Journal of Data Mining and Knowledge Discovery
ISSN: 2229–6662 & ISSN: 2229–6670, Volume 3, Issue 2, 2012, pp.-83-87.
Available online at http://www. bioinfo. in/contents. php?id=42

APPROACH FOR INTRUSION DETECTION SYSTEM USING DATA MINING

PAREKH S.P., MADAN B.S. AND TUGNAYAT R.M.

Department of I.T., J.D.I.E.T., Yavatmal, MS, India.
Corresponding Author: Email- suchita13488@gmail.com; bhagyashreemadan@yahoo.in; tugnayatrm@rediffmail.com

Received: March 15, 2012; Accepted: April 12, 2012

Abstract- The number of hacking and intrusion incidents is increasing alarmingly each year as new technology rolls out. In this paper report, we designed Intrusion Detection System (IDS) that implements pre-defined algorithms for identifying the attacks over a network. In this paper we discuss the term Intrusion Detection System using Data Mining which is generally used with the networking applications where the hacking attempts are made by the hackers. The key ideas are to use data mining techniques to discover consistent and useful patterns of system features that describe program and user behavior and use the set of relevant system features to compute (inductively learned) classifiers that can recognize anomalies and known intrusions.
Here the HOP-COUNT-filtering (HCF) algorithm is used to identify whether the hacking attempt is made or not, and the Data Mining principles are used for identifying the normal IP Address and Hacked IP address. The Data Mining help to prevent the hacked IP packets and alarm is generated by the system and information is shown at the receiver side.
The packets in the network are captured online i.e., as they come on the interface of the network. The IDS is designed to provide the basic detection techniques so as to secure the systems present in the networks that are directly or indirectly connected to the internet.
Keywords- DDoS attacks,Data mining,IDS, IP spoofing, hop-count.

Citation: Parekh S.P., Madan B.S. and Tugnayat R.M. (2012) Approach For Intrusion Detection System Using Data Mining. Journal of Data Mining and Knowledge Discovery, ISSN: 2229–6662 & ISSN: 2229–6670, Volume 3, Issue 2, pp.-83-87.

Introduction

One of the main challenges in the security management of large-scale high-speed networks is the detection of suspicious anomalies in network traffic patterns due to Distributed Denial of Service (DDoS) attacks or worm propagation. A secure network must provide the following:

- Data confidentiality: Data that are being transferred through the network should be accessible only to those that have been properly authorized.
- Data integrity: Data should maintain their integrity from the moment they are transmitted to the moment they are actually received. No corruption or data loss is accepted either from random events or malicious activity.
- Data availability: The network should be resilient to Denial of Service attacks.

The first threat for a computer network system was realised in 1988 when 23-year old Robert Morris launched the first worm, which overid over 6000 PCs of the ARPANET network. On February 7th, 2000 the first DoS attacks of great volume where launched, targeting the computer systems of large companies like Yahoo!, eBay, Amazon, CNN, ZDnet and Dadet.
These threats and others that are likely to appear in the future have lead to the design and development of Intrusion Detection

Systems. An intrusion detection system (IDS) inspects all inbound and outbound network activity and identifies suspicious patterns that may indicate a network or system attack from someone attempting to break into or compromise a system

What is IDS?

Intrusion detection is often used as another wall to protect computer systems. Intrusion detection (ID) is defined as "The problem of identifying individuals who are using a computer system without authorization (i.e., 'crackers') and those who have legitimate access to the system but are abusing their privileges (i.e., the 'insider threat')".

The goal of an IDS is to detect malicious traffic. In order to accomplish this, the IDS monitors all incoming and outgoing traffic. There are several approaches on the implementetion of an IDS. Among those, two are the most popular

- **Anomaly detection**

This technique is based on the detection of traffic anomalies. The deviation of the monitored traffic from the normal profile is measured. Various different implementations of this technique have been proposed, based on the metrics used for measuring traffic profile deviation.

- **Misuse/Signature detection**

This technique looks for patterns and signatures of already known attacks in the network traffic. A constantly updated database is usually used to store the signatures of known attacks. The way this technique deals with intrusion detection resembles the way that anti-virus software operates.

Drawbacks of current IDS

Intrusion Detection Systems (IDS) have become a standard component in security infrastructures as they allow network administrators to detect policy violations. These policy violations range from external attackers trying to gain unauthorized access to insiders abusing their access.

Current IDS have a number of significant drawbacks-

- Current IDS are usually tuned to detect known service level network attacks. This leaves them vulnerable to original and novel malicious attacks.
- Data overload: Another aspect which does not relate directly to misuse detection but is extremely important is how much data an analyst can efficiently analyze. That amount of data he needs to look at seems to be growing rapidly. Depending on the intrusion detection tools employed by a company and its size there is the possibility for logs to reach millions of records per day.
- False positives: A common complaint is the amount of false positives an IDS will generate. A false positive occurs when normal attack is mistakenly classified as malicious and treated accordingly.
- False negatives: This is the case where an IDS does not generate an alert when an intrusion is actually taking place. (Classification of malicious traffic as normal)

Data mining can help improve intrusion detection by addressing each and every one of the above mentioned problems.

Data mining

Data mining (DM), also called Knowledge-Discovery and Data mining is, at its core, pattern finding. Data miners are experts at using specialized software to find regularities (and irregularities) in large data sets. Here are a few specific things that data mining might contribute to an intrusion detection project:

- Remove normal activity from alarm data to allow analysts to focus on real attacks
- Identify false alarm generators and "bad" sensor signatures
- Find anomalous activity that uncovers a real attack
- Identify long, ongoing patterns (different IP address, same activity)

To accomplish these tasks, data miners employ one or more of the following techniques:

- Data summarization with statistics, including finding outliers
- Visualization: presenting a graphical summary of the data [1]
- Clustering of the data into natural categories [2]
- Association rule discovery: defining normal activity and enabling the discovery of anomalies [3]
- Classification: predicting the category to which a particular record belongs [4]

Literature review

Computer Forensics, which views computer systems as scenes of a crime, is computer security technologies that analyze what attackers have done. Most of their applications focus on how to identify malicious network behaviors and the characteristics of attack packets, and the way to identify attack patterns based on their analyses. Abdullah et al. [5] used package dump tools, such as tcpdump and pcap, to collect and analyze network packets and to identify network attacks from different network states and packets' distribution.

Yu et al. [6] provided another example of integrating computer forensics with IDS. A knowledge-based system was deployed to collect forensic features from malicious network behaviors. This system performed excellently in improving the hit rate of intrusion alerts. Yin et al. [7] proposed an approach that built a Markov chain to describe users'normal operations. A state of the chain records the probability of entering the next state. However, this approach focuses on system calls generated instead of commands submitted.

Chau et al. [8] used a pattern extraction technique to identify particular crime data, such as segmenting and extracting a suspect from a picture on a security video.

Cabrera et al. [9] deployed sequential pattern mining to identify attack patterns that hackers frequently submit, and classified the *modus operandi* that suspects used in the commission of crimes into predefined crime types. These techniques and applications truly contribute to network security. However, they cannot easily authenticate remote-login users, and cannot detect specific types of intrusions, e.g., when an unauthorized user logs in to a system with a legal UID and password. Authentication based on the user's operation habits is what we propose.

The IDIS uses data mining and forensic techniques to respectively analyze and identify user operation characteristics, which as a kind of biological characteristics are essential in identifying a user.

This system can identify attack patterns that hackers often use as well. By long-term observation, user habits can be effectively identified.

Structure of proposed system

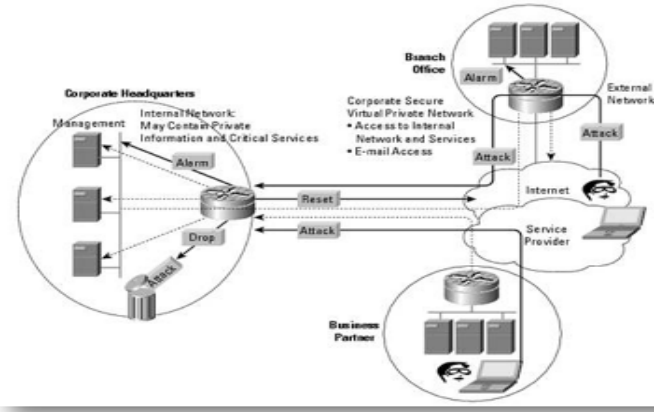

Fig. 1- Structure of Proposed System

IP spoofing

IP networks are vulnerable to source address spoofing. For example, a compromised Internet host can spoof IP packets by using a raw socket to fill arbitrary source IP addresses into packet heade rs.

IP spoofing is commonly associated with malicious network activities, such as Distributed Denial of Service (DDoS) attacks which block legitimate access by either exhausting victim servers or saturating stub networks access links to the Internet.

Proposed system

We propose a lightweight scheme that validates incoming IP packets at an Internet server without using any cryptographic methodology or router support. Our goal is not to achieve perfect authentication, but to screen out most bogus traffic with little collateral damage. The fundamental idea is to utilize inherent network information—that each packet carries and an attacker cannot easily forge—to distinguish spoofed packets from legitimate ones. The inherent network information we use here is the number of hops a packet takes to reach its destination: although an attacker can forge any field in the IP header, he cannot falsify the number of hops an IP packet takes to reach its destination, which is solely determined by the Internet routing infrastructure. The hop-count information is indirectly reflected in the TTL field of the IP header, since each intermediate router decrements the TTL value by one before forwarding a packet to the next hop. Based on hop-count, we propose a novel filtering technique, called *Hop-Count Filtering* (HCF), to weed out spoofed IP packets at the very beginning of network processing, thus effectively protecting victim servers' resources from abuse. The rationale behind HCF is that most randomly-spoofed IP packets, when arriving at victims, do not carry hop-count values that are consistent with the IP addresses being spoofed. As a receiver, an Internet server can infer the hop-count information and check for consistency of source IP addresses. HCF builds an accurate IP-to-hop-count (IP2HC) mapping table, while using a moderate amount of storage, by clustering address prefixes based on hop-count. To capture hop-count changes under dynamic network conditions, we also devise a safe update procedure for the IP2HC mapping table that prevents pollution by attackers. The same pollution-proof method is used for both initializing IP2HC mapping table and inserting additional IP addresses into the table.

Hop-count filtering(HCF)

The ability to filter spoofed IP packets near victim servers is essential to their own protection and prevention of becoming involuntary DoS reflectors. Although an attacker can forge any field in the IP header, he cannot falsify the number of hops an IP packet takes to reach its destination. More importantly, since the hop-count values are diverse, an attacker cannot *randomly* spoof IP addresses while maintaining consistent hop-counts. On the other hand, an Internet server can easily infer the hop-count information from the Time-to-Live (TTL) field of the IP header. Using a mapping between IP addresses and their hop-counts, the server can distinguish spoofed IP packets from legitimate ones. Based on this observation, we present a novel filtering technique, called *Hop-Count Filtering* (HCF)—which builds an accurate IP-to-hop-count (IP2HC) mapping table to detect and discard spoofed IP packets**.** And also use the final verifiers to verify all the packets is received correct hosts.

Logical implementation of hop-count

The idea is to count two factors before accepting the packet

Actual TTL – TTL computed after sending packet from source to destination

Final TTL – TTL supposed to come as stored in database

Actual TTL = {INITIAL TTL- (packet received-1)}

Packet received will be in the form of (message + number of routers)

Final TTL = {Initial TTL- (stored routers)}

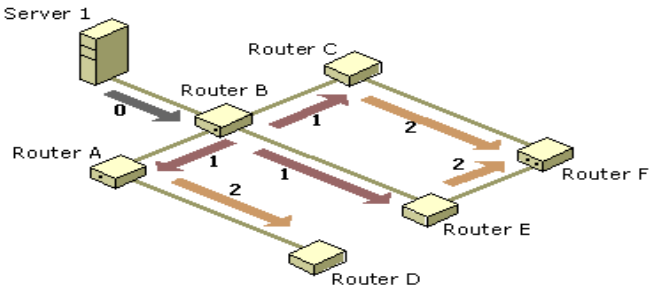

Fig. 2- Different routing paths

Example

Suppose number of routers stored =2

Initial TTL =34

Then: Without Hacking a packet transmits from Source-router1-router2-destination

Actual TTL = {34- (3-1)}

= 34- 2

= 32

Packet received will be in the form of (message + number of routers)

Final TTL = {34- (2)}

= 34- 2

= 32

Since both actual TTL and Final TTL are equal we conclude packet is not spoofed.

Then: With Hacking a packet transmits from Source-router1-hacker-router2-destination
Actual TTL = {34- (4-1)}
= 34- 3
= 31
Packet received will be in the form of (message + number of routers) (here hacker is enabled between routers)
Final TTL = {34- (2)}
= 34- 2
= 32
Since both actual TTL and Final TTL are not equal we conclude packet is spoofed.

Project development flow diagram

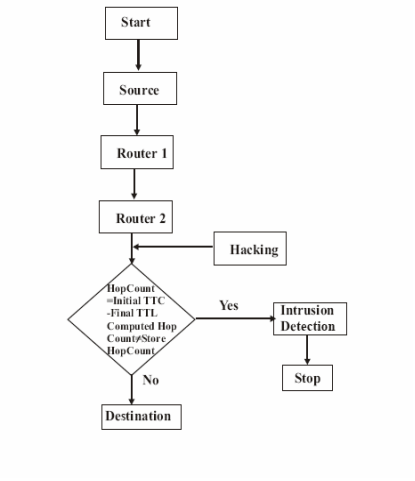

Fig. 3- Flow Chart Description of Proposed System

Advantages of using data mining
Data mining can help improve intrusion detection by adding a level of focus to anomaly detection. By identifying bounds for valid network activity, data mining will aid an analyst in his/her ability to distinguish attack activity from common everyday traffic on the network.

- **Variants-** Since anomaly detection is not based on predefined signatures the concern with variants in the code of an exploit are not as great since we are looking for abnormal activity versus a unique signature. An example might be a Remote Procedure Call (RPC) buffer overflow exploit whose code has been modified slightly to evade an IDS using signatures. With anomaly detection, the activity would be flagged since the destination machine has never seen an RPC connection attempt and the source IP was never seen connecting to the network.
- **False positives-** In regards to false positives there has been some work to determine if data mining can be used to identify recurring sequences of alarms in order to help identify valid network activity which can be filtered out.
- **False negatives-** …detecting attacks for which there are no known signatures. By attempting to establish patterns for normal activity and identifying that activity which lies outside identified bounds, attacks for which signatures have not been developed might be detected. An extremely simple example of how this would work would be to take a web server and develop a profile of the network activity seen to and from the system. Let us say the web server is locked down and only connections to ports 80 and 443 are ever seen to the server. Thus, whenever a connection to a port other than 80 or 443 is seen the IDS should identify that as an anomaly. While this example is quite simple this could be extended to profiling not only individual hosts, but entire networks, users, traffic based on days of the week or hours in a day, and the list goes on.
- **Data overload-** The area where data mining is sure to play a vital role is in the area of data reduction. With current data mining algorithms there exists the capability to identify or extract data which is most relevant and provide analysts with different "views" of the data to aid in their analysis.

Application of proposed system
Intrusion Detection System in data mining applications includes the following:

- Intrusion Detection System can be used in software applications which are related to Networks e.g. online data transfer system, social networking websites.
- Can be used in Centralized Information Access Prevention.
- Compare to other network security devices IDS work internally to the security of network.

Advantages of proposed system

1. Reliable
2. Economical
3. Easy to use and implement.
4. Prevent unwanted access to systems.
5. Will be used in future for hacking prevention.

Conclusion
This paper has presented a way of the data mining technique that have been proposed towards the enhancement of IDSs. We have shown the way in which data mining has been known to aid the process of Intrusion Detection and the way in which the technique have been applied and evaluated by researcher. Finally, we proposed a data mining approach that we feel can contribute significantly in the attempt to create better and more effective Intrusion Detection Systems.

References

[1] Rao A.A., Srinivas P., Chakravarthy B., Marx K., and Kiran P. (2006) *A Java Based Network Intrusion Detection System (IDS)Session ENG,* 206-118.
[2] Lappas T. and Pelechrinis K. *Data Mining Techniques for (Network) Intrusion Detection Systems.*
[3] Barbara D., Wu N. and Jajodia S. (2001) *The First SIAM Int. Conference on Data Mining, (SDM).*
[4] Lee W. and Stolfo S. (1998) *The 7th USENIX Security Symposium.*

[5] Abdullah K., Lee C., Conti G. and Copeland J.A. (2005) *The IEEE Workshop on Information Assurance Workshop.*
[6] Yu J.Q., Reddy Y.V.R., Selliah S., Kankanahalli S., Reddy S. and Bharadwaj V. (2004) *TRINETR: An Intrusion Detection Alert Management System*, 235-240.
[7] Yin Q., Shen L., Zhang R. and Li X. (2004) *World Congress on Intelligent Control and Automation*, 4370-4374.
[8] Chau M., Xu J.J. and Chen H. (2002) *National Conference on Digital Government Research*, 271-275.
[9] Cabrera J.B.D., Lewis L. and Mehra R.K. (2001) *SIGMOD Record*, 30 (4), 25-34.