A New Data Mining Approach for Network Intrusion Detection

Mr. Narayan Arjunwadkar(Student),
*[*narjunwadkar@gmail.com*]*
Sinhgad Institute Of technology
(Affiliated to University Of Pune),
Lonavala, Pune-410401.

Mr. Jeevan Barguje(Student),
[jeevanbarguje1@gmail.com]
Sinhgad Institute Of technology
(Affiliated to University Of Pune),
Lonavala, Pune-410401.

Mr.Rohit Barde(Student),
[rohitbarde.007@gmail.com]
Sinhgad Institute Of technology
(Affiliated to University Of Pune),
Lonavala, Pune-410401.

Mr. Sachin Hirave(Student),
[hiravesachin07@gmail.com]
Sinhgad Institute Of technology
(Affiliated to University Of Pune),
Lonavala, Pune-410401.

Prof. N.K.Patil (Guide)
[*nkpatil@linuxmail.org*]
Sinhgad Institute Of technology
(Affiliated to University Of Pune),
Lonavala, Pune-410401.

ABSTRACT

In Information Security, intrusion detection is the act of detecting actions that attempt to compromise the confidentiality, integrity or availability of a resource. Intrusion detection does not, in general, include prevention of intrusions. In this paper, we are mostly focused on data mining techniques that are being used for such purposes. We debate on the advantages and disadvantages of these techniques. Finally we present a new idea on how data mining can aid IDSs.

General Terms- Security, Data mining

Keywords- Denial of Service, Data mining, IDS, Network security

I. 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.

II. IDS TAXONOMY

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 implementation 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 soft- ware operates.

III. DATA MINING WHAT IS IT?

Data mining (DM), also called Knowledge-Discovery and Data Mining, is the process of automatically searching large volumes of data for patterns using association rules. Data mining, the extraction of hidden predictive information from large databases, is a powerful new technology with great potential to help companies focus on the most important information in their data warehouses. Data mining tools predict future trends and behaviors, allowing businesses to make proactive, knowledge-driven decisions. The automated, prospective analyses offered by data mining move beyond the analyses of past events provided by retrospective tools typical of decision support systems. Data mining tools can answer business questions that traditionally were too time consuming to resolve. They scour databases for hidden patterns, finding predictive information that experts may miss because it lies outside their expectations.

Most companies already collect and refine massive quantities of data. Data mining techniques can be implemented rapidly on existing software and hardware platforms to enhance the value of existing information resources, and can be integrated with new products and systems as they are brought on-line. When implemented on high performance client/server or parallel processing computers, data mining tools can analyze massive databases to deliver answers to questions such as, "Which clients are most likely to respond to my next promotional mailing, and why?"

This white paper provides an introduction to the basic technologies of data mining. Examples of profitable applications illustrate its relevance to today's business environment as well as a basic description of how data warehouse architectures can evolve to deliver the value of data mining to end users.

The most commonly used techniques in data mining are:

Artificial neural networks*:* Non-linear predictive models that learn through training and resemble biological neural networks in structure.

Decision trees*:* Tree-shaped structures that represent sets of decisions. These decisions generate rules for the classification of a dataset. Specific decision tree methods include Classification and Regression Trees (CART) and Chi Square Automatic Interaction Detection (CHAID).

Genetic algorithms*:* Optimization techniques that use processes such as genetic combination, mutation, and natural selection in a design based on the concepts of evolution.

Nearest neighbor method*:* A technique that classifies each record in a dataset based on a combination of the classes of the k record(s) most similar to it in a historical dataset (where k ³ 1). Sometimes called the k-nearest neighbor technique.

IV. DATA MINING AND IDS

Data mining techniques can be differentiated by their different model functions and representation, preference criterion, and algorithms. The main function of the model that we are interested in is classification, as normal, or malicious, or as a particular type of attack. We are also interested in link and sequence analysis.

Additionally, data mining systems provide the means to easily perform data summarization and visualization, aiding the security analyst in identifying areas of concern. The models must be represented in some form. Common representations for data mining techniques include rules, decision trees, linear and non-linear functions (including neural nets), instance-based examples, and probability models.

A. Off Line Processing

The use of data mining techniques in IDSs, usually implies analysis of the collected data in an offline environment. There are important advantages in performing intrusion detection in an offline environment, in addition to the real-time detection tasks typically employed.

Below we present the most important of these advantages:

- In off-line analysis, it is assumed that all connections have already finished and, therefore, we can compute all the features and check the detection rules one by one.
- The estimation and detection process is generally very demanding and, therefore, the problem cannot be addressed in an online environment because of the various the real- time constraints. Many real-time IDSs will start to drop packets when flooded with data faster than they can process it.
- An offline environment provides the ability to transfer logs from remote sites to a central site for analysis during off- peak times.

B. Data Mining and Real Time IDSs

Even though offline processing has a number of significant advantages, data mining techniques can also be used to enhance IDSs in real time. Lee et al were one of the first to address important and challenging issues of accuracy, efficiency, and usability of real-time IDSs. They implemented feature extraction and construction algorithms for labeled audit data. They developed several anomaly detection algorithms. In the paper, the authors explore the use of information-theoretic measures, i.e., entropy, conditional entropy, relative entropy, information gain, and information cost to capture intrinsic characteristics of normal data and use such measures to guide the process of building and evaluating anomaly detection models. They also develop efficient approaches that use statistics on packet header values for network anomaly detection.

A real-time IDS, called ”Judge”, was also developed to test and evaluate the use of those techniques. A serious limitation of their approaches (as well as with most existing IDSs) is that they only do intrusion detection at the network or system level. However, with the rapid growth of e-Commerce and e-Government applications, there is an urgent need to do intrusion and fraud detection at the application-level. This is because many attacks
may focus on applications that have no effect on the underlying network or system activities.

C. *Multisensor Correlation*

The use of multiple sensors to collect data by various sources has been presented by numerous researchers as a way to in- crease the

performance of an IDS.

- Lee et al state that using multiple sensors for ID should increase the accuracy of IDSs.
- Kumar states that, Correlation of information from different sources has allowed additional information to be inferred that may be difficult to obtain directly."
- Lee et al. note that, an IDS should consist of multiple co- operative lightweight subsystems that each monitor a separate part (such as an access point) of the entire environment."
- Dickerson and Dickerson also explore a possible implementation of such a mechanism. Their architecture consists of three layers:

  – A set of Data Collectors (packet collectors)

  – A set of Data Processors

  – A Threat analyzer that utilizes fuzzy logic and basically performs a risk assessment of the collected data.

- Honigetal propose a model similar to the one by Dickerson and Dickerson and also has components for feature extraction, model generation and distribution, data labeling, visualization, and forensic analysis.
- Helmeretal state that the use of a data warehouse facilitates the handling of the accumulated data and allows distributed attacks to be more easily detected, providing administrators with additional tools for doing auditing and forensics.

V. DECISION TREE

A decision tree model consists of a set of rules for dividing a large heterogeneous population into smaller, more homogeneous groups with respect to a particular target variable. A decision tree may be painstakingly constructed by hand in the manner of Linnaeus and the generations of taxonomists that followed him, or it may be grown automatically by applying any one of several decision tree algorithms to a model set comprised of pre-classified data. The target variable is usually categorical and the decision tree model is used either to calculate the probability that a given record belongs to each of the categories, or to classify the record by assigning it to the most likely class. Decision trees can also be used to estimate the value of a continuous variable, although there are other techniques more suitable to that task.

Just as with CART, the C4.5 algorithm recursively visits each decision node, selecting the optimal split, until no further splits are possible. However, there are interesting differences between CART and C4.5.

VI. C4.5 ALGORITHM

C4.5 is an algorithm used to generate a decision tree. C4.5 is an extension of Quinlan's earlier ID3 algorithm. The decision trees generated by C4.5 can be used for classification, and for this reason, C4.5 is often referred to as a statistical classifier.

Algorithm

C4.5 builds decision trees from a set of training data in the same way as ID3, using the concept of information entropy. The training data is a set $S = s_1,s_2,...$ of already classified samples. Each sample $s_i = x_1,x_2,...$ is a vector where $x_1,x_2,...$ represent attributes or features of the sample. The training data is augmented with a vector $C = c_1,c_2,...$ where $c_1,c_2,...$ represent the class to which each sample belongs. At each node of the tree, C4.5 chooses one attribute of the data that most effectively splits its set of samples into subsets enriched in one class or the other. Its criterion is the normalized information gain (difference in entropy) that results from choosing an

attribute for splitting the data. The attribute with the highest normalized information gain is chosen to make the decision. The C4.5 algorithm then recurses on the smaller sublists. This algorithm has a few base cases.

- All the samples in the list belong to the same class. When this happens, it simply creates a leaf node for the decision tree saying to choose that class.
- None of the features provide any information gain. In this case, C4.5 creates a decision node higher up the tree using the expected value of the class.
- Instance of previously-unseen class encountered. Again, C4.5 creates a decision node higher up the tree using the expected value.

In pseudocode the algorithm is

1. Check for base cases
2. For each attribute *a*
1. Find the normalized information gain from splitting on *a*
3. Let *a_best* be the attribute with the highest normalized information gain
4. Create a decision *node* that splits on *a_best*
5. Recurse on the sublists obtained by splitting on *a best*, and add those nodes as children of *node*

J48 is an open source java implementation of the C4.5 algorithm in the weka data mining tool.

Improvements from ID3 algorithm

C4.5 made a number of improvements to ID3. Some of these are:

- Handling both continuous and discrete attributes - In order to handle continuous attributes, C4.5 creates a threshold and then splits the list into those whose attribute value is above the threshold and those that are less than or equal to it.
- Handling training data with missing attribute values - C4.5 allows attribute values to be marked as ? for missing. Missing attribute values are simply not used in gain and entropy calculations.
- Handling attributes with differing costs.
- Pruning trees after creation - C4.5 goes back through the tree once it's been created and attempts to remove branches that do not help by replacing them with leaf nodes.

The C4.5 *algorithm* is Quinlan's extension of his own ID3 algorithm for generating decision trees . Just as with CART, the C4.5 algorithm recursively visits each decision node, selecting the optimal split, until no further splits are possible. However, there are interesting differences between CART and C4.5.

- Unlike CART, the C4.5 algorithm is not restricted to binary splits. Whereas CART always produces a binary tree, C4.5 produces a tree of more variable shape.

- For categorical attributes, C4.5 by default produces a separate branch for each value of the categorical attribute. This may result in more "bushiness" than desired, since some values may have low frequency or may naturally be associated with other values.

- The C4.5 method for measuring node homogeneity is quite different from the CART method and is examined in detail below.

C4.5 Algorithm

In general, steps in C4.5 algorithm to build decision tree are:

- Choose attribute for root node

- Create branch for each value of that attribute

- Split cases according to branches

- Repeat process for each branch until all cases in the branch have the same class

Choosing which attribute to be a root is based on highest gain of each attribute. To count the gain, we use formula 1,below:

$$Gain(S, A) = Entrophy(s) - \sum_{i-1}^{n} \frac{|si|}{|s|} * Entrophy(si)$$

with

{S1, ..., Si, …, Sn} = partitions of S according to values of

attribute A

n = number of attributes A

|Si| = number of cases in the partition Si

|S| = total number of cases in S

$$Entrophy(s) = \sum_{i=1}^{n} - pi * \log_2 pi$$

with

S : Case Set

n : number of cases in the partition S

pi : Proportion of Si to S

VII. CONCLUSION

The application we have built can produce decision tree that conforms to variables and case's data from network. Accuracy level of the prediction data of this application is very depended to chosen variable that will be the basis to make the decision tree. For the next improvement research, we can explore for variable(s) that can produce highest data accuracy level.

VIII. REFERENCES

1. Christos Douligeris, Aikaterini Mitrokotsa, "DDoS attacks and defense mechanisms: classification and state-of-the-art" ,Computer Networks: The International Journal of Computer and Telecommunications Networking, Vol. 44, Issue 5

, pp: 643 - 666, 2004.

2. Eric Bloedorn et al, "Data Mining for Network Intrusion Detection: How to Get Started," Technical paper.

3. Presentation on Intrusion Detection Systems, Arian Mavriqi.

4. Larose, Daniel T., Discovering Knowledge in Data: an Introduction to Data Mining, John Wiley and Sons, USA.