services. Samhain equipped with a central log server. Messages are sent via encrypted TCP connections. Clients need to authenticate to the server. Database and configuration files can be signed, log file entries and e-mail reports are signed and support for stealth operation.

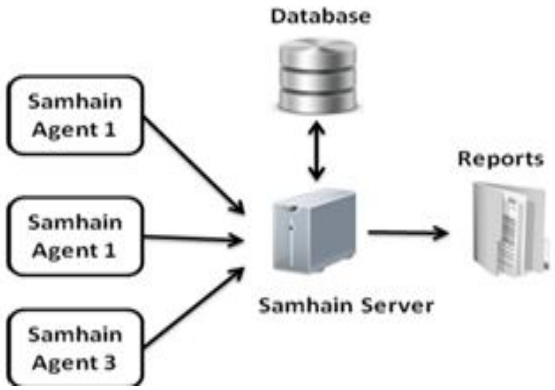

Fig 7. Samhain architecture

Advantages of Samhain:

- Samhain support for logging to a central server via encrypted and authenticated connections,

signing of database and configuration files is an added advantage.

- Samhain can perform incremental checks on growing logfiles i.e. verify at each check that the data present at the preceding check have not been modified, which is one of the main requirement by Sect. 10.5.5 of the PCI DSS.(Payment Card Industry (PCI) Data Security Standard (DSS).

IV. COMPARISON OF OPEN SOURCE INTRUSION DETECTION TOOLS

This paper discussed about Network Intrusion Detection Systeom tools Snort and Bro & Host Intrusion Detection System tools OSSEC,Tripwire,AIDE and ,Samhain. While choosing an Network Intrusion Detection System, Snort is one of the best lightweight IDS/IPS which can run on many operating systems. Snort can easily be deployed on any node of a network, with minimal disruption to operations. It has

TABLE I
COMPARISON OF OPEN SOURCE NIDS TOOLS

Tools / Features	SNORT	BRO
Supported Platforms	Unix like systems , Windows, MacOS X,etc	Unix like systems, and Mac OS
License	GNU GPL v.2	BSD license
PGP Signed	●	●
IPS feature	●	x
Support to High speed Network	Medium	High
Probe attacks	●	●
buffer overflows	●	●
SQL injection	●	●
Web application attacks	●	●
DOS attacks	●	●

TABLE II
COMPARISON OF OPEN SOURCE HIDS TOOLS

Tools / Features	OSSEC	Tripwire	AIDE	Samhain
Supported Platforms	Unix like systems and Windows	Linux, all POSIX/UNIX Sys.	Unix like systems	Unix like systems and Windows
License	GNU GPL v.2	GNU GPL	GNU GPL	GNU GPL
PGP Signed	●	x	●	●
IPS feature	●	x	x	x
File integrity checking	●	●	●	●
Windows registry monitoring	●	x	x	●
Rootkit detection	●	●	●	●

very high speed networks. On the other side Bro is suitable for those who are working with high speed network. Bro is flexible and highly customizable, but Bro-ids is not suitable for those who are working with windows environment. With the help of the script, snort2bro, Snort signatures can be converted automatically into Bro's signature syntax. However, one can't benefit from the additional capabilities that Bro provides as the approaches of the two systems are just too different. Bro organisation is now stopped maintaining the snort2bro script, and there are now many newer Snort options which it doesn't support and now the snort2bro script is now no longer part of the Bro distribution. The comparison of these two Network Intrusion Detection tools is shown at Table I. Open source HIDS tool Samhain can perform incremental checks on growing logfiles, this feature is not available on OSSEC, AIDE and Tripwire, and moreover Samhain can also monitor which ports are open on a particular localhost. Tripwire and Samhain are able to encrypt and sign the database whereas AIDE cannot. OSSEC performs analysis on the server side, which means that the server can become a performance bottleneck. Hence OSSEC might show degrade performance when numbers of agents increases. Samhain does this analysis on the client side, and agents forward reports based on policy violations to the server. This minimizes both the network load and the computational load on the server. Table II gives the brief comparisons of Host based Intrusion Detection tools.

CONCLUSION

Network security is primary concern of any organisation. By using Intrusion detection tools one can protect their home or organisation from several types of attacks. Open Source Intrusion Detection tools allows the users customise installation as per their security requirement. Each Intrusion Detection System Tools have their own advantages and disadvantages, choosing the best one depend on organisational requirements. By combining NIDS and HIDS we are able to find out any attacks that bypass NIDS and to find out whether a network intruder has been successful or not at the targeted host.

REFERENCES

[1] Michael E. Whitman. "Principles of Information Security", 2012
[2] Behrouz A. Forouzan, " Cryptography and Network Securiy" 2nd edition 2012.
[3] Tian Fu, " An Analysis of Packet Fragmentation Attacks vs. Snort Intrusion Detection System", International Journal of Computer Engineering Science (IJCES), May 2012.
[4] Rainer Wichmann," The Samhain HIDS Overview of available features", November 1, 2011
[5] Miguel A. Calvo Moya," Analysis and Evaluation of the snort and bro network intrusion detection Systems" 2008
[6] Sourcefire website http://www.sourcefire.com, 30 Oct 2013
[7] http://www.cisco.com/web/about/ac49/ac0/ac1/ac259/sourcefire.html, Oct 2013

[8] Bro website ,http://www.bro.org/, 30 Oct 2013
[9] Brian L. Tierney, "An Overview of the Bro Intrusion Detection System" 2004
[10] Herve Debar "Evaluation of the Diagnostic Capabilities of Commercial Intrusion Detection Systems" 2004
[11] Martin Roesch, "SNORT – Light weight Intrustion detection for networks",Proceedings of LISA '99: 13th Systems Administration Conference Seattle, Washington, USA, November 7–12, 1999
[12] Kathleen A. Jackson, "intrusion detection system (ids) product survey", 1999
[13] http://en.wikipedia.org/wiki/OSSEC, 30 Oct 2013
[14] SNORT R Users Manual 2.9.5, The Snort Project, released on May 29, 2013
[15] OSSEC website, http://www.ossec.net/, 30 Oct 2013
[16] SNORT website , http://www.snort.org , 30 Oct 2013
[17] http://aide.sourceforge.net/, 30 Oct 2013
[18] Tripwire website http://www.tripwire.com,30 Oct 2013
[19] Samhain website, http://www.la-samhna.de/samhain/, 30 Oct 2013.

★★★