DOI: 10.21276/sjet.2017.5.3.8

Scholars Journal of Engineering and Technology (SJET) **ISSN 2321-435X (Online)**
Sch. J. Eng. Tech., 2017; 5(3):122-130 **ISSN 2347-9523 (Print)**

(An International Publisher for Academic and Scientific Resources)
www.saspublisher.com

Review Article

Intrusion Detection System Techniques and Tools: A Survey

Resmi AM[1], Dr. R Manicka chezian[2]
[1]Ph D. Research Scholar, Dept of Computer Science, NGM college, (Autonomous), Pollachi-642001, India
[2]Associate Professor, Dept. of Computer Science, NGM College (Autonomous), Pollachi-642001, India

***Corresponding author**
Resmi AM
Email: resmiswararag@gmail.com

Abstract: An Intrusion Detection System (IDS) is a system that tries to perform intrusion detection by comparing observable behaviour against suspicious patterns. The objective of intrusion detection is to monitor network resources and to detect abnormal and irregular behaviours and abuses. This concept has been around for the past several years but only recently it has seen a dramatic rise in interest of researchers and system developers for incorporation into the overall information security infrastructure. This survey gives the overall study about the IDS, its nature and techniques, tools used in the area of intrusion detection. Finally the survey gives the real-time working performance of top selected Intrusion Detection and Intrusion Prevention tools. This paper helps in analysing and evaluating of various IDS tools used in high-speed networks.
Keywords: Intrusion Detection System, Anomaly Detection, SNORT, SURICATA, Bro IDS.

INTRODUCTION

An intrusion detection system (IDS) examines network traffic for any suspicious and irregular activity and alerts the system or network administrator [1]. In some cases the IDS it may also counter to anomalous or malicious traffic by taking action such as blocking or isolating the user or source IP address from accessing the network. The goal of Intrusion detection systems is to identify attacks with a high detection rate and a low false alarm rate [2].

Fig-1: IDS Process

The fig 1 shows the basic process of IDS, which performs monitoring, analysis, alert and response to the detected defect. The IDS are created with the software which assesses the network security by monitoring the network activities. The software's are allows the network controller to inspect the network for vulnerabilities and thus securing potential loopholes before attackers take advantage of them. The IDS are in different types such as Network based IDS (NIDS), Host based IDS (HIDS), anomaly based IDS (AIDS) and Network-node Intrusion detections systems (NNIDS). Based on the type, different types of tools are developed. In this survey, we provided the types of intrusion detection techniques and tools.

Classification of Intrusion Detection Systems

Intrusion detection systems can be classified as six types, which are listed below fig 2:

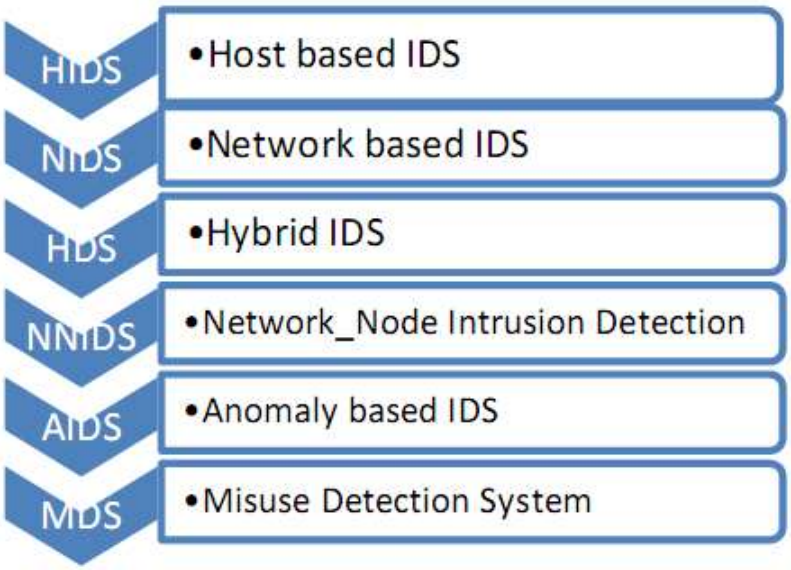

Fig-2: IDS Types

Host-Based Intrusion Detection System

Host-based intrusion detection system are designed to monitor, detect, and respond to user system activity and attacks on a given host [3]. Some robust tools offer centralized audit policy management, supply data forensics, statistical analysis and evidentiary support, as well as provide some measure of access control. Host-based intrusion detection is best suited to combat internal threats and abnormal behaviors in the local networks, because of its ability to monitor and respond to specific user actions and file accesses on the host. The greater part of computer threats origin within concerns. Host based IDS relies on the single system and the audit log details are stored in every machine. If attacker takes over a system, then the attacker can tamper with IDS binaries and modify audit logs.

Network Intrusion Detection

Network intrusion detection deals with information passing on the wire between hosts, which typically referred to as "packet sniffers". The network IDS devices intercept packets traveling along various communication mediums and protocols [4]. The TCP/IP protocol is usually used. This captures the packets and analysed in a number of approaches. Several Network based Intrusion Detection devices simply compare the packet to a signature database. This verifies whether it contains any known attacks and malicious packet or not. It also verifies the packet and its activity, because that might indicate malicious behaviour of a specified transaction. In either case, NID should be regarded primarily as a boundary resistance. NID has historically been incapable of operating in the following environments:

1. Switched Networks
2. Encrypted Networks
3. High-Speed Networks (Anything Over 100 Mbps)

The difference between host-based and network-based intrusion detection is that Network Intrusion Detection (NID) deals with data transmitted from host to host but Host based ID is concerned with what happens on the hosts themselves.

Hybrid Intrusion Detection System:

Hybrid intrusion detection systems facilitate management and alert notification from both network and host-based intrusion detection devices. Hybrid solutions provide logical complement to NID and HID - central intrusion detection management. Recently, Cisco released a module for their Catalyst 6000 switch that incorporates network intrusion detection directly in the switch, overcoming the first of these flaws. Additionally, ISS (Internet Security System) Network indicated that they are now capable of "packet-sniffing" at gigabit speeds [5].

Network-Node Intrusion Detection (NNID):

Network-node intrusion detection (NNID) was developed to work around the intrinsic defects in traditional Network IDs. Network-node pulls the packet intercepting technology off of the wire and puts it on the host. With NNID, the "packet-sniffer" is positioned in such a way that it captures packets after they reach their final target or destination system. The received packet at the destination is then analysed just as if it were traveling along the network through a conventional "packet-sniffer". This scheme came from a HID-centric assumption that each critical host would already be taking advantage of host based technology. In this scheme a network-node (NN) is simply another component that can connect to the HID agent. A major disadvantage is that it only evaluates packets addressed to the host on it exists. Traditional network intrusion

detection on the other hand monitors packets on the entire subnet. In this case, "Packet-sniffers" are incapable of viewing a complete subnet when the network uses high speed communications, switches or encryption. The advantage of NNID is its ability to defend specific hosts against packet based security issues in these complex environments. This will be very effective where conventional NID is ineffective.

Anomaly based IDS:

Anomaly based detection systems observes activities that deviate significantly from the established normal usage profiles as possible intrusions. For example, the normal profile of a user may contain the averaged frequencies of some system commands used in their login sessions [6]. This will raise an alarm when the frequencies are differs. So this have follows a continuous monitoring process. The key advantage of anomaly detection is that it does not necessitate preceding information's or data of intrusion, so it can thus detect new intrusions.

Misuse Detection Systems:

Misuse detection systems use patterns of well-known attacks or feeble spots to find the intrusions. This system matches and identifies known intrusion using the set of patterns. For example, if a user failed to login more than four attempts within a specific time, then it will declare as password guessing attacks. This can be detected using a signature "if". The main disadvantage is that it lacks the ability to detect the unknown attacks. The concept behind misuse detection schemes is that there are ways to represent attacks in the form of a pattern or a signature so that even variations of the same attack can be detected [7]. This means that these systems are similar to virus detection systems. They can detect many or all known attack patterns, but are of little use for as yet unknown attacks. An interesting point to note is that whereas anomaly detection systems try to detect the complement of bad behaviour, misuse detection systems try to recognize known bad behaviour using the given patterns. The major dilemma in misuse detection systems are how to write a signature that encompasses all possible variations of the pertinent attack, and how to write signatures that do not match non-intrusive activity.

POPULAR IDS TOOLS

There are several IDS tools are available at free of cost. The followings are the list of IDS tools with its descriptions, advantages and disadvantages.

Snort

Snort is open source software and light weight software developed by Martin Roesch in 1998, and is an open source network intrusion detection and prevention system [8]. Snort software act as a packet sniffer, packet logger or as a network intrusion detection and prevention system [NIDS, NIPS]. Snort will read network packets and display them on the console if it is in a sniffer mode, it will log packets to disk at the time of packet logger selection. It will monitor the network traffic and analyse the traffic against a rule set defined by the user at the time of network intrusion detection and prevention system. This comes under network IDS. This has been developed for Linux and Windows to detect emerging threats. SNORT has the ability to make concurrent traffic analysis and packet logging on Internet Protocol (IP) networks. This also can perform protocol analysis, content searching and matching. Snort uses both signature-based intrusion detection as well as anomaly-based methods, and can rely on customized user rules or signatures sourced from databases like Emerging Threats.

Rule in Snort

The header is made up with the following along with the instructions: log or alert; type of network packet: tcp/udp/icmp/etc., source and destination IP addresses and ports; an alert message and possibly some qualifiers, signatures and classification type [9]. When Snort discovers a packet that matches the wrong signature value, it collects the rules from the user and follows the action listed in the user customized rule list. Totally, there are two possible and common actions such as alert and pass can be done.

Table 1: A Sample Snort Rule and Its Various Parts

Action	**Alert**
Protocol	Tcp
Source And Port	Corvitz1 Any
Destination And Port	$Home_Net Any
Alert Message	"Msg Info: Psybnc Anomaly Access"
Qualifier	Flow:From_Server, Established
Signature	Content:"Welcome!Psybnc! 1am3rz.De"
Classification Type	Bad: Unknown

Table 1.0 shows the rule sample of SNORT. The "**alert"** action produces an action. The action can store the log or it can be, emailed or it can create an alarm to a windows machine. The "**pass"** action just

stops the processing of packets if the signature matches in the packet.

Table 2: Memory and Processor Time Used by Snort

Memory Consumption	Processor Consumption
25.8	43.3
30.54	46.4
35	55.3
29.1	42.14
35	52
30.3	42.7

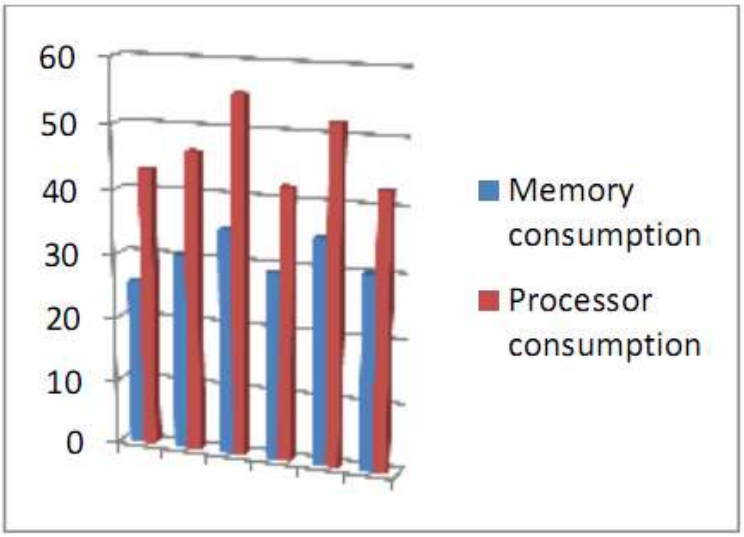

Fig-3: Memory and Processor Utilization

This graph shows the performance analysis of memory and processor time used by Snort. The snort tools are very easy to install and run. And the user can customize the rules. It produces pop-up windows and alerts. These pop-up windows are controlled by Windows Messenger Service. Snort has several above advantages. However, it has less GUI and packet capturing process is very slow and its supports only medium level of high speed network.

From the above points, it shows the installation and management of Snort is easy and moderate for all type of applications.

Suricata

Suricata is a fast and robust network intrusion detection engine. It is capable of real time intrusion detection (IDS), as like Snort Suricate also based on a signature-based methodology, rule/policy driven security, and anomaly-based approach for detecting intrusions [10]. Suricata inspects the network traffic using a powerful and extensive rules and signature language, and has powerful Lua scripting support for detection of complex threats. Suricata performs per rule alert filtering and thresholding, global alert filtering and thresholding and per host/subnet thresholding and rate limiting settings.

Rule In Suricata

It has the following details in the rule creation process. This includes Action, Header and rule options.

Action

The action part contains the pass, drop, reject, and alert. If a signature matches and contains "**Pass"**, Suricata stops scanning the packet and skips to the end of all rules. This is performed only for the current packet. If the action is specified as Drop, then this only concerns the IPS/inline mode.

If the program finds a signature that matches, containing "**Drop"**, it stops immediately. The packet will not be sent any further.

If that contains a keyword like Drawback, then the receiver does not receive a message of what is going on, resulting in a time-out (certainly with TCP). This tool generates an alert for this packet. If the "**Reject"** keyword is selected, it rejects the packet. Both receiver and sender receive a reject packet. There are two types of reject packets that will be automatically selected in this case. If the offending packet involved with TCP protocol, it will be a Reset packet. For all other protocols it will be an ICMP error packet. This also

generates an alert, When in Inline/IPS mode enabled; the wrong packet will also be dropped like with the 'drop' action in this process. If a signature matches and contains "**Alert"**, the packet will be treated like any other non-threatening packet, except for this one an alert will be generated by Suricata. Only the system administrator can notice this alert.

Direction:-another feature in the SURICATA rule is direction. This tells the signature matching direction. Nearly every signature has an arrow to the right. This means that only packets with the same direction can match.

Alert tcp 1.2.3.4 1024 - > 5.6.7.8 80

Rule Options: the final portion of SURICATA is rule options, which specifies settings format in the rule.

Table 3: A Sample Suricata Rule and Its Various Parameters

Rule Parameters	**Example**
Action	Pass, Drop, alert and Reject
Direction	tcp $Home_PC any → $external_PC any
Rule options	Meta-information, headers, payloads and flows

```
drop tcp $HOME_NET any -> $EXTERNAL_NET any (msg:"ET
TROJAN Likely Bot
Nick in IRC (USA +..)"; flow:established,to_server;
flowbits:isset,is_proto_irc; content:"NICK "; pcre:"/NICK
.*USA.*[0-9]{3,}/i"; classtype:trojan-activity;
reference:url,doc.emergingthreats.net/2008124;
reference:url,www.emergingthreats.net/cgi-
bin/cvsweb.cgi/sigs/VIRUS/TROJAN_IRC_Bots;
sid:2008124; rev:2;)
```

Table 3.0 shows the Suricata rule example with various parameters. This includes the above specified three parameters, action, direction and rule options. As like snort the Suricata rules are specified with four different actions.

The main advantage of Suricata is user can customize and use the same Snort's rule sets. It has advanced features such as multi-threading capabilities and GPU acceleration. But it creates more false alarms at the time of detection. The system and network resource usage is exhaustive in the Suricata.

As per the above analysis Suricata is also good like Snort tool, and it have rule customization and rule reusability, but it creates many false alarms in real-time usage.

Bro IDS

Bro IDS is anomaly-based intrusion detection, and is usually employed in combination with Snort, as the two complement each other quite nicely. Interestingly, Bro is actually a domain-specific language for networking applications in which Bro IDS is written. The technology is especially effective at traffic analysis, and is often used in forensics and related use cases. This policy engine has its own language. Bro IDS consists of the following major components such as libpcap, Event Engine and Policy Script Interpreter [11]. Bro capture packets from the network interfaces using libpcap library libpcap takes care of all the traffic that comes from the network layer and filters out the non-important elements. The filtered packet stream is forwarded to the Event engine. The received packets are combined together for taking necessary actions by the event engine. The policy script interpreter matches the packets with the rules. This finds the suspicious and dangerous actions and discards the unmatched packets.

Bro has a very different approach compared to the other tools. The rules in Bro work with scripts. It's a script driven IDS. This can support high throughput environments. The processing time is less when comparing with the existing SNORT and SURICATA. The Bro IDS distributes the load to the multiple servers.

This platform can be customized for a variety of network security in addition to NIDS. It can do some very powerful and versatile tasks. And this tool can

detect patterns of activity other IDS systems cannot. But this is difficult to interpret and configure. User need more skill on programming to execute the rules.

The Bro software is more adaptive in nature; it provides domain-specific scripting language enables site specific monitoring policies. And it provides high performance and flexible in nature. While it supports such standard functionality as well, Bro's scripting language indeed facilitates a much broader spectrum of very different approaches to finding malicious activity, including semantic misuse detection, anomaly detection, and behavioural analysis.

Open WIPS-ng

OpenWIPS-ng is an open source and modular Wireless IPS (Intrusion Prevention System). It captures wireless traffic and detects and identifies standard and hidden networks in order to attempt to detect intrusions [12]. Pattern based IDS have also been designed in recent studies that concentrate on the configuring of an IDS solution that will allow the method of detection to be based on an essential part of the network such as protecting specific protocols as a basis for the method of detection. It is composed of three parts:

Sensor(s): a "Dumb" device that capture wireless traffic and sends it to the server for analysis. Also responds to attacks.

Server: Aggregates the data from all sensors, analyses it and responds to attacks. It also logs and alerts in case of an attack.

Interface: GUI manages the server and displays information about the threats on your wireless network(s). This is signature based intrusion detection. This can run in the commodity hardware. This performs scanning, detection and intrusion prevention process.

The open wireless intrusion prevention technique is modular and plug-in based. It needs several hardware's and components. The main advantage of using this tool is it can prevent the users from connecting to other networks. It has multiple sensor support so the detection accuracy is higher. Even though it has more advantages, there are several negative aspects in this tool, which are this is only supports to the wireless security solution. If the legitimate users are attacked, it will ban both from the network until the issue is solved or some time. The traffic between sensor and server is not encrypted

This tool allows an administrator to download plug-ins for additional features, and it relies on wireless networks. Most of its detection functions are based on the plug-ins, Shared libraries. Some basic checks need to be always run and run before plugins. This need fast reaction, because it runs the sensors. If the attacker found, it de-authenticate and alert the admin. This uses Script Wrapper to a scripting language.

OSSEC

OSSEC is a Host based IDS. It is scalable, multi-platform, open source Host-based Intrusion Detection System (HIDS). It only stores alerts, not every single log [13]. So storage overhead is reduced. It has a powerful correlation and analysis engine, integrating log analysis facility; file integrity checking; Windows registry monitoring process; centralized policy enforcement; root-kit detection; real-time alerting and active response. This is capable of detection DOS attack.

The advantages of OSSEC are it is easy to install and customize. And it can supports multi-platform. This performs File Integrity checking for UNIX and Windows platforms. And the main advantage of this tool it can performs Registry Integrity checking for Windows. In OSSEC, Transitioning to newer versions of the platform can be difficult and the upgrade process overwrites existing rules with out-of-the-box rules. So the user rules can be deleted. This tool utilizes the pre-sharing mechanisms. But the Pre-sharing keys can be problematic when Windows in server-agent mode.

OSSEC is composed of multiple portions. It has an ability to monitor distributed servers. It receives information from agents and from agent less device. Here the distributed server stores the access logs and system auditing logs. And the agents are the small program unit, which forward the details to the server for analysis and correlation. The Agentless scheme allows the user to perform file integrity monitoring on them without the server or agent. This scheme used to monitor firewalls, routers and even Unix like systems.

Fragroute

Fragrouter is a network intrusion detection (NIDS) evasion toolkit. It implements most of the attacks described in the Secure Networks such as Insertion, Evasion, and Denial of Service (DOS) etc., It evades Network Intrusion [14]. Fragroute can exploit TCP/IP protocols and it is a one way fragmenting router IP packets get sent from the attacker to the Fragrouter, which transforms them into a segmented data-stream to dispatch to the victim. Fragrouter assista an attacker launch IP-based attacks while avoiding detection. It features a simple rule-set language to holdup, replicate, drop, break, overlap, print, reorder, segment or source-route with all outbound packets destined for a destination system, with minimal support for randomized or probabilistic behavior. The following is sample rule format used in the fragroute

fragroute –f <lconfigfile> dst<destination>

The main advantage of fragroute is it doesn't require additional libraries [15].

Security Onion

It is actually an Ubuntu-based Linux distribution for IDS and network security monitoring (NSM), and consists of several of the above open source technologies working in concert with each other [16]. The security onion platform gives comprehensive intrusion detection, network security monitoring and log management by combining the best of Snort, Suricata, Bro and as well as other tools such as Sguil, Squert, Snorby, ELSA, Xplico [18], among others. For those desiring the best of the aforementioned tools in one single package, Security Onion is worth considering among all. Security Onion contains three major functionalities, such as full packet capturing process, network-based (NIDS) and host-based intrusion detection intrusion detection systems (HIDS) and powerful analysis tools, and provides log and alert data for detected events and activities. Security Onion provides multiple IDS options.

Sguil: Sguil is a graphical interface providing real-time access to events, session data and packet data captured by the Snort or Suricata IDS systems. Sguil facilitates the practice of Network Security Monitoring and event driven analysis.

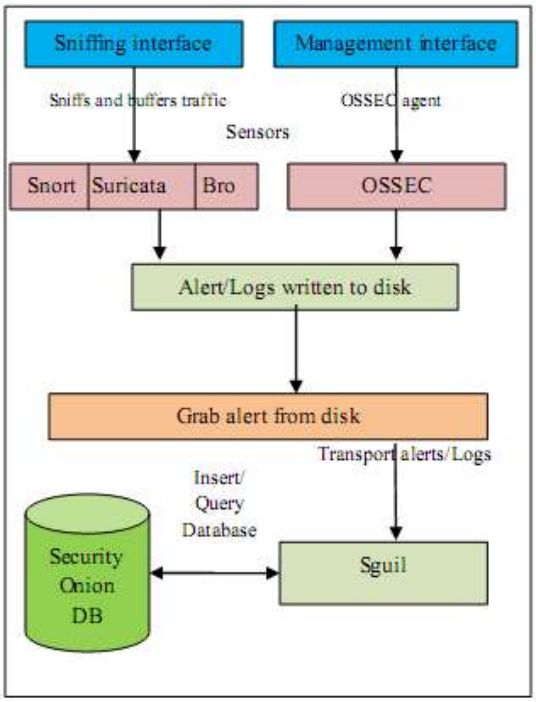

Fig-4: Security Onion Architecture

The fig 4.0 shows the overall architecture of the security onion, which contains two types of interfaces. One is sniffing interface and other one is management interface. This sniffing interface collects traffic information's and different sensory events, this will be applied into the ID tools such as Snort, Suricata and Bro ids. The management interface is configured using static IP. The collected alert and logs are stored into the disk, the alerts will be transmitted to the Squil ie the security onion framework for further decisions.

Table 4: Comparisons Of Different Intrusion Detection And Prevention Tools.

Tool name	Provider	type	Description	Platform
SNORT	Cisco system	NIDS, NIPS	Can Detect Dos, CGI, Intrusion, Port Scans, SMB And Layer Attacks. SNORT Has The Ability To Make Concurrent Traffic Analysis And Packet Logging On Internet Protocol (IP) Networks	(Cross Platform) Linux, windows
SURICATA	Open information security foundation	NIDS, NIPS	Automatic Protocol Detection, File Matching Process And Compatible With SNORT	Linux, unix ,MAC,windows etc.,
Bro IDS	Vern Paxson	NIDS, AIDS	Employed In Combination With Snort	Linux, MAC OS X, FreeBSD
OpenWIPS-ng	Aircrack-NG	NIPS	Openwips-Ng Is An Open Source And Modular Wireless IPS (Intrusion Prevention System).	Linux
Security Onion	-	NIDS	It Contains Snort, Suricata, Sguil, Squert, Snorby, Bro, Networkminer, Xplico, And Many Other Security Tools	Linux
OSSEC	Daniel B. Cid	HIDS	It Has A Powerful Correlation And Analysis Engine, Integrating Log Analysis	Cross-platform
FRAGROUTE	Dug Song	NIDS	This Tool Was Written In Good Faith To Aid In The Testing Of Network Intrusion Detection Systems, Firewalls, And Basic TCP/IP Stack Behaviour.	Linux

Table 5: Features Based Comparison Between Different Intrusion Detection Tools

Parameter	Snort	SURICATA	**Bro IDS**	**Openwips-Ng**	**OSSEC**	**FRAGROUTE**
Contextual Signatures	No	Yes	Yes	No	Yes	No
IPS Feature	Yes	Yes	No	Yes	No	No
Dos Attack	Yes	Yes	Yes	Yes	Yes	Yes

The table 3.0 shows the comparison between the different tools. This shows the basic comparison with the platform support and description basis [17]. The following chapter gives the comparison based on the performance of each tool in terms of its different features. Every ID has its own feature and advantage. As per different parameter, the comparisons are made. When comparing the IDS, both network and host based IDS are compared with the common feature sets.

CONCLUSION

In this survey we reviewed and studied about various intrusion detection system and tools. The survey gives the overview and its merits and demerits of the tools that are used to detect and prevent the intrusions. In this paper we analyzed six types of IDS and seven intrusion system tools, the IDS types are network based, host based intrusion detection system, hybrid IDS, net-host based, anomaly based IDS and misuse intrusion detection systems. From the comparison made between these tools and techniques, we summarize some points. Several tools support only little type of security threats and issues. Defining rules properly will lead the highest detection rate, so the rules should be configured properly. Several tools are still having trouble in detection of accurate intruders with minimum hardware and sensor supports. So there is a need to provide a comprehensive analysis to make a new and effective tool with high accuracy in detection and less in computational cost.

REFERENCES

1. Maqbool BB, Bashir U, Chahcoo M. "Intrusion Detection and Prevention System: Issues and Challenges. International Journal of Computer Applications. 2013; 76.17.
2. Mukherjee, Biswanath L, Heberlein T, Levitt KN. Network intrusion detection. IEEE network. 1994;8(3): 26-41.
3. Sandip K. "Host based intrusion detection system. International Conference on Mechanical Engineering and Technology (ICMET-London 2011). ASME Press, 2011.
4. Vigna, Giovanni, Kemmerer RA. NetSTAT: A network-based intrusion detection system." Journal of computer security. 1999; 7(1): 37-71.
5. Ali AM, Zaim AH, Ceylan KG. A hybrid intrusion detection system design for computer network security. Computers & Electrical Engineering. 2009;35(3): 517-526.

6. Garcia-Teodoro, Pedro. Anomaly-based network intrusion detection: Techniques, systems and challenges. computers & security. 2009; 28(1) : 18-28.
7. Depren, Ozgur. An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks. Expert systems with Applications. 2005; 29(4): 713-722.
8. Fu T. An Analysis of Packet Fragmentation Attacks vs. Snort Intrusion Detection System. International Journal of Computer Engineering Science (IJCES), May 2012.
9. Roesch, Martin. Snort: Lightweight Intrusion Detection for Networks. LISA. 1999; 99(1).
10. Day, David, Burns B. A performance analysis of snort and suricata network intrusion detection and prevention engines. Fifth International Conference on Digital Society, Gosier, Guadeloupe. 2011.
11. Mehra, Pritika. A brief study and comparison of snort and bro open source network intrusion detection systems. International Journal of Advanced Research in Computer and Communication Engineering. 2012;1(6): 383-386.
12. Open WIPS-ng. Available: http://www.openwips-ng.org/. Last accessed 10th Sep 2015.
13. OSSEC website, http://www.ossec.net/, 30 Oct 2013
14. Holestein, Michael. How does fragroute evade nids detection. Intrusion detection FAQ, 2002.
15. https://www.monkey.org/~dugsong/fragroute/
16. Burks, Doug. "Security Onion. nd.[Online]. Available: http://blog. securityonion. net/p/securityonion. html.[Accessed 11 May 2014] (2012).
17. Alnabulsi, Hussein, Islam MR, Mamun Q. Detecting SQL injection attacks using SNORT IDS. Computer Science and Engineering (APWC on CSE), 2014 Asia-Pacific World Congress on. IEEE. 2014.
18. Bejtlich, Richard. The practice of network security monitoring: understanding incident detection and response. No Starch Press, 2013.