INTRUSION DETECTION SYSTEMS. TECHNIQUES AND TOOLS.

Assoc. Prof. Sorin Popa Ph. D
University of Craiova
Faculty of Economics and Business Administration
Craiova, Romania
Prof. Nicolaie Giurgiteanu Ph. D
University of Craiova
Faculty of Economics and Business Administration
Craiova, Romania

Abstract: Attacks on the computer infrastructures are becoming an increasingly serious problem.

There are available several information security techniques and tools to protect valuable information stored on computer systems against unauthorized access, use and destruction.

This paper offers a perspective on the current state of Intrusion Detection Systems (IDS) describing the functionalities and components of such systems. A classification based on the type of protected system is made, along with the particular use of each of them. Further, it investigates the main techniques used to detect intruders, revealing the modality for discovering abnormal system events. Finally, the focus is on different types of intrusion detection tools available, few examples of them being presented.

JEL classification: C83, C88

Key words: intrusion detection, abnormal events, audit trail analysis, real-time analysis

1. INTRODUCTION

Internet and internal attacks on enterprise networks continues to increase. These attacks range from relatively simple to sophisticated techniques exploiting security vulnerabilities [7].

At the same time the diversity of operating system platforms, routers, network protocols, applications, web servers, databases, etc., has increased the complexity of enterprises network. So, trying to spot an attack becomes extremely difficult.

As result, intrusion events to computer systems and maintaining proper network security it's nearly impossible without using sophisticated techniques and tools to detect unauthorized access.

An intrusion event is defined as any set of actions that attempt to compromise the integrity, confidentiality or availability of a resource.

In a computer network, intrusion detection is the task of detecting, analyzing, monitoring and responding to any intrusion event, internal or external, that may result in a security threat.

Typically, responses to an unauthorized access to a computer network can be made automated, by notifying a security administrator via a console, e-mail, pager; stopping the offending session; shutting the system down; turning off down Internet links; disabling users; or executing a predefined command procedure.

Therefore, Intrusion Detection Systems (IDS) are required as an additional control for protecting network systems, complementing other preventive controls (e.g. firewalls) as the next line of defense within the organization [9].

2. IDSs FUNCTIONALITY

An IDS is a device, or a program, placed inside a protected network that collect information from a variety of system and network sources, and then analyze the information for signs of intrusion and misuse [10].

Monitoring activity to identify malicious or suspicious alerts, an IDS can be compared with a spam filter that raises an alarm if specific things occur [9].

An IDS offers the opportunity to detect an attacker that is able to pass through the router and pass through the firewall. The detection can take place at the beginning of the attack, during the attack, or after it has occurred. IDS activate an alarm, which can take defensive action [7].

In order to enhance its effectiveness, an IDS system needs to limit false positives - incorrectly identifying an attack when there is none, and false negatives - an activity is intrusive but it is not reported as intrusive because it is not anomalous. This means that while the number of false alarms should be reduced, real attacks should not go unnoticed.

The degrees of false positives and false negatives together represent the sensitivity of the system.

The goal of IDS is to accurately detect anomalous network behaviour or misuse of resources (i.e. incidents), sort out true attacks from false alarms, and notify network administrators of the activity.

Given the goal of an IDS, its functions can be [9]:

- Monitoring users and system activity;
- Auditing system configuration for vulnerabilities and misconfigurations;
- Assessing the integrity of critical system and data files;
- Recognizing known attack patterns in system activity;
- Identifying abnormal activity through statistical analysis;
- Managing audit trails and highlighting user violation of policy or normal activity;
- Correcting system configuration errors;
- Installing and operating traps to record information about intruders.

An IDS may embody one or more of these functions, depending on the type of IDS.

2.1 IDS components

According to the Common Intrusion Detection Framework (CIDF), an IDS consists of four functional components [11]:

1. information sources (event generators) - provides a stream of event records and can be drawn from different levels of the target system. The most common sources (or observation points or sensors) in intrusion detection, are networks, hosts, and applications. These sensors normally generate a lot of alerts, all of them are analyzed, and only the relevant alerts (those that have a high enough value to be considered a security relevant system event) are reported as incidents.

2. analyzers - use the output of sensors, modeling and analyzing the collected data events, to decide whether those events indicate signs of intrusions. Today, artificial intelligence has become an indispensable tool in the analyzers of IDS.

3. incident databases - stores instances of all alerts, supporting the analysis process.

4. response units - carry out a set of actions controlled by the analyser, that instructs them to act when an intrusion is detected [11]. The actions can be passive measures, such as reporting intrusion alerts to administrators, or active measures, such as blocking intrusions.

The overview of this process is depicted in Figure 1.

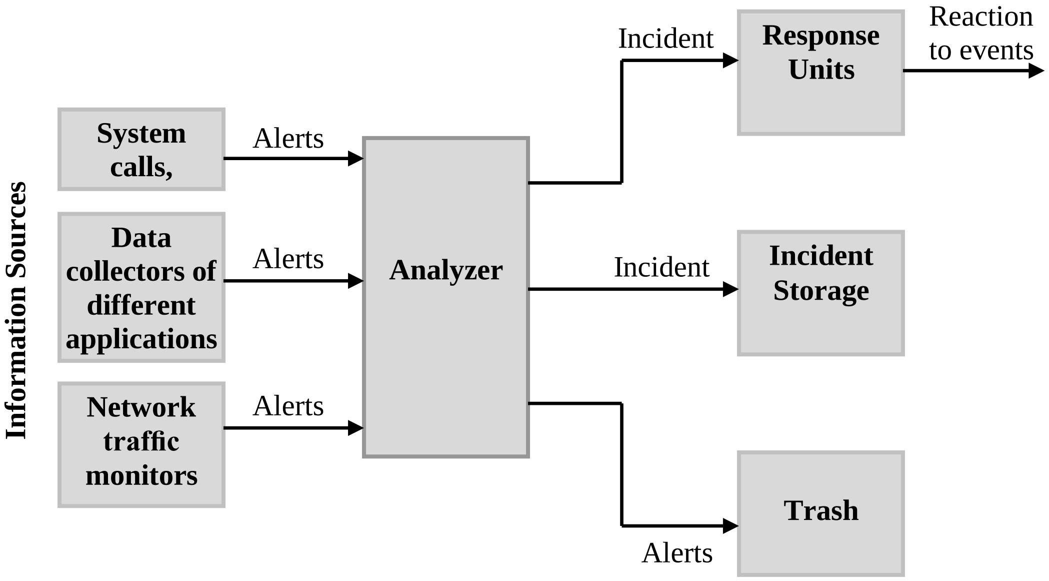

Figure no. 1 IDS components

2.2 Classification of IDSs

Based on the type of protected system or data source (event generators), there are two main types of IDSs: host-based IDSs (HIDS) and network-based IDSs (NIDS) [8].

Host-based IDS

A HIDS runs on an individual computer system (host) and is used to analyze data that originates on the computer, such as application and operating system event logs, in order to detect attacks and protect that specific computer (Figure 2).

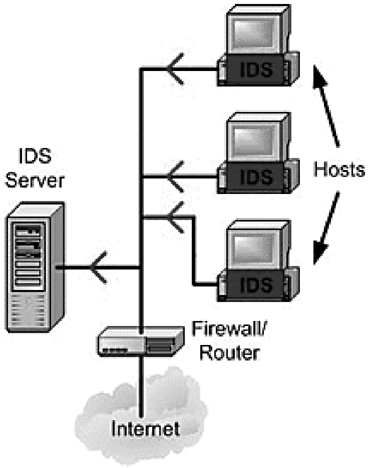

Figure no. 2 Host-based IDS

Host-based intrusion detection is particularly effective at detecting insider misuse because of the target data source’s proximity to the authenticated user [10].

This allows HIDSs to provide a comprehensive analysis of the system activities, to determine exactly which processes and users are involved in a particular attack on the system.

Network-based IDS

A NIDS resides on the edge of a network and monitor packets traffic throughout that network, processing data and trying to detect suspicious activities or patterns, and alerting system administrators when potential hostile traffic is detected (Figure 3).

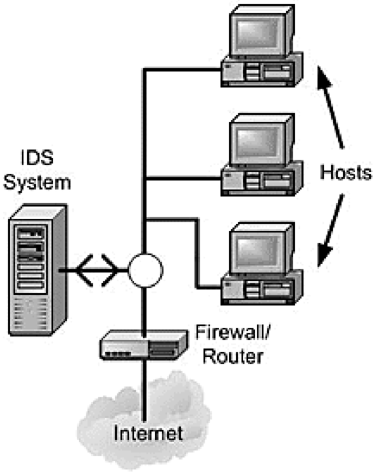

Figure no. 3 Network-based IDS

Instead of analyzing information that originates and resides on a computer, NIDS uses techniques like “packet-sniffing” to pull data from TCP/IP or other protocol packets traveling along the network and derived straight from the output of routers and switches. This surveillance of the connections between computers makes NIDS great at detecting access attempts from outside the trusted network.

A NIDS can be used to protect multiple hosts that are connected to a network segment, or a switch, which is monitored by the IDS. However, network-based IDSs may have difficulty processing all packets in a large or busy network, and therefore they may fail to recognize an attack launched during periods of high traffic.

Typically, a NIDS performs deep incoming packets inspection and tries to find suspicious patterns known as signatures. The signatures are represented as a set of rules that are frequently updated by the security community. When a packet is matched against a signature, an alert is raised, indicating an attempted intrusion or misuse.

3. IDS TECHNIQUES

For each of the two types of IDS, there are four basic techniques used to detect intruders: anomaly detection, misuse detection (signature detection), target monitoring, and stealth probes.

3.1 Anomaly intrusion detection

Anomaly intrusion detection technique uses the usage behavior of an object, defined in patterns of activities that appear to be normal. Any significant deviation from that behaviour (i.e. an abnormal, or unusual behaviour that lies outside of the patterns) is considered as a possible attack, and it is reported as a possible intrusion.

Anomaly intrusion detection involves in first building a pattern of normal behaviour of a monitored object, and then monitoring the object’s actual behaviour to detect significant deviations from normal behaviour recorded in the pattern. The monitored objects can be users, programs, network traffic or other resources in a system.

A normal usage object’s pattern is constructed from the statistical measures over historical data collected over a period of normal operation of the object.

The main advantage of anomaly intrusion detection technique is that it has the potential to detect novel and unknown attacks without advance knowledge about these attacks.

3.2 Misuse intrusion detection

Also called signature detection, this technique uses specifically well-known patterns of unauthorized behavior to predict and detect subsequent similar attempts. These specific patterns, called signatures, are encoded in advance and used to match against the user behavior to detect intrusion.

This technique first defines signatures of known intrusions, and then monitors current activities for such signatures, and reports matched cases. An intrusion signature specifies a suspect set of sequences of actions or events that lead to an attack or misuse.

Intrusion signatures are not only useful to detect intrusions but also to discover intrusion attempts. A partial satisfaction of an intrusion signature may indicate an intrusion attempt. The knowledge of past intrusions can be encoded into expert system rules.

Despite this approach is very effective at detecting known attacks whose signatures have been encoded into rules, without generating an overwhelming number of false alarms, it isn’t effective to detect novel, or unknown attacks.

3.3 Target Monitoring

The systems using this technique do not actively search for anomalies or misuse, but instead look for the modification of specified files (e.g. mission/system critical files). This corrective technique, is designed to uncover an unauthorized action after it occurs in order to reverse it. One way to check for the unauthorized editing of files is by computing an integrity checksum hashes beforehand and comparing this to new hashes of the file at regular intervals. This type of system is the easiest to implement, because it does not require constant monitoring by the administrator.

3.4 Stealth Probes

This technique attempts to detect any attackers that choose to carry out their mission over prolonged periods of time. Stealth probes collect a wide-variety of data throughout the system, checking for any methodical attacks over a long period of time. They take a wide-area sampling and attempt to discover any correlating attacks. In effect, this method combines anomaly detection and misuse detection in an attempt to uncover suspicious activity.

4. TYPES OF INTRUSION DETECTION TOOLS

Intrusion detection systems can run on either a periodic or continuous feed of information and hence they use two different intrusion detection approaches:

1. Audit trail (interval-based) analysis
2. Real-time analysis

4.1 Audit Trail Analysis

An audit trail is a series of records of system events, about an application, network traffic or user activities. If audit trails have been designed and implemented to record appropriate information, and in conjunction with appropriate tools and procedures, they can assist in intrusion detection. A typically kind of audit record used in intrusion detection is an event-oriented log, which usually contain records describing system events, application events, or user events.

Network monitoring and traffic analysis are certainly useful as an auditing method, and they can permit to reduce the chance that a successful intrusion will occur.

In order to perform basic network monitoring, is need to collect traffic information for a certain period of time, the volume of collection depending on the volume of traffic and resources consumption. Recording every event possible means a noticeable consumption of system resources (both the local system and network involved). Specifying which events are to be audited is also difficult because certain types of attacks may pass undetected.

Using a statistical approach the amount of data collected will be then analyzed, examining elements like source and destination IP addresses, the source and destination ports for TCP or UDP traffic, ICMP messages. Any suspicious or unauthorized activity will be reported.

Tools for audit trial analysis have the advantage that investigations can go back in time and do historical analysis of events that have occurred in the past. More sophisticated tools can also perform automated audit trail and management, show trend analysis by attack category, system, type of system, etc.

4.2 Real-time analysis

With real-time analysis, an IDS performs online verification of system events. Generally, a stream of network packets is constantly monitored constantly.

With this type of processing, intrusion detection uses the knowledge of current activities over the network to sense possible attack attempts.

Real-time analysis IDS tool requires a large amount of RAM (buffers) since no data storage is used. Therefore, the amount of data collected by the detector is small; such an IDS may sometime miss packets, because realistic processing of too many packets is not available. Only selective packets in a data stream get inspected, and the inspection process only looks for "state" information, such as whether a packet contains malicious code.

One method to perform real-time analysis intrusion detection is to use a packet sniffer in "promiscuous mode" so it can read and analyze all network segment traffic. It does this by examining both the packet header fields and packet contents, looking for specific types of network attacks, such as IP spoofing and packet floods. When the sniffer detects a potential problem it reponds immediately by notifying a console. The advantage of this method is that there are certain network-oriented attacks (IP spoofing, packet storms, etc.) that are best detected via packet examination.

Because a packet sniffer monitors traffic only the links, not the network nodes, the next level for real-time analysis intrusion detection tools is to monitor security-related activity occurring on the various systems and devices that make up the network. Monitoring for attacks from both the inside and the outside the network becomes much easier, since all of the devices are being watched.

When suspicious activity is detected, like attempts to access unauthorized critical files, or when a user illegally obtains administrator access, the real-time activity monitor can take immediate action before damage is done. This action typically includes notifying a console, sending e-mail, disabling a user account, terminating the intruder's process, terminating the intruder's session, shutting the system down, or executing a command procedure.

Activity monitors include two basic categories: single system and manager/agent [4].

A single system activity monitor runs on only one system in the network and detects intrusions based solely on what it finds on that system. Such an intrusion detection tool is installed and executed on a stand-alone, system-by-system basis.

A manager/agent intrusion detection solution has agents covering systems and network devices throughout the enterprise. These agents are connected to various managers, which are in turn connected to an intrusion detection console[4].

From the console the monitoring activity can be done throughout the entire network and the suspicious activity can be corellated as it occurs in multiple locations in the network.

4.3 Examples of IDS tools

Wireshark

One of the best open source netwok packet analyzer available today, that provides a graphical user interface and traffic statistics, is Wireshark.

As a “de facto” standard across many industries and educational institutions, Wireshark has a rich feature set which includes the following:

- multi-platform - runs on Windows, Linux, OS X, Solaris, FreeBSD, NetBSD, and many others;
- capture live packet data from a network interface and perform offline analysis;
- display packets with very detailed protocol information;
- import and export packet data from and to a lot of other capture programs;
- filter packets on many criteria;
- decryption support for many protocols, including IPsec, ISAKMP, Kerberos, SNMPv3, SSL/TLS, WEP, and WPA/WPA2;
- create various statistics;
- output can be exported to XML, PostScript®, CSV, or plain text.

Snort

Snort is a free and open source packet sniffer and logger that can be used as a lightweight Network Intrusion Detection System (NIDS), capable of performing packet logging and real-time traffic analysis on IP networks. It can monitor all of the traffic on a network link to perform protocol analysis, content searching/matching, and is commonly used to actively block or passively detect, a variety of attacks and probes, such as buffer overflows, stealth port scans, web application attacks, and more.

For a PC-based solution, Snort can run on the top of many platforms such Linux, FreeBSD, and Windows.

Snort can be combined and integrated in third-party free software solutions such as Sguil, OSSIM, and the Basic Analysis and Security Engine (BASE) to provide a visual representation of intrusion and attack data, for convenient analysis.

Snort has a "rule base" that contains patterns or "signatures" of malicious traffic, "rule base" that can be updated. In order to minimize false alerts, the user can choose which rules in Snort's rule base he wants to use and which to ignore, to adapt Snort to a specific network environment.

There are four modes that Snort can be configured to operate:

1. Sniffer mode, which simply reads the packets off of the network and displays them in a continuous stream on the console (screen).

2. Packet Logger mode, which logs the packets to disk.

3. Network Intrusion Detection System (NIDS) mode, the most complex and configurable configuration, which allow Snort to analyze network traffic for matches against a user-defined rule set and performs several actions based upon what it sees.

4. Intrusion prevention system (IPS) is a newly added feature and allows Snort to take preventive actions against malicious traffic such as dropping or re-directing packets to another destination.

Sax2

Sax2 is an intrusion detection and response system that performs real-time packet capturing, 24/7 network monitoring, advanced protocol analyzing and automatic expert detection.

Sax2 makes it easy to isolate and solve network problems, identify network bottleneck and bandwidth use, detect network vulnerabilities and discovered the network whether there is a breach of security strategy and the signs of being attacked in the network of hazard, and then intercept and stop before their invasion.

Network administrators can directly monitor http requests, email messages, ftp transfers, as well as real-time activities and message details for the two popular instant messengers: MSN and QQ.

The key features of Sax2 are:

- intrusion detection and prevention;
- conduct of audits;
- traffic statistics;
- customize security policy;
- real-time alert and response.

5. Conclusions

Intrusion detection is critical in today's complex information systems, the investigation of abnormal security events being mandatory.

There is multitude of available data from computer system and network traffic that can be used for determining how well an access policy is implemented and if vulnerabilities exist on the network.

With appropriate procedures and tools this data can be turn into valuable information for intrusion detection.

References

1. S. Axelsson — Intrusion Detection Systems: A Taxonomy and Survey. Technical Report No 99-15, Dept. of Computer Engineering, Chalmers University of Technology, Sweden, March 2000, http://www.ce.chalmers.se/staff/sax/taxonomy.ps
2. R. Bejtlich — The Tao of network security monitoring. Addison Wesley, 2004.
3. M. Botha, R. von Solms — "Utilizing fuzzy logic and trend analysis for effective intrusion detection", Computers & Security, volume 22, pages 423–434, 2003.
4. R. A. Clyde — "Intrusion Detection Methodologies", A White Paper, AXENT Technologies, Inc., 2001
5. D. Elson — Intrusion Detection, Theory and Practice, March 27, 2000, http://online.securityfocus.com/infocus/1203
6. K .K. Frederick — Network Intrusion Detection Signatures, December 19, 2001, http://online.securityfocus.com/infocus/1524
7. T. Jackson, J. Levine, J. Grizzard, H. Owen — "An investigation of a compromised host on a honeynet being used to increase the security of a large enterprise network", Proceedings of the 2004 IEEE Workshop on Information Assurance and Security, IEEE, 2004.
8. S. Kumar, — Classification and detection of computer intrusions. A Ph.D. Thesis. Purdue University, 1995, http://ftp.cerias.purdue.edu/pub/papers/sandeep-kumar/kumar-intdet-phddiss.pdf.
9. C. Pfleeger, S. Pfleeger — Security in computing. Prentice Hall, 2003.
10. P. Proctor — The practical Intrusion Detection Handbook. Prentice Hall, 2001.
11. S. Staniford-Chen, B. Tung, P. Porras, C. Kahn, D. Schnackenberg, R. Feiertag, M. — The Common Intrusion Detection Framework - Data Formats. IETF, 1998, http://mirrors.isc.org/pub/www.watersprings.org/pub/id/draft-staniford-cidf-data-formats-00.txt.

	Stillman	
12.	M. Stillerman, C. Marceau, M. Stillman	“Intrusion detection for distributed applications”, Communications of the ACM, volume 42, pp. 62–69, 1999.
13.	* * *	www.snort.org
14.	* * *	www.wireshark.org
15.	* * *	www.ids-sax2.com