Applying Data Mining Techniques to Intrusion Detection

Jonathon Ng, Deepti Joshi, Shankar M. Banik
Department of Mathematics and Computer Science
The Citadel
Charleston, South Carolina, USA
{hng, djoshi, baniks1}@citadel.edu

***Abstract*—In our current society, the threat of cyber intrusion is increasingly high and harmful. With the rise of usage in computers, criminal activity has also shifted from physical intrusion into cyber intrusion. Intrusion detection systems provide the ability to identify security breaches in a system. A security breach will be any action the owner of the system deems unauthorized. Current methods used for these systems include using anomaly detection or a signature database. In this research we use both anomaly detection and a signature database using data mining techniques. Our solution provides a tool that would run data mining tools against a log file to detect patterns that may be considered an unauthorized activity. The tool gains additional patterns as time goes by and grows more effective. It allowed us to detect brute force password cracking and Denial-of-Service (DoS) attacks on a system in the Ubuntu platform.**

Keywords-Cybersecurity; Intrusion Detection; Data Mining

I. INTRODUCTION

People want to keep their possessions secure. A security system in a house is common place in current times. Now that technology has advanced we have adopted security systems on our computers as well. One such security system is an Intrusion Detection System. Intrusion Detection Systems (IDS) provide the ability to identify security breaches in a system. A security breach will be any action the owner of the system deems unauthorized. Current methods used for these systems include using anomaly detection or a signature database. Signature databases hold "definitions" of an attack. A definition describes each attack that may occur in an environment. An anomaly is any unusual event that occurs in an environment. This method is used to detect attacks that have not been defined yet. The rule "anything that is broken in the house is considered an attack" will be a general anomaly that will be able to detect the definition previously provided.

In this research we use both anomaly detection and a signature database using data mining techniques. Our solution provides a tool that would run data mining tools against a log file to detect patterns that may be considered an unauthorized activity. After the pattern is confirmed by the owner of the system as an attack, the attack pattern will be stored in a signature database. The tool gains additional patterns as time goes by and grows more effective. Our tool currently uses the concept of clustering log entries that repeat multiple times and detects brute force password cracking and DoS attacks on a system in the Ubuntu platform.

The paper is organized as follows. Section II provides a brief description of the most popular attacks that are detected by IDS. Section III provides a literature review on data mining in IDS. In Section IV, we present the design of our data mining tool that we have developed for intrusion detection. We present our results in section V. Section VI concludes our paper.

II. BACKGROUND

In this section we present a list of the most popular attacks that an intrusion detection system may need to detect. The types of attacks presented here, and studied for this part of the work are DoS Attacks. These attacks are used to make system unavailable to legitimate users. A few DoS attacks that may occur are:

Internet Control Message Protocol (ICMP) Flood – ICMP Flood occurs when a user sends a large amount of pings to an address either from multiple addresses or from the same address to slow down a system. The attack can be seen in a log by viewing multiple pings from a machine or continuous pings from many machines in a short duration. One way to prevent this is by disabling ICMP.

SYN flood – SYN flood uses the TCP connection to send a SYN flag to a host and never sends an ACK response after the host sends a SYN/ACK. The attack can be seen in a log by viewing multiple SYN requests from a machine or many machines in a short duration. When a machine performs the attack, a block can be placed on the IP or IPs performing the SYN Flood request.

Server connection limit – This attack is not necessarily an attack but is more of the limitation on a server. This occurs when the amount of connections to a server has exceeded the server's capabilities. Therefore, the administrator of the server should understand what kind of traffic to expect so they could purchase a server that could handle the amount of traffic it expects.

Failed Log In – This attack occurs whenever a person is trying to guess another person's password by inputting multiple password attempts that fail. This can be detected by a log that shows that a specific user account is being provided an incorrect password. This log can also be mistaken for a person who has forgotten their password. One way to prevent this attack is by creating a password guessing limit and requiring a request for a password change.

Structured Query Language (SQL) Injection – An SQL Injection attack attempts to exploit SQL databases that do not format input requests from a server. This can be detected in a

978-1-4799-8828-0/15 $31.00 © 2015 IEEE
DOI 10.1109/ITNG.2015.146

log file when an input shows an SQL statement instead of a normal input such as a name or a password. An example would be the phrase " <SQL CODE> '- -"which is used to comment out any code following the code being input and finishing what the malicious user believes is the SQL statement being used. One way to prevent this kind of attack is by verifying inputs before using them on the database.

III. Methodology

A. Project Goal

The research presented in this paper is part of our on-going work which is to investigate the role of data mining algorithms in an Intrusion Detection System. Most importantly we would like to find out if discovering patterns within a log file will provide patterns of intrusion attacks.

In this paper we test the ability of Data Mining using a clustering technique to discover DoS Attacks. Clustering refers to the grouping of similar data. This grouping allows users to see patterns of reoccurring activities or popular trends. The description of reoccurring activities is highly compatible with the description of a denial of service attack. We also investigate the use of Data Mining to incorporate both the signature and anomaly database scheme.

B. PreProcessing Log Files

The first step in designing and implementing our data mining tool for intrusion detection, was to analyze and parse the network log files. In order to do so, the following steps were implemented. 1) Extract the date and time, 2) Extract data until first colon, 3) Extract data within parenthesis, 4) Decode remaining information. The program adds a new column when a part of the line is split based on special characters, thus the header has the column information. The extracted data is added to the body of the file.

C. Data Mining Tool for Detecting Attacks

We categorize any action that may bring down a system or retrieve information as an attack whether it is honest or malicious. In the current version of our tool we have implemented a clustering algorithm which matches connections that appear multiple times. This enables us to detect possible password guessing or DoS attacks.

IV. Results

For this work, we decided to use log patterns within a Linux Operating system as our base for data collection. After creating a virtual machine of Ubuntu, we allowed the system to run for a certain amount of time with no attacks and stored the log file. Afterwards we performed simulated attacks of an Internet Control Message Protocol which disables a computer by sending large amounts of "pings". We also conducted brute force password attacks.

We formatted the normal log and attack log files (see Figure 1 for a sample) to run through our Data Mining tool. The normal log would be used to set definitions of any normal activities in a normal log database so that the pattern generator will ignore them as findings. Any patterns that do not appear on the normal logs but do appear on the attack logs will be sent to the user for analysis and if confirmed will be placed in the attack database.

The program is able to detect if a pattern exists in the normal log before being considered an attack. Then the program will detect if the attack is already in the database of attack patterns and will alert if there is, otherwise it will prompt the user to add to the database. Figure 2 shows an example of the attack pattern results discovered.

```
08-10-2013_09:53:41,networklogtest,sshd[412],Received signal 15,
terminating.,,,,,,,,,,,,,,,,,,,,
08-10-2013_09:53:41,networklogtest,sshd[738],Server listening on
0.0.0.0 port 22.,,,,,,,,,,,,,,,,,,,,
08-10-2013_09:53:41,networklogtest,sshd[738],Server listening on ,,
port 22.,,,,,,,,,,,,,,,,,,
08-10-2013_09:53:54,networklogtest,login[877],pam_unix, check pass,
user unknown,auth,,,,,,,,,,,,,,,,,
```

Figure 1: Sample Pre-Processed File

```
,networklogtest,sudo,pam_unix, session opened for user root by
admintest,,,,1000,,,,,,,,,session,,,,,, contains 13 repeats.
,networklogtest,sudo,pam_unix, session closed for user
root,,,,,,,,,,,,,session,,,,,, contains 13 repeats.
,networklogtest,sshd[2490],pam_unix, check pass, user
unknown,,,,,,,,,,,,,,,,,,,,auth contains 6 repeats.
,networklogtest,sshd[2490],Failed password for invalid user fakeid from
10.0.0.2 port 55980 ssh2,,,,,,,,,,,,,,,,,,,, contains 6 repeats.
,networklogtest,login[2493],pam_unix, check pass, user
unknown,auth,,,,,,,,,,,,,,,,, contains 5 repeats.
,networklogtest,login[2493], authentication
failure,pam_unix,,auth,admintest,0,0,/dev/tty1,,,,,,,,,,,,,, contains 5
repeats.
```

Figure 2: Attack Pattern Results

V. Conclusion

After running through our tool we were able to successfully detect the simulated attacks that we made. As can be seen in the UML model (Figure 3), we would like to add more Patterns to detect other attacks. We would also like to create a Real Time Intrusion Detection System which will actively detect intrusions while the machine is running.

References

[1] Antoniou, Stelios. "The PING of Death and Other DoS Network Attacks." Train-Signal. n.p. 14 May 2009. Web. 4 August 2013. http://www.trainsignal.com/blog/ping-of-death-and-dos-attacks

[2] Fergal, Glynn. "SQL Injection Tutorial: Learn About Injection Attacks, Vulnerabili-ties and How to Prevent SQL Injections." Veracode. n.p. n/a. Web. 4 August 2013. <http://www.veracode.com/security/sql-injection>

[3] Joao B.D. Cabrera, Lundy Lewis, Xinzhou Qin, Wenke Lee, Raman K. Mehra, Proac-tive Intrusion Detection - A Study on Temporal Data Mining, Applications of Data Min-ing in Computer Security. Barbara and S. Jajodia (eds), Kluwer Academic Publishers, May 2002.

[4] Wenke Lee, Applying Data Mining to Intrusion Detection: The Quest for Automation, Efficiency, and Credibility, in SIGKDD Explorations, 4(2), December 2002.

[5] Wenke Lee, Sal Stolfo, and Kui Mok, Algorithms for Mining System Audit Data, Da-ta Mining, Rough Sets, and Granular Computing, T. Y. Lin, Y. Y. Yao, and L. A. Zadeh (eds), Physica-Verlag, 2002.

[6] Wenke Lee, Sal Stolfo, Phil Chan, Eleazar Eskin, Wei Fan, Matt Miller, Shlomo Hershkop, and Junxin Zhang, Real Time Data Mining-based Intrusion Detection, in Proceedings of The 2001 DARPA Information Survivability Conference and Exposition (DISCEX II), Anaheim, CA, June 2001.