Article can be accessed online at http://www.publishingindia.com

Intrusion Detection Systems: A Review

Kajal Rai*, M. Shyamala Devi**

Abstract

Protection of computer resources and stored documents is an important issue in today's world. Intruders have made many winning attempts to overthrow prestigious company networks. Although the current security solutions such as firewalls, and anti-virus software have their significant roles in securing organizations but they do not detect all types of attacks of today's cyber world. Intrusion detection is a mechanism used to detect various attacks on a network. There are many Intrusion detection Systems (IDSs) available today. Some of them are open source while some are commercially used. In this paper we give a brief introduction to open source IDSs: SNORT and BRO. Some of the common types of attacks on various layers of TCP/IP and how SNORT and BRO handle these attacks are discussed. Current research in intrusion detection is also included. It is concluded that intrusion detection is a challenging task due to the advent of many unknown attacks.

Keywords: Intrusion Detection, Snort, Bro

1. Introduction

Intrusion Detection Systems are systems that monitor computer system events to discover the malicious or suspicious activities in the system and issue alarm when such thing happens. In today's enterprise environment, harmful activities are increasing day-by-day. Therefore IDSs have become an important and essential part of the organizations.

1.1. Need for Intrusion Detection

In all types of network, security is a primary issue especially in big organizations as they have very important and confidential data which if get hacked will bring down company's profile (Rafeeq, 2003). Generally, we secure our systems by building firewalls or employ some authentication mechanisms such as passwords or some encryption techniques which create a protective covering around them. All the above techniques provide a level of security but they cannot give protection against malicious codes, inside attacks or unsecured modems. We need more security mechanisms such as IDS because firewalls cannot detect attacks inside the network since they are mostly deployed at the boundary of the network, and thus only control traffic entering or leaving the network. But a huge percentage of intrusions may be from within the network and IDS can monitor and analyze various events in the network and if the system has been misused it gives immediate report to the administrator.

1.2. Categories of Intrusion Detection Systems

Based on the sources of audit information IDS can be classified into two categories namely: Host-based and Network-based.

Host-based IDS (HIDS): HIDS detects the attacks against a single host. It is deployed on the systems which are more vulnerable to attacks such as web server. To perform intrusion detection HIDS gathers information from its system calls, operating system audit trails, application logs, etc. Tripwire and OSSSEC are some of the examples of HIDS.

* Research Scholar, Department of Computer Science and Applications, Panjab University, Chandigarh, India.
E-mail: kajalrai.pu@gmail.com
** Professor, Department of Computer Science and Applications, Panjab University, Chandigarh, India.
E-mail: syamala@pu.ac.in

Network-based IDS (NIDS): NIDS detects attacks on various systems in the network. NIDS uses network traffic for detecting suspicious activities in the network to prevent illegal access to network resources. Examples of NIDS are Cisco Secure IDS, BRO, SNORT, Dragon, etc.

On the basis of analyzing events there are mainly two types of approaches, one is misuse based and another is anomaly based.

Misuse based IDS: It uses various matching techniques to find a match between system activities and already known attack signatures stored in the database. It is also called Signature-based approach.

Advantages:

- It detects attacks without signaling false positive alarms.
- It supplies an easy to use tool to monitor systems so that decision about attacks can be taken easily.

Disadvantages:

- Misuse based IDS cannot detect new attacks.
- The systems must be updated every time when a new attack is exposed.
- Misuse detectors are intended to notice attacks that have signatures already known to the system. When a well-known attack is modified a little and a variant of that attack comes, they are not capable to detect them.

Anomaly based IDS: Anomaly based IDS detects the behaviors of user and system activities. First, It create profiles of users, servers and all the network connections using the known normal behaviors and after that generates alarm if the new data deviates from the data already stored in the database of user and system profiles.

Advantages

- They are able to detect novel attacks.
- Even when complete information of the attack does not exists then also anomaly-based IDSs are able to detect attacks.
- They can be used to acquire signature information used by signature-based IDS.

Disadvantages

- Anomaly-based IDS signals a large number of false positives as behaviors of users and network are not always known in advance.
- This approach requires a huge set of training data for setting the thresholds for normal activities of the systems.

2. Common Attacks on Various Layers of TCP/IP

TCP/IP model consists of five layers namely Physical layer, Data link layer, Network layer, Transport layer and Application layer. The layers which are most vulnerable to attacks are network, transport and application layers. The most common attacks that occur on these layers are discussed below.

2.1. Network Layer Attacks

There are various attacks on network layer such as eaves dropping, IP spoofing, Denial-of-Service, Man-in-the-middle attack, and Sniffer attack. The most common attacks among these and which does huge damage to the system resources are IP spoofing and Denial-of-Service attack.

IP Spoofing: In IP Spoofing the intruder sends messages with forge IP address to the target host by giving an impression that the messages are coming from a trusted host to get illegal access to the target host or other nodes in the network. The intruder first applies various techniques to locate an IP address of a trusted host in the network and then alters the packet headers so that it seems that the trusted host is sending the packets (Slideshare, 2012).

Denial-of-Service attack (DoS): DoS attack is an attempt to make the system or network resources inaccessible to its intended users. There are many possible ways to make DoS attacks such as improper use of bandwidth, and disk space or resetting the state information of TCP sessions or by SYN Floods. Dos attack is very prominent on network layer but it can be made on transport layer and application layer also.

2.2. Transport Layer Attacks

There are numerous attacks on transport layer such as SYN-Floods attack, SSL man-in-the-middle attack, TCP connection hijacking, UDP flood attack, and Port scan attack. Among all these SYN-Floods attack and TCP connection hijacking are the most commonly occurring attacks.

SYN-Floods: When a client wants to establish a TCP connection with the server, then the client will have to go through the three-way handshake process. In this process, first the client requests the server by sending a synchronized message known as SYN. After that the server replies to the client by sending acknowledgement message known as SYN-ACK. Finally the client sends ACK to the server and the connection established. DoS attack is made by flooding the target host with forged connection requests by sending thousands of SYN. This is known as SYN Floods. Because of this attack the server reaches up to its limit to the number of clients it can reply and many authorized requests will not be responded (Wesley, 2006).

TCP Connection hijacking attack: TCP Connecting hijacking is also known as Man-in-the-Middle attack of transport layer. This attack occurs when the authentication between hosts has completed. The attacker takes control of the connection and sends forged packets to one of the hosts to retrieve important information. The sequence numbers of packets from an ongoing connection are acquired and forged packets are sent with the next sequence number (Jon, 2008).

2.3. Application Layer Attacks

Nowadays application layer is being the common target of attackers. Scripting vulnerabilities, buffer overflows, cookie poisoning, hidden field manipulation, parameter tampering, cross-site scripting, SQL injection, etc, are the various attacks on application layer and the most commons of these are SQL injection and cross-site scripting attacks.

SQL Injection attack: A SQL injection attack is an attack in which the attacker captures the original Structured Query Language (SQL) commands and inserts some code in SQL to make changes in the database. An active SQL injection attack can read sensitive data from the database, modify the contents of the database, and can carry out illegal operations on the database.

Cross-site scripting (XSS) attack: XSS is an attack which is commonly found in web application. The attacker will add client-side script into the web pages and the other web users would see the changed web pages.

3. Open Source Intrusion Detection Systems

There are many open source intrusion detection systems available. BRO, SNORT, Suricata, etc. are some examples. Here, we discuss briefly Snort and Bro open source intrusion detection systems.

3.1. BRO Intrusion Detection System

Bro system was designed and developed by Vern Paxson of ICSI's Center for Internet Research (ICIR). The project is started in 1995 at Lawrence Berkeley National Laboratory (LBL). It was available first time in 1998 (Sommer, 2003). Bro is a freeware, Unix-based NIDS that monitors network traffic to find malicious activity. It is a signature-based IDS. It comes under the BSD (Berkeley Software Distribution) license. It is a connection based approach which keeps internal virtual connection state for every connection, defined for all network packets (Miguel, 2008). It supports many application layer protocols including DNS, FTP, HTTP, SMTP, etc.

Working of BRO

Step1: Packets are captured from a network interface.

Step2: Raw packets are ordered into streams of data that are put into protocol specific "engines" inside the core for handling parsing of the specific protocol.

Step3: Events of interest in the protocol are acted upon by analyzers for that protocol.

Step4: (a) Real-time logs are written to disk that are categorized according to protocol.

(b) Alerts are generated for display.

The first step is to capture network packets and Bro does this by using LIPCAP which is a library for packet capturing, and then these packets are passed into the Bro core, where the packets are grouped into streams of data for analysis. This stream then passes into protocol decoder engine. The engine manages HTTP traffic based on protocol verification checks and port checks. Then the data stream is passed to the central part to the HTTP

decoder. Finally, the analyzers conclude what information is produced from the core associated to the particular protocol. This information then performs actions, such as writing the information to the log file, etc. The administrator can see which connections are creating problems in the network by looking these logs files (Ryan, 2003). The most important components of BRO are the analyzers. There are analyzers for application-layer decoding, anomaly detection, misuse detection and connection analysis. Bro is able to detect many attacks; some of the frequently occurring attacks handled by Bro are Stepping stone attack and SYN-flood attack.

Handling Stepping Stone attack: In stepping stone, an intermediary host is used by an attacker to hide their origin to access a system. For example, instead of connecting from host A to B directly, he may connect to intermediate host C first, and then from C to B. Bro addresses the problem by correlating the timing of packets on incoming connections with those on outgoing connections.

Handling SYN-Floods: By default Bro creates state for each SYN packet and the Bro's analyzer counts the number of connection requests or attempts to each host. If a count reaches a particular threshold value set by the administrator, Bro starts to ignore most of the packets and sample these packets to the destination (Sommer, 2003).

3.2. SNORT Intrusion Detection System

Snort was created by Martin Roesch in 1998, and is a single threaded open source NIDS developed by Sourcefire. It is a signature based IDS. Snort is the most widely deployed IDS technology worldwide. It can be used as a packet sensor, a packet logger or as a full network intrusion detection system. It can do protocol analysis, content searching or matching and can detect a variety of attacks such as buffer overflows, stealth port scans, etc.

Components of Snort: Snort is divided into several logical components such as packet decoder, preprocessor, etc. These components work at the same time to successfully detect attacks (Pritika, 2012). It consists of the following major component.

- **Packet Decoder**: It takes packets from available and chosen network interfaces and sends it to the preprocessor.
- **Preprocessors:** Preprocessor does apply some useful techniques for classifying the data before the packet is being used by an attacker.
- **Detection Engine:** Detection engine uses rules of Snort for detecting any malicious activity that exists in a packet.
- **Logging and Alerting System:** Logging and Alerting System is used to log different activities and generate an alert if malicious activities are found.
- **Output Modules:** Output modules compute alerts and logs and produce final output.

Snort can be used to detect most vulnerable attacks of the application layer such as SQL injection attack and cross-site scripting attack. Snort has a default rule set that contains a large number of signatures for detecting these intrusions (Search, 2012). Snort can be combined with other software, such as SnortSnarf, OSSIM, sguil, Snorby, Razorback and Basic Analysis and Security Engine (BASE) to provide results of detected intrusions graphically (Snort ids, 2011).

4. Review of Current Research in Intrusion Detection Systems

A lot of research is being performed in the field of network security. Here we are presenting some of the latest research performed in intrusion detection. We present this based on the approach used in developing IDS, types of attacks these IDSs can address and according to agent based technology.

4.1. Research Based on Approach

Misuse Based: In the work done by (Zhang 2009), they proposed an IDS architecture which is based on the content of communication to the database. A sniffer (packet analyzer) is implemented to capture database communication packets and from these captured packets SQL commands are extracted and then matched with known attacks patterns. Sniffer is used to log network packets and analyze them. Sniffer sends only sequences of doubtful packets to the database and thus reduces the amount of data passing through the database. The main drawback of this model is that it only detects database intrusions and hence cannot be deployed as general IDS for monitoring entire network traffic.

Anomaly based: In (Abraham, 2007), a distributive intrusion detection system is implemented that uses co-operative agents which are distributed across the network. This paper evaluates hybrid fuzzy classifiers like neuro-fuzzy and genetic-fuzzy to accurately find intrusion in the network. By using hybrid classifiers the IDS becomes robust and flexible. The main drawback of this system is that it uses excessive system resources.

Hybrid IDS: CONDOR (David, 2012) is a hybrid IDS that combines misuse as well as anomaly based. It uses packet sniffer to do sequence capturing and also uses the sliding window algorithm to get the sequence patterns. In addition, it also generates new signatures from the collected sequences and thus reduces an administrator intervention. The disadvantage of this system is that it cannot stop or detect polymorphic worm propagation and other zero day attacks**.**

4.2. Research Based on Attacks

(Farhan, 2010) uses two methods for anomaly detection: Conformal Predictor k-nearest neighbor and Distance based Outlier Detection (CPDOD) algorithm to detect anomalies in Mobile Adhoc Networks (MANETs). The Conformal Prediction for k-nearest neighbor (CP-KNN) algorithm does the matching of patterns obtained from the network and other examples in the class using K-nearest neighbor. Local Distance Outlier Factor (LDOF) is used to determine whether a data point is an outlier with respect to all the clusters formed in the network by finding the relative location of a point to its neighbors. The system issues an alarm against three common attacks: resource consumption attack, dropping routing traffic Attack and black hole attack.

In (Cristian, 2013), the authors proposed a multi-agent architecture aimed to detect SQL injection attacks. This architecture is based on distributed and hierarchical strategy and the functionalities of the system are structured on layers. Agents in each layer are assigned specific tasks such as data gathering, classification and visualization of data. Case-based Reasoning (CBR) is used by the classifier agent and by using case-based reasoning the agent has the benefit to learn from experiences. Visualization agent incorporated neural network and Support vector Machine (SVM) to support the graphical view of intrusion detected in the network.

4.3. Research Based on Multiagent Technology

In (Alvaro, 2013) and (Ganpathy, 2012), the authors proposed an intelligent agent based IDS for MANETS using a combination of classification methods namely, outlier detection, and multi-class SVM. In (Alvaro, 2013) the system uses agent technology with neural networks and case-based reasoning so that the agents can learn from previous experiences and detect attacks in the network. Temporal restrictions are also imposed on agents for doing real time processing in the network. The main advantage of this system is that average execution time is reduced by using CBR with temporal restrictions for detecting attacks and assigning tasks to various agents in the network.

In (Ganpathy, 2012) two new algorithms: an Intelligent Agent Weighted Distance Outlier Detection algorithm and an Intelligent Agent-based Enhanced Multi-class Support Vector Machine algorithm are proposed for detecting the intruders in a distributed database environment. The advantage of this method is that it reduces the false positive rates.

The intrusion detection systems which are based on multi-agent technology mainly used mobile agents in their systems. Use of mobile agents in intrusion detection has several benefits. They minimize the delay of response as they directly dispatch useful information to the destination hosts. Mobile agents can also handle the traffic in the network efficiently. Other benefits of mobile agents are they are platform independent, robust and the systems which use mobile agents are highly scalable because mobile agents can be easily cloned, dispatched and distribute themselves to the new machines whenever essential.

5. Discussion

There are number of IDSs available. Some of them are open source while others are not. Snort and Bro are the two open source IDSs which can be downloaded from their websites. While choosing the IDS the user must keep various parameters in mind such as functionality, and ease of use. Bro IDS is a system for experimentation but one should use that if he has the knowledge of UNIX. Bro IDS has the ability to rum on high speed links. On the other hand Snort focuses on simplicity and performance.

Snort is easy to deploy and also provides graphical user interface for its output. It becomes the de facto standard and today there are millions of users of Snort because of its easy-to-use nature. Dragon, NetRanger and Cisco Secure ID are commercial IDS which are very much used in organizations. They provide high level of security and well documented reports for management decisions.

6. Conclusions and Scope for Future Work

In this paper two Intrusion detection systems BRO and SNORT and work performed by different researches are briefly presented. This review provides a framework for having a general idea about the intrusion detection systems and also gives the current research work which is taking place in this field. There are numerous IDSs have been built for the security of computer systems from threats caused by the attackers. All these systems are capable of detecting attacks in the network and issue alarms when found malicious activities. But still there is a need to do more work in this field as attacks are increasing day by day; moreover, hackers find new ways of exploiting the network resources by using various evasion techniques. There is a need for a powerful intrusion detection system which can detect all possible attacks as early as possible. Multiagent technology is the upcoming technology in this field as it is more scalable, robust and can also reduce network traffic. The future work will be to develop agent based IDS for detecting attacks in the network.

References

[1] Abraham, A., Jain, R., Thomas, J. & Han, S. Y. (2007). D-SCIDS: Distributed soft computing intrusion detection system. *Journal of Network and Computer Applications*, January, 30(1), 81-98. Elsevier.

[2] Herro, A., Navarro, M., Corchado, A. & Julian, V. (2013). RT-MOVI-CAB-IDS: Addressing real-time intrusion detection. *Journal of Future Generation Computer System*, 29(1), 250-261. Elsevier.

[3] Pinzón, C. I., Paz, J. F. D., Herrero, A., Corchado, E., Bajo, J. & Corchado, J. M. (2013). 'idMAS-SQL: Intrusion detection based on MAS to detect and block SQL injection through data mining. *Journal of Information Science*, May, 231, 15-31. Elsevier.

[4] Day, D. J., Flores, D. A. & Lallie, H. S. (2012). *CONDOR: A Hybrid IDS to Offer Improved Intrusion Detection*. IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications.

[5] Abdel-Fattah, F., Dahalin, M. Z., Jusoh, S. (2010). Dynamic intrusion detection method for mobile Ad-Hoc network using CP DOD algorithm. *IJCA Special Issue on Mobile Ad-hoc Networks MANETs*.

[6] Ganpathy, S., Yogesh, P. & Kannan, A. (2012). Intelligent agent-based intrusion detection system using enhanced multiclass SVM. *Journal of Computational Intelligence and Neuroscience*.

[7] Erickson, J. (2008). *Hacking: The Art of Exploitation* (2nd ed.). Networking Section TCP/IP Hijacking.

[8] Moya, M. A. C. (2008). *Analysis and Evaluation of the Snort and Bro Network Intrusion Detection Systems*.

[9] Oryspayuli, O. D. (2006). *What Intrusion Detection Approaches Work Well if only TCP/IP Packet Header Information is Available*? University of Twente, Enschede- The Netherlands.

[10] Mehra, P. (2012). 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*, August, 1(6), 383-386.

[11] Rehman, R. (2003). *Intrusion Detection Systems with Snort Advanced IDS Techniques Using Snort, Apache, MySQL, PHP, and ACID*. Prentice Hall.

[12] Trost, R. (2009). *Practical Intrusion Analysis: Prevention and Detection for the Twenty-First Century*. Addison Wesley Professional.

[13] Renjit, J. A. & Shumuganathan, K. L. (2011). Multi-agent-based anomaly intrusion detection. *Information Security Journal: A Global Perspective*, 20 (4-5), 185-193.

[14] Search Security. (2012). Retrieved from http://searchsecurity.techtarget.com/answer/Can-Snort-stop-application-layer-attacks (accessed on November 19, 2013).

[15] Slideshare. (2012). Retrieved from http://www.slideshare.net/sukhsandhu/security-problems-in-tcp-ip (accessed on October, 2013).

[16] Snort ids. (2011). Retrieved from www.wikipedia.org/wiki/snort (software) (accessed on October, 2013).

[17] Sommer, R. (2003). *Bro: An Open Source Network Intrusion Detection System*.

[18] Seeberg, V. E. (2005). Bro-An IDS. Retrieved from http://www.infosikring.no/writings.html. (accessed on October 2013).

[19] Eddy, W. M. (2006). Defenses against TCP SYN flooding attacks. *The Internet Protocol Journal*, 9(4). Retrieved from http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_9-4/syn_flooding_attacks.html (accessed on December, 2013).

[20] Zhang, Y., Ye, X., Xie, F. & Peng, Y. (2009). *A Practical Database Intrusion Detection System Framework*. In IEEE Ninth International Conference on Computer and Information Technology, Xiamen, China.