Intrusion Detection Techniques and Open Source Intrusion Detection (IDS) Tools

Rana M Pir
Lecturer
Leading university, sylhet Bangladesh

Abstract **- Network based technology and Cloud Computing is becoming popular day by day as many enterprise applications and data are moving into cloud or Network based platforms. Because of the distributed and easy accessible nature, these services are provided over the Internet using known networking protocols, Protocol standards and Protocol formats under the supervision of different management's tools and programming language. Existing bugs and vulnerabilities in underlying technologies and legacy protocols tend to open doors for intrusion so many Attacks like Denial of Service (DDOS), Buffer overflows, Sniffer attacks and Application-Layer attacks have become a common issue today. Recent security incidents and analysis Have manual response to such attacks and resolve that attacks are no longer feasible. In Internet and Network system application or platform facing various types of attacks in every day. Firewalls security and spam filters are in place but they have simple rules such as to allow or deny protocols, ports or IP addresses. Some DoS and other attacks are too complex for today's firewalls, so firewalls cannot prevent that all attacks. In this paper we define and discuss various types and techniques of Intrusion Detection, Intrusion Prevention and the IDS tools that are employed to detect these attacks and discuss some open source tools to prevent and detection of intrusion and how can we use Open Source tools in our system.**

Index Terms **- Intrusion detection system, Intrusion prevention system, HIDS, NIDS, DoS, DDoS, DIDS, Intrusions, cloud security, vulnerabilities, anomaly detection, IDS, Network Attacks**

I. INTRODUCTION

Intrusion detection is the process of monitoring the attacks and events occurring in a computer or network system and analyzing them for signs of possible incidents of attacks, which are violations or imminent threats of violation of computer security policies, acceptable use policies, or standard security practices. Incidents have many causes, such as malware (e.g., worms, spyware, Denial of Service (DDOS), Buffer overflows, Sniffer attacks and Application-Layer attacks), attackers gaining unauthorized access to systems from the Internet, and authorized users of systems and misuse their privileges or attempt to gain additional privileges for which they are not authorized. As network attacks have increased in number and severity over the past few years, intrusion detection systems have become a necessary addition to the security infrastructure of most organizations.

This Paper is intended as a primer in intrusion detection, developed for those who need to understand what security goals intrusion detection mechanisms serve, how to select and configure intrusion detection systems for their specific system and network environments, how to manage the output of intrusion detection systems, and how to integrate intrusion detection functions with the rest of the organizational security infrastructure.

II. TYPE TYPES OF ATTACKS

- **Denial-of-Service (DOS) attacks,** It is an attempt to forbid the authorized users from utilizing the requested service/ resource. A more advanced Distributed Denial of Service occurs when in a distributed environment the attacker sends or rather floods the server or a target system with numerous connection requests knocking the target system to the knees, leaving them no other option to restart their system. Some well known DOS attacks are:
  - SYN Attack where the attacker exploits the inability of the server to handle unfinished connection requests. Server is flooded with connection requests. The server crashes waiting for the acknowledgments of the requests.
  - Ping of Death where the attacker sends a ping request which is larger than 65,536 bytes which is the maximum allowed size for the IP, causing the system to crash or restart
- **Logon Abuse attacks**, a successful logon abuse attack would bypass the authentication and access control mechanisms and grant a user with more privileges that authorized.
- **Application-Level Attacks,** The attacker exploits the weakness in the application layer – for example, security weakness in the web server, or in faulty controls in the filtering of an input on the server side. Examples include malicious software attack (viruses, Trojans, etc), web server attacks, and SQL injection.
- **Spoofing attack,** the attacker impersonates an legitimate user. IP spoofing is a common example where the system is convinced that it is communicating with a trusted user and provides access to the attacker. The attacker sends a packet with an IP address of a known host by alerting the packet at the transport layer.
- **Sniffer Attack**, A sniffer is an application that can capture network packets. Sniffers are also known as network protocol analyzers. While protocol analyzers are really network troubleshooting tools, they are also used by hackers for hacking network. If the network packets are not encrypted, the data within the network packet can be read using a sniffer. Sniffing

refers to the process used by attackers to capture network traffic using a sniffer. Once the packet is captured using a sniffer, the contents of packets can be analyzed. Sniffers are used by hackers to capture sensitive network information, such as passwords, account information etc.

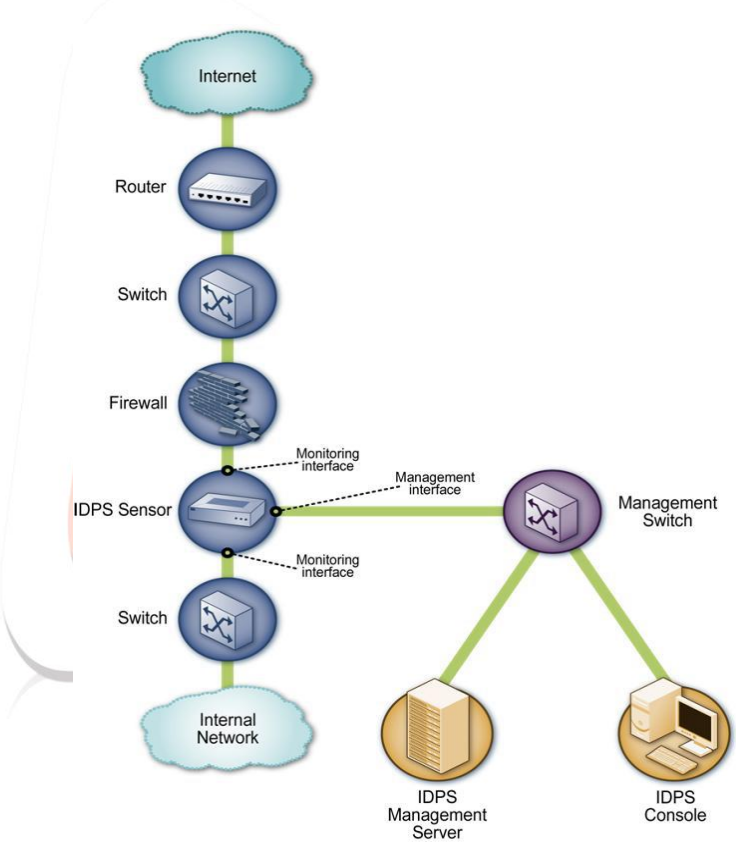
Sniffer tool usage	
Ethical usage	**Unethical usage**
Packet capturing	User identity and password stealing
Network traffic usage and analysis	Email or instant message data stealing
Packet conversion for data analysis	Packet spoofing and data theft
Network troubleshooting	Monetary or reputational damage

Many types of sniffer attacks like A LAN sniff, A protocol sniff, An ARP sniff, TCP session stealing, Application-level sniffing, Web password sniffing.

III. TYPES OF IDPS TECHNOLOGIES: [2]

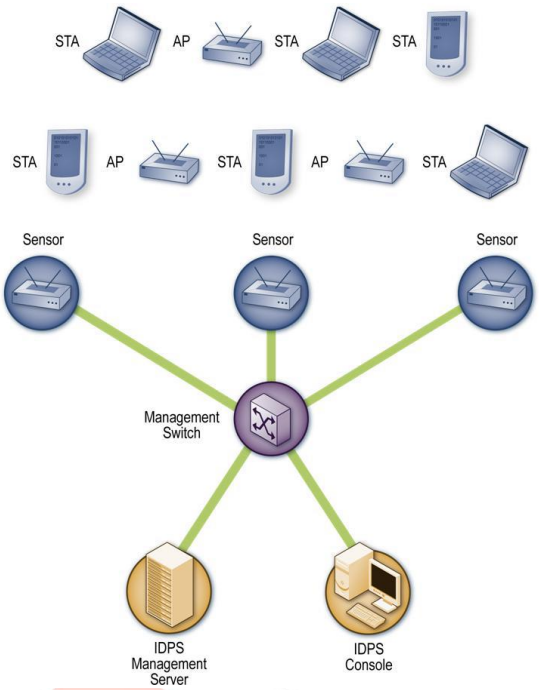
The types of IDPS technologies are differentiated primarily by the types of events that they monitor and the ways in which they are deployed. This publication discusses the following four types of IDPS technologies:

- **Network-Based(NIDS)**, which monitors network traffic for particular network segments or devices and analyzes the network and application protocol activity to identify suspicious activity. Types of IDPS Technologies



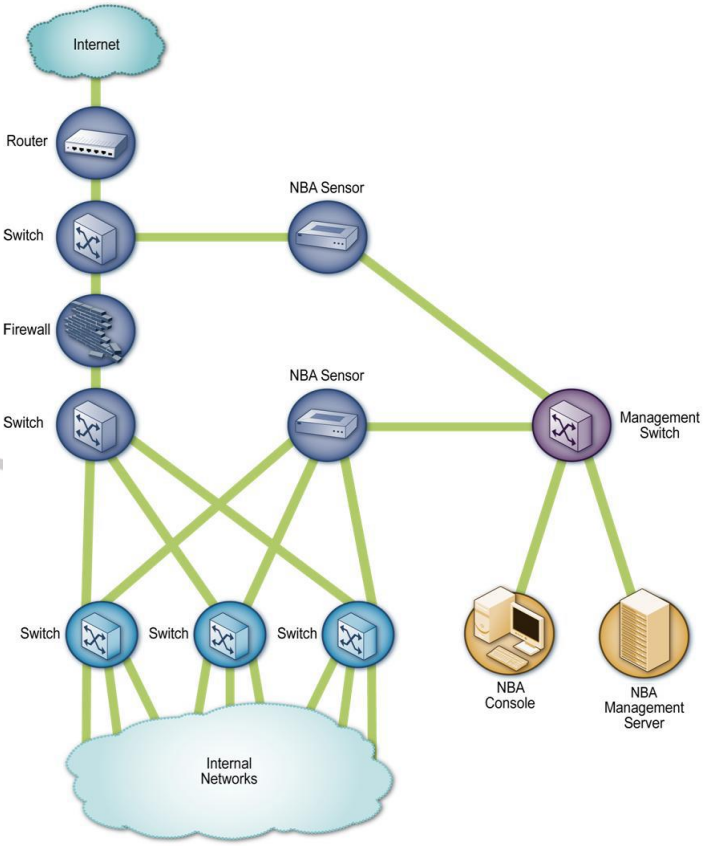
Network-Based IDPS Sensor Architecture Example [2]

- **Wireless**, which monitors wireless network traffic and analyzes it to identify suspicious activity involving the wireless networking protocols themselves. It cannot identify suspicious activity in the application or higher-layer network protocols (e.g., TCP, UDP) that the wireless network traffic is transferring. It is most commonly deployed within range of an organization's wireless network to monitor it, but can also be deployed to locations where unauthorized wireless networking could be occurring.



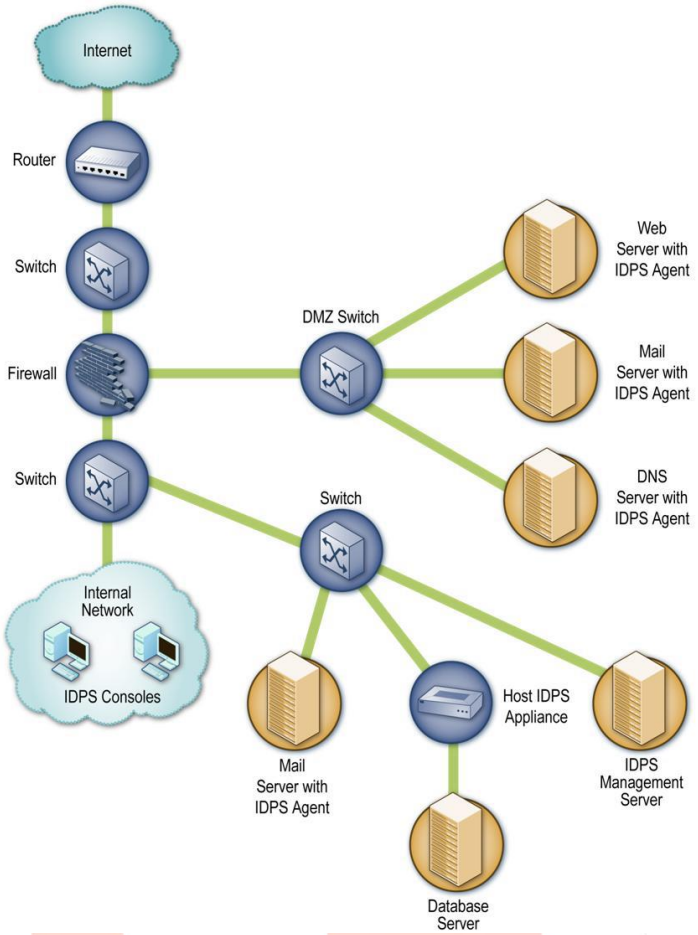
Wireless IDPS Architecture [2]

- **Network Behavior Analysis (NBA)**, which examines network traffic to identify threats that generate unusual traffic flows, such as distributed denial of service (DDoS) attacks, certain forms of malware, and policy violations (e.g., a client system providing network services to other systems)



NBA Sensor Architecture Example [2]

- **Host-Based (HIDS)**, which monitors the characteristics of a single host and the events occurring within that host for suspicious activity. Examples of the types of characteristics a host-based IDPS might Monitor is network traffic (only for that host), system logs, running processes, application activity, file access and modification, and system and application configuration changes. Host-based IDPSs are most commonly deployed on critical hosts such as publicly accessible servers and servers containing sensitive information.



Host-Based IDPS Agent Deployment Architecture Example [2]

IV. DETECTION METHODOLOGIES:[2]

Most IDPSs use multiple detection methodologies, either separately or integrated, to provide more broad and accurate detection. The primary classes of detection methodologies are as follows:

- **Signature-based**, which compares known threat signatures to observed events to identify incidents. This is very effective at detecting known threats but largely ineffective at detecting unknown threats and many variants on known threats. Signature-based detection cannot track and understand the state of complex communications, so it cannot detect most attacks that comprise multiple events.
- **Anomaly-based detection**, which compares definitions of what activity is considered normal against observed events to identify significant deviations. This method uses profiles that are developed by monitoring the characteristics of typical activity over a period of time. The IDPS then compares the characteristics of current activity to thresholds related to the profile. Anomaly-based detection methods can be very effective at detecting previously unknown threats. Common problems with anomaly-based detection are inadvertently including malicious activity within a profile, establishing profiles that are not sufficiently complex to reflect real-world computing activity, and generating many false positives.
- **Stateful protocol analysis**, which compares predetermined profiles of generally accepted definitions of benign protocol activity for each protocol state against observed events to identify deviations. Unlike anomaly-based detection, which uses host or network-specific profiles, Stateful protocol analysis relies on vendor-developed universal profiles that specify how particular protocols should and should not be used. It is capable of understanding and tracking the state of protocols that have a notion of state, which allows it to detect many attacks that other methods cannot. Problems with Stateful protocol analysis include that it is often very difficult or impossible to develop completely accurate models of protocols, it is very resource-intensive, and it cannot detect attacks that do not violate the characteristics of generally acceptable protocol behavior.

Comparison of IDPS Technology Types [2]

Technology Type	Types of Malicious Activity Detected	Scope per Sensor or Agent	Strengths
Network-Based	Network, transport, and application TCP/IP layer activity	Multiple network subnets and groups of hosts	Able to analyze the widest range of application protocols; only IDPS that can thoroughly analyze many of them

Wireless	Wireless protocol activity; unauthorized wireless local area networks (WLAN) in use	Multiple WLANs and groups of wireless clients	Only IDPS that can monitor wireless protocol activity
NBA	Network, transport, and application TCP/IP layer activity that causes anomalous network flows	Multiple network subnets and groups of hosts	Typically more effective than the others at identifying reconnaissance scanning and DoS attacks, and at reconstructing major malware infections
Host-Based	Host application and operating system (OS) activity; network, transport, and application TCP/IP layer activity	Individual host	Only IDPS that can analyze activity that was transferred in end-to-end encrypted communications

V. OPEN SOURCE INTRUSION DETECTION TOOLS

Snort: Snort is a free and open source network intrusion prevention system (NIPS) and network intrusion detection system (NIDS) This network intrusion detection and prevention system excels at traffic analysis and packet logging on IP networks. Through protocol analysis, content searching, and various pre-processors, Snort detects thousands of worms, vulnerability exploit attempts, port scans, and other suspicious behavior. Snort uses a flexible rule-based language to describe traffic that it should collect or pass, and a modular detection engine. Also check out the free Basic Analysis and Security Engine (BASE), a web interface for analyzing Snort alerts.

OSSEC: is an Open Source Host-based Intrusion Detection System. It performs log analysis, integrity checking, Windows registry monitoring, rootkit detection, real-time alerting and active response. It runs on most operating systems, including Linux, OpenBSD, FreeBSD, Mac OS X, Solaris and Windows. In addition to its IDS functionality, it is commonly used as a SEM/SIM solution. Because of its powerful log analysis engine, ISPs, universities and data centers are running OSSEC HIDS to monitor and analyze their firewalls, IDSs, web servers and authentication logs.

OSSIM: OSSIM stands for Open Source Security Information Management. Its goal is to provide a comprehensive compilation of tools which, when working together, grant network/security administrators with a detailed view over each and every aspect of networks, hosts, physical access devices, and servers. OSSIM incorporates several other tools, including Nagios and OSSEC HIDS

Sguil: is built by network security analysts for network security analysts. Sguil's main component is an intuitive GUI that provides access to realtime events, session data, and raw packet captures. Sguil facilitates the practice of Network Security Monitoring and event driven analysisl.

ArcSight SIEM platform: ArcSight provides a suite of tools for SIEM—security information and event management. The best-known seems to be ArcSight Enterprise Security Manager (ESM), described as the "brain" of the SIEM platform. It is a log analyzer and correlation engine designed to sift out important network events. The ESM itself is a standalone appliance, and the management programs run on Linux, Windows, AIX, and Solaris. For open-source alternatives see OSSEC HIDS and OSSIM.

Honeyd: is a small daemon that creates virtual hosts on a network. The hosts can be configured to run arbitrary services, and their TCP personality can be adapted so that they appear to be running certain versions of operating systems. Honeyd enables a single host to claim multiple addresses on a LAN for network simulation. It is possible to ping the virtual machines, or to trace route them. Any type of service on the virtual machine can be simulated according to a simple configuration file. It is also possible to proxy services to another machine rather than simulating them. It has many library dependencies, which can make compiling/installing Honeyd difficult.

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.

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.

Other open source tools like Kismet, OpenDLP, Suricata, AIDE, and TRIPWIRE are also used for creating intrusion detection and prevention System.

VI. CONCLUSION

Network security is primary and important of any organization. Using IDPS and IDS We understand and detect intrusion atomically by using Intrusion detection tools protect their home or organization from several types of attacks. Open Source Intrusion Detection tools allows the users customize installation as per their security requirement. Each Intrusion Detection System Tools have their own advantages and disadvantages, choosing the best one depend on open source tools characteristic we choose best tools and used in our infrastructure and successfully detect or prevent intrusion.

REFERENCES

[1] NIST, Guide to Intrusion Detection and Prevention Systems (IDPS)
[2] NIST Special Publication on Intrusion Detection
[3] DOUGLAS J. BROWN, BILL SUCKOW, and TIANQIU WANG, "A Survey of Intrusion Detection Systems"
[4] Rebecca Bace and Peter Mell. NIST Special Publication on Intrusion Detection Systems," 16 August 2001.
[5] Eleazar Eskin. Anomaly Detection over Noisy Data Using Learned Probability Distributions," Proceedings of the Seventeenth International Conference on Machine Learning (ICML-2000), Palo Alto, California, July 2000.
[6] Anup K. Ghosh, James Wanken, and Frank Charron. \Detecting Anomalous and Unknown Intrusions Against Programs," Annual Computer Security Applications Conference (ACSAC'98), Scottsdale, Arizona, 7-11 December 1998.
[7] Karthikeyan .K.R and A. Indra, "Intrusion Detection Tools and Techniques – A Survey" International Journal of Computer Theory and Engineering, Vol.2, No.6, December, 2010 1793-8201
[8] A Survey of Intrusion Detection systemsy. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
[9] SURYA BHAGAVAN AMBATI, DEEPTI VIDYARTHI, "A BRIEF STUDY AND COMPARISON OF, OPEN SOURCE INTRUSION DETECTION SYSTEM TOOLS" International Journal of Advanced Computational Engineering and Networking, ISSN: 2320-2106, Volume-1, Issue-10, Dec-2013
[10] OSSEC website, http://www.ossec.net/, 30 Oct 2013
[11] SNORT website , http://www.snort.org , 30 Oct 2013
[12] Tripwire website http://www.tripwire.com,30 Oct 2013
[13] Martin Roesch, "SNORT – Light weight Intrustion detection for networks",Proceedings of LISA '99: 13th Systems Administration Conference Seattle, Washington, USA, November 7–12, 1999