ISSN 2320-9984

Volume 2 Issue 1 | March 2014 ISSN: 2320-9984 (Online)

International Journal of Modern Engineering & Management Research

Website: www.ijmemr.org

Intrusion Detection System Using Double Filtering And Session Layer Approaches

Hemendra Singh Yadav
Research Scholar
Department of Computer Science,
RKDF Institute of Science & Technology,
Bhopal, (M.P.) [INDIA]
Email : yadavhemendra@gmail.com

Prof. Niresh Sharma
Research Guide
Department of Computer Science,
RKDF Institute of Science & Technology,
Bhopal, (M.P.) [INDIA]

Abstract*—*Developing intrusion detection system with data mining is current research area now a day. Several paper and thesis are being prepared over this topic. This paper describes work carried out to prepare intrusion detection system using double filtering not only at TCP layer but also session layer. Because session layer always play a crucial role in the area of distributed networking. The key of data mining of intrusion detection lies in how to effectively distinguish normal behaviours and abnormal behaviours from plenty of initial data attributes and how to automatically and effectively generate intrusion rules after collecting the initial data of network.

1. INTRODUCTION

Data mining is the process of analyzing data from different perspectives and summarizing it into useful information that can be used to increase revenue. Data mining software is one of a number of analytical tools for analyzing data. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases.

Intrusion detection technology is immature and should not be considered as a complete defense, we believe, it can play a significant role in overall security architecture. If an organization chooses to deploy an IDS, a range of commercial and public domain products are available that offer varying deployment costs and potential to be effective. Because any deployment will incur ongoing operation and maintenance costs, the organization should consider the full IDS life cycle before making its choice. When AN IDS is properly deployed, it can provide warnings indicating that a system is under attack, even if the system is not vulnerable to the specific attack. These warnings can help users alter their installation's defensive posture to increase resistance to attack. In addition, IDS can serve to confirm secure configuration and operation of other security mechanisms such as firewalls. After describing the role IDS might play in an organization, we survey the most commonly used intrusion detection techniques and discuss representative systems from the commercial, public, and research areas.

2. INTRUSION DETECTION SYSTEM

2.1 Intrusions and Intrusion Detection

Intrusion detection has been an active field of research for about two decades, starting in 1980 with the publication of John Anderson's Computer Security Threat Monitoring and Surveillance, which was one of the earliest papers in the field. Dorothy

Denning's seminal paper, "An Intrusion Detection Model," published in 1987, provided a methodological framework that inspired many researchers and laid the groundwork for commercial products such as those we discuss in this thesis. Still, despite substantial research and commercial investments, IDS technology is immature and its effectiveness is limited. Within its limitations, it is useful as one portion of a defensive posture, but should not be relied upon as a sole means of protection. Many recent media reports point to the need for comprehensive protection of which IDS is a crucial part.

In the 1980s, most intruders were experts, with high levels of expertise and individually developed methods for breaking into systems. They rarely used automated tools and exploit scripts. Today, anyone can attack Internet sites using readily available intrusion tools and exploit scripts that capitalize on widely known vulnerabilities. Today, damaging intrusions can occur in a matter of seconds. Intruders hide their presence by installing modified versions of system monitoring and administration commands and by erasing their tracks in audit and log files.

2.2 Intrusion Detection System Diagram

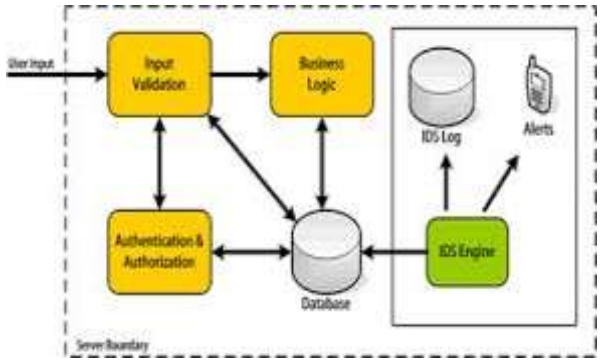

Figure 1: The block diagram of IDS

2.3 Intrusion Detection Approaches

We can view ID as an instance of the general signal-detection problem. In this case, we view intrusion manifestations as the signal to be detected and consider manifestations of "normal" operations to be noise.

Each approach has strengths and weaknesses. Both suffer from the difficulty of characterizing the distributions. For signature-based IDS to detect attacks, it must possess an attack description that can be matched to sensed attack manifestations. This can be as simple as a specific pattern that matches a portion of a network packet or as complex as a state machine or neural network description that maps multiple sensor outputs to abstract attack representations. Anomaly-based detectors equate "unusual" or "abnormal" with intrusions.

The primary strength of anomaly detection is its ability to recognize novel attacks. Its drawbacks include the necessity of training the system on noise with the attendant difficulties of tracking natural changes in the noise distribution. Changes can cause false alarms, while intrusive activities that appear to be normal can cause missed detections. Anomaly-based systems have difficulty classifying or naming attacks. We can also classify IDSs based on the phenomenology that they sense.

Network-based systems look at packets on a network segment, typically one serving an enterprise or a major portion of one. While network-based systems can simultaneously monitor numerous hosts, they can suffer from performance problems, especially with increasing network speeds. Many network-based systems make simplifying assumptions about such network pathologies as packet fragmentation and can suffer from resource exhaustion problems when they must maintain attack-state information for many attacked hosts over a long period of time. In spite of these deficiencies, they are popular because they are easy to deploy and manage as standalone components and they have little or no impact on the protected system's performance. Host-based systems operate on the protected host, inspecting audit or log data to detect intrusive activity. A variety of log and audit functions can serve to drive ID algorithms; these can be supplemented by

sensors that monitor the interaction of applications with the host operating system.

Host-based systems can monitor specific applications in ways that would be difficult or impossible in a network-based system. They can also detect intrusive activities that do not create externally observable behavior. Because they consume resources on the protected host, they can affect performance substantially. Successful intrusions that gain high levels of privilege might be able to disable host-based IDSs and remove traces of their operation. Intrusions that install UNIX root kits are examples.

2.4 Organizational Issues

Installing and effectively using IDSs on networks and hosts requires a broad understanding of computer security. Information technology infrastructures are becoming so complex that no one person can understand them, let alone administer them in a way that is operationally secure. An organization must fully appreciate the commitment required before deploying IDS. Otherwise, the project could well waste time, money, and staff resources in the IDS life cycle's initial phases. Although these issues are discussed in detail elsewhere, we cover them briefly here to illustrate the problem's scope.

2.4.1 Preparation

Before an organization invests in security technologies, it must understand which of its assets require protection and determine the real and perceived threats against those assets. We can characterize threats by the likely type of attack and attacker capabilities (that is, resources and goals) and the organization's tolerance for loss of, damage to, or disclosure of protected assets.

Attacker motives can be arbitrary (curiosity or vandalism) or targeted to meet a specific objective such as revenge or gaining a competitive advantage. Motives can make some forms of attack more likely than others. Gaining a competitive advantage might require compromising specific information such as a marketing plan. Each form of attack requires diverse detection strategies. For example, information retrieval is likely to occur during a stealthy attack, while information corruption might require speed. Determining whether the potential attacker is inside or outside the organization's infrastructure affects the type and placement of IDS. Often the most significant obstacle to an information security improvement initiatives lack of management support. Managers have any goals to meet and must often make tradeoffs. Security only becomes important when it impinges on the organization's high-priority interests and reputation. Deploying and operating IDS requires significant management support at the level of the corporate chief information officer and information security manager. Without this support, this technology's successful operation and use will be short-lived, sustained only by the interest of those internal champions who believe in its benefit.

2.4.2 Defense in Depth

ID is only one aspect of a layered defensive posture or "defense in depth". Defense in depth begins with the establishment of appropriate and effective security policies. Effective policies help ensure that threats to critical assets are understood, managers and users are adequately trained, and actions to be taken when an intrusion is identified are defined. A good security policy puts ID in its proper perspective and context. Whenever possible, the policy should reflect the mission of the organization that promulgates it. Therefore, it should codify the rules governing enterprise operations as they are reflected in its information infrastructure and should explicitly exclude activities or operations not needed to support the enterprise's mission. A mission-oriented security policy can aid in configuring both firewalls and IDSs.

Establishing layered security architecture is advantageous whether IDS is deployed or not. In addition to formulating a security policy, the essential steps consist of implementing user authentication and access

controls, eliminating unnecessary services, applying patches to eliminate known vulnerabilities, deploying firewalls, using file integrity checking tools such as Tripwire, and so forth. Because most real-time commercial IDSs base their detection approach on known attempts to exploit known vulnerabilities, an administrator's time is often better spent minimizing vulnerability through the application of patches or other security measures.

In addition to helping to validate the inner firewall's rules, it also protects the inside should the Web server be compromised and used as a base to attack the inside. If we assume that the protected enterprise is mission-oriented and only runs a limited set of applications and protocols, we can configure the inner sensor to recognize as intrusive any unexpected protocols. Host-based sensors on each workstation or server can look for both unexpected applications and abnormal behaviour on the part of supported applications and the host operating system. When we combine the use of multiple firewalls and sensors configured to support a mission-specific security policy with a proactive vulnerability remediation policy, the removal of unneeded services, and the regular and careful use of integrity checking tools, the intruder's task becomes much more difficult.

2.5 Intrusion Detection Technology

Commercial ID technology is immature and dynamic to the point of instability. Both commercial and research products evolve rapidly. One consequence of this rapid change is that product lists, surveys, and re-views are quickly outdated. The "Technology" sidebar describes a sample of commercial, research, and public domain tools. Relatively little has been done in the area of evaluating IDS systems, but we present the results of several attempts in this area. Both anecdotal evidence and the results from the few completed evaluations indicate that current IDSs are not as effective as could be desired. Because no comprehensive evaluation of commercial products has been performed. In 1999, IBM Zurich tested two commercial systems, Real-Secure 3.0.x and Net-Ranger 2.1.2, using exploit scripts and tools available in their vulnerability database that were compatible with their test environment and for which the IDS claimed coverage in its documentation. Real-Secure detected 30 of 42 attacks, while Net-Ranger detected 18 of 32. Deployed in an operational setting, Real- Secure issued some 8,000 alarms in a month, over half of which were due to a weekly scan of the network performed for maintenance purposes. Both systems had fairly high false-alarm rates, but issued false alarms for different classes of activity. The most comprehensive evaluations of IDS systems reported to date were the 1984 and 1999, 13 offline evaluations performed by MIT's Lincoln Laboratory. The systems evaluated are the results of research funded by DARPA. In these evaluations, investigators took sensor data in the form of sniffed network traffic, Solaris BSM audit data, Windows NT audit data (added in 1999), and file system snapshots and tried to identify the intrusions that had been carried out against a test network during the data-collection period.

3. PREVIOUS RELATED WORK

Some interest has already been expressed in data mining as an aid to intrusion detection. There are, however, differences in the approaches taken in the use of mining techniques. The research papers presented below illustrate these differences and provide an insight into the potentially diverse application of data mining for intrusion detection purposes.

- Columbia University : A pioneering research activity that emphasizes data mining as the leading paradigm in intrusion detection was MADAM ID, part of the larger JAM Project at the Computer Science Department of Columbia University, lead by Salvatore Stolfo. The original idea behind JAM (Java Agents for Meta-Learning) was to use data mining techniques to correlate knowledge derived from separate, heterogeneous

data sets into a rule-set capable of providing a general description of an environment comprising these sets.

This work lead to the further use of data mining techniques to build better models for intrusion detection by analyzing audit data using associations and frequent episodes, and utilizing the resulting rules when constructing ID classifiers. The results from this effort are collectively labeled as MADAM ID (Mining Audit Data for Automated Models for Intrusion Detection). The project has received a number of distinctions from peers, such as achieving high scores at the DARPA 1998 Offline Intrusion Detection Evaluation and being awarded Best Paper in Applied Research at KDD 99. As an extension to the above achievements, Columbia University is currently coordinating a project to create a complete intrusion detection system operating real-time. The project, named Project IDS.

- Iowa State University: A different approach is taken by Iowa State University in their design of an intrusion detection system. Data mining techniques are incorporated into agent based architecture at high level in order to correlate information collected by low-level agents. The low-level agents are designed to propagate information upwards as well as sharing information on their designated level. The expected benefits of this design are the potential to recognize attacks spread over both space and time, localize their origin and co-ordinate a suitable response. The system also illustrates a successful combination of agent technologies with data mining. Data mining hence may become a prominent component in large-scale IDS warranting continued examination of other, potentially useful techniques.

- Intrusion scenarios detection based on Data Mining: The disadvantage of current scenario detection method is that they depend on human knowledge to build attack scenario model, its labour consuming. This paper proposes a new method of scenario detection based on data mining.

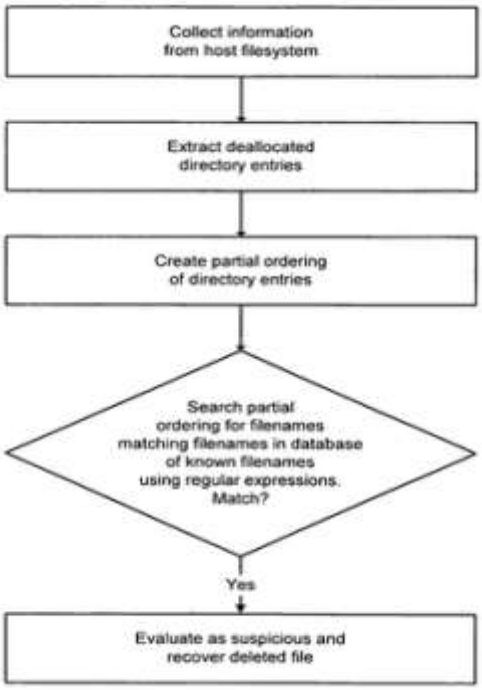

Figure 2: Flowchart of Functioning IDS

4. PROPOSED WORK

4.1 Research Ideas and Critical Analysis

Before getting into the mechanisms of IDS, it is necessary to understand the commonly used network protocols. TCP is a transport layer protocol and is situated above the IP layer in the *(Open Systems Interconnection) OSI* model. The transport layer is responsible for establishing sessions, data transfer and tearing down virtual connections. The existence of the port numbers with the Data component makes the TCP header the most important PDU (Protocol Data Unit) in network security. It is also worth noting that even though the PDU itself will appear to behave normally, the data in it can very well be malicious and anomaly detection is very effective in detecting this. An example of a Transport layer PDU or TPDU.

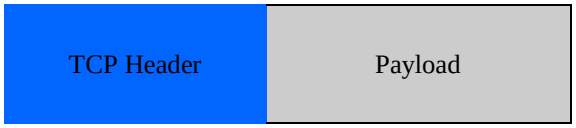

Figure 3: Transport Layer PDU

Next is the IP layer in the OSI Model. The IP header is considered to be the next most important PDU (and is called the packet) having source and destination IP addresses with data.

Figure 4: Packet

An IDS using unsupervised learning can effectively be used to analyse header and data components of packets and classify them as normal or malicious even though there is no knowledge of the data. The next section will describe the steps that were taken for creating a self organised map and how it was used to group together similar data given a multidimensional data set as input.

4.2 Proposed Working Model

4.2.1 Packet filtering

We should know what packet filtering is. Packet filtering is a process of allowing or blocking packets at OSI: physical, data-link, network, transport, session, presentation or application layer.

4.2.2 Capabilities of a Packet Filter

A packet filter has to have the following capabilities:

I. Examination of each packet data and headers. Each packet is examined when it comes to the packet filter. This is done with the help of filtering rules defined in the next point.

II. Set of rules which define what to do with the packet. These rules define what a packet filter should look for when it receives a packet. It usually looks for the information we've already talked about, like source IP address, destination IP address, source port number, destination port number, etc.

III. What actions are taken based on the result of examination. There are numerous actions which can be used when a packet filter receives a packet and has filtering rules defined. Based on defined filtering rules, a packet filter can do the following:

a. Accept only packets that are certainly safe based on a set of rules. Drop all other packets.

b. Drop only packets that are certainly unsafe based on a set of rules. Accept all other packets.

c. If a packet is received for which there is no filtering rule defined, ask a user what to do with it.

d. Block a user coming from a defined source IP address, because too many packets were received in too short of a time window.

e. Almost any action can be applied against a packet or a set of packets – the sky is the limit. If we want to send a HTTP response, which includes "Hello, my name is Santa Claus" to every HTTP request coming from IP xxx.xxx.xxx.xxx, we could define a rule that could do that.

The following categories of attacks:

- Attack type
- Number of network connections involved in the attack
- Automation level

4.3 Double filtering

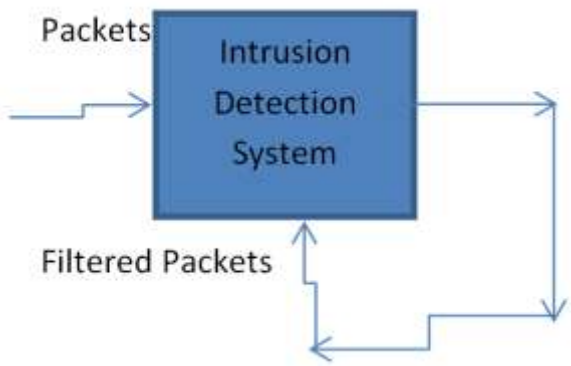

Figure 5: Double Filtering

4.3.1 Re filtering of packets

Figure 5 showing how double filtering will take place. Output of IDS is again input to IDS. This Method will improve results surely.

4.3.2 Advantage of double filtering

1. It will reduce chances of presence of suspicious packet.
2. Some infected packets are programmed to behave as normal packet during intrusion detection after getting passed it comes to its original form and functioning. This type of packet can be caught by double filtering.

4.4 Session layer filtering

1. First check average duration time required for process.
2. Any distributed object that is thrown is the same, required or called by function of other server.
3. Before commitment of transaction we will insert timer during which we check the validity of both server and client involved in session.

4.4.1 Algorithm for session layer filtering

1. Calculate average duration time

If average duration time is T[ad] then

T[ad] =last 5 transaction time/5

2. If current transaction time is T[cd] then

If during transaction T[cd] <= T[ad]

{Prompt transaction go to step 3}

Else

{Abort transaction}

3. Check thrown object content (data, code) to IDS before final processing.

4. If object data is infected/suspicious.

{Abort transaction}

Else

{Continue}

5. Commit transaction

5. CONCLUSION

In this paper we have developed a session layer algorithm for packet filtering. Packet filtering at session layer reduces deep penetration because data link layer and network layer filtering catch packet at last moment then it is very difficult to recover.

Double packet filtering optimize results upto 98%. Some drawback of this algorithm is slow processing due to double filtering. Second drawback is session layer algorithm implementation is very complex to execute.

REFERENCES

[1] Changxin Song, Ke Ma, 2009. "Design of Intrusion Detection System Based on Data mining Algorithm," 2009 International Conference on Signal Processing Systems.

[2] Ya-Li Ding, Lei Li, Hong-Qi Luo 2009. "A Novel Signature Searching For Intrusion Detection System Using Data Mining," 2009 Proceedings of the eighth International Conference On Machine Learning And Cybernetics, Baoding,12-15

July2009.pp.10-16.

[3] Ye-Changguo, WEI Nianzhong, Wang Tailei, Zhang Qin, Zhu Xiaorong, 2009, "The Research On The Application Of Association Rules Mining Algorithm In Network Intrusion Detection", 2009. First International Workshop on Education Technology And Computer Sciencepp.453-501.

[4] Yu-Xin Ding, Hai-Sen Wang, Qing-WEI liu, 2008."Intrusion Scenarios Detection Based On Data Mining." 2008, Proceedings Of The Seventh International Conference On Machine Learning And Cybernetics, Kunming,vol.51,No.3,pp-632-654,Feb2008.

[5] Nan Zhang, Wei Zhao, 2007."PRIVACY Preserving Data Mining System," 2007, published by the IEEE computer society, vol4, no.2, 2007.

[6] Hu Zheng Bing, Shirochin V.P. 2005. " Data Mining Approaches For Signatures Search In Network Intrusion Detection," 2005 IEEE Workshop On Intelligent Data Acquisition and Advance Computing Systems, Bulgaria,2005.pp.153-157.

[7] Yu-Fang Zhang, Zhong-YANG Xiong, Xiu-Qiong Wang 2005. "Distributed Intrusion Detection Based On Clustering."2005 Proceedings Of The Fourth International Conference On Machine Learning and Cybernetices, Guangzhou.

[8] D.N.Wu, and S. Jajodia 2001. "Detecting Novel Network Intrusions Using Bayes Estimators", Proceedings Of the First SIAM Int. Conference on Data Mining, (SDM 2001), Chicago, I,Vol 5 No.3.

[9] Skorupka, C., J. Tivel, L. Talbot, D. Debarr, W. Hill, E. Bloedorn, and A. Christiansen 2001. "Surf the Flood: Reducing High-Volume Intrusion Detection Data by Automated Record Aggregation," Proceedings of the SANS 2001 Technical Conference, Baltimore, MD.

[10] Bloedorn, E., L. Talbot, C. Skorupka, A. Christiansen, W. Hill, and J. Tivel 2001. "Data Mining applied to Intrusion Detection: MITRE Experiences," submitted to the 2001 IEEE International Conference on Data Mining.

[11] Breunig, M. M., H. P. Kriegel, R. T. Ng, and J. Sander 2001. "LOF: Identifying Density-Based Local Outliers", Proceedings of the ACM Sigmod 2000 Intl. Conference On Management of Data, Dallas, TX.

[12] Clifton, C., and G. Gengo 2000. "Developing Custom Intrusion Detection Filters Using Data Mining", 2000 Military Communications International, Los Angeles, California, October 22-25.

[13] Domingos, P., and G. Hulten 2000. "Mining High Speed Data Streams", in Proceedings of the Sixth ACM SIGKDD Conference on Knowledge Discovery and Data Mining, p. 71-80.

[14] Ramaswarny, S., R. Rastogi, and K. Shim, 2000. "Efficient Algorithms for Mining Outliers from Large Data Sets", Proceedings of the ACM Sigmod 2000 Int. Conference on Management of Data, Dallas, TX.

[15] Lee, W., & Stolfo, S.J. (2000) A framework for constructing features and models for intrusion detection systems. ACM Transactions on Information and System Security, 3 (4) (pp. 227-261).