A Novel Approach to Improve Detection Rate and Search Efficiency of NIDS

Prof. (Mrs) Manisha R. Patil[1], Mrs. Madhuri D. Patil[2]

[1] Professor, Department of Computer Engineering, Smt. Kashibai Navale College of Engineering, Pune, India
[2] Student, Department of Computer Engineering, Smt. Kashibai Navale College of Engineering, Pune, India

Abstract

Signature based Network Intrusion Detection System (NIDS) applies a set of rules to identify the traffic and classify known attacks by comparing with the signature. As the detection rate and speed of searching for signature in NIDS have two main aspects, this paper gives data mining approve to improve on detection rate and we use an algorithm to use the known signature to find the signature of the related attack quickly.

Keywords: Classification, Signature Based NIDS, Association Rule Mining, Data Mining, Apriori Algorithm, Network Intrusion Detection.

1. INTRODUCTION

Information is an important asset in an organization which has large amount of personal and critical data with it. Protecting that information from attacks should be the main goal when security is concerned. Intrusion Detection System (IDS) are software or tools that monitor events that take place in a computer or a network, looking for evidence of intrusion [1] .The process of monitoring the events occurring in a computer system or network and analyzing them for sign of intrusions is known as Intrusion Detection System (IDS) [3]. Network Intrusion Detection Systems just analyze the traffic on network and sets alarm for the attack detection, there are two type of NIDS that are anomaly based NIDS and Signature base NIDS. Anomaly based NIDS tries to determine whether deviation from the established normal usage patterns can be flagged as intrusion The normal usage patterns are constructed from the statistical measures of the system features, for example, the CPU and I/O activities by a particular user or program[3]. The behavior of the user is observed and any deviation from the constructed normal behavior is detected as intrusion and signature based NIDS tries to flag intrusion by comparing signatures of attacks with the incoming packets on network [2]. There are two advantages of signature based NIDS. The first is it detects attack without generating overwhelming number of false alarms. The second is that it can quickly diagnose the use of specific attack tool [2] and on other side disadvantage of signature based NIDS is it can only detect known attacks. But in most of the networks signature based NIDS are preferred. The main two problems with signature based NIDS are its detection rate and signature search is time consuming and error prone work. To solve first problem till now data mining is used with GP (genetic programming) to improve on detection rate, in this paper we are going to propose purely a data mining concept to improve on detection rate and second problem was solved until Signature Apriori [5] was proposed. But the signature Apriori waste much time for generate the unnecessary candidate item sets and scan the database .If the size of database is large, the Signature Apriori will be not effective in the signature search . In this system, we used Modified Signature Apriori algorithm to search for the attack quickly.

2. Classification Algorithm

In NIDS the attacks are detected same way as how the classification works. In order to classify the network attacks, we used a well classification algorithm that is C4.5 which is one of old and comparatively good algorithm. Dataset input to C4.5 algorithm is KDD99 which contains attack records with 41 attributes for each connection record plus one class label. The raw data was processed into connection records, which consist of about five million connection records [6] [7]. C4.5 finds the gain ratio of the attributes and uses an attribute for classification whose gain ration is highest. Gain ratio is calculated as (1)

$$\text{SplitInfo}_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right) \qquad (1)$$

This value represents the potential information generated by splitting the training data set, D, into v partitions, corresponding to the v outcomes of a test on attribute A. Note that, for each outcome, it considers the number of records having that outcome with respect to the total number of records in D. It differs from information gain, which measures the information with respect to classification that is acquired based on the same partitioning. The gain ratio is defined as: The expected information needed to classify a record in D is given by

$$Info(D) = -\sum_{i=1}^{m} pi \log_2(p_i) \qquad (2)$$

Where pi is the probability that an arbitrary record in D belongs to class Ci and is estimated by $|C_{i,D}|/|D|$.

A log function to the base 2 is used, because the information is encoded in bits. Info(D) is just the average amount of information needed to identify the class label of a tuple in D. Note that, at this point, the information we have is based solely on the proportions of tuples of each class. Info(D) is also known as the entropy of D.

$$\text{InfoA(D)} = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \text{Info}(D_j) \qquad (3)$$

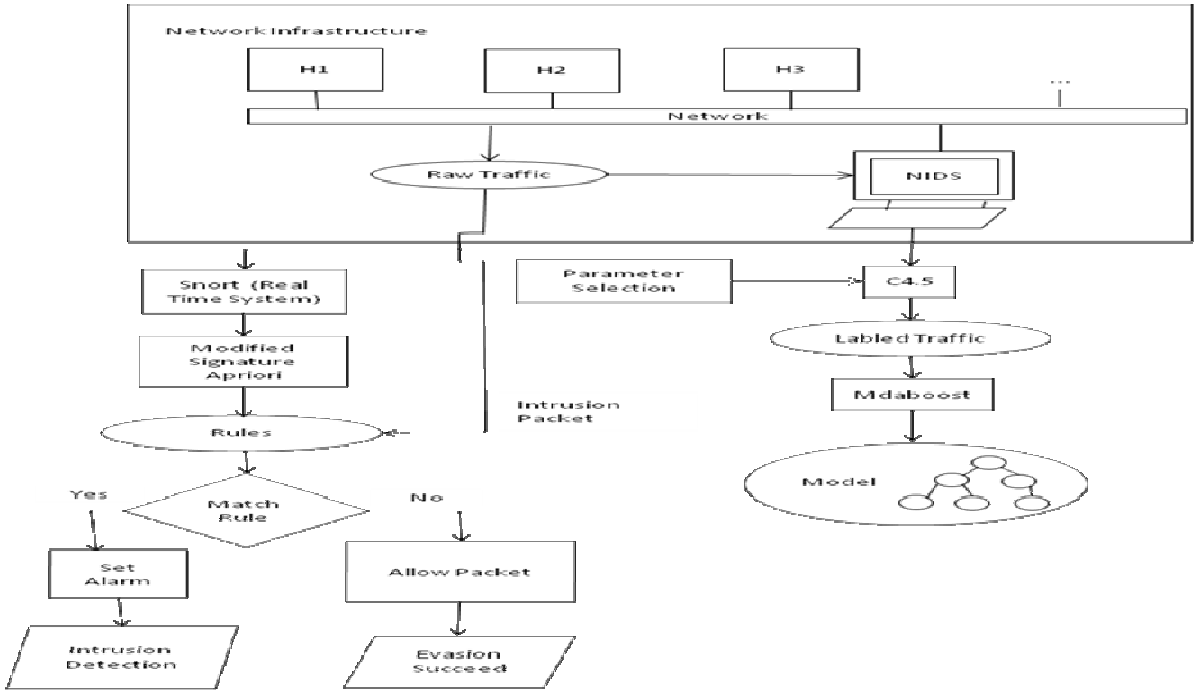
$$\text{Gain (A)} = \text{Info(D)} - \text{Info}_A\text{(D)} \qquad (4)$$

$$\text{GainRatio (A)} = \frac{\text{Gain(A)}}{\text{SplitInfo(A)}} \qquad (5)$$

In this way the detection of attacks is done and these attacks are represented as decision tree.

3. Ensemble Algorithm

The decision tree is given as text file to the ensemble algorithm. The Adaboost algorithm is one of the good ensemble algorithm but there are two drawbacks with this algorithm. (1) AdaBoost cannot be used in the boosting by filtering framework, and (2) AdaBoost does not seem to be noise resistant. In order to solve them, there is a new boosting algorithm MadaBoost by modifying the weighting system of AdaBoost [5]. So the ensemble algorithm used in this system is Mdaboost. This algorithm improves the accuracy of the classification of attack records and with this improvement in classification will automatically improve the detection rate of NIDS.



4. Signature Apriori Algorithm

The concept of Signature Apriori is based on Apriori algorithm [8]. The Apriori algorithm is an algorithm for mining frequent itemsets. This algorithm uses the prior knowledge of frequent itemset properties. Apriori employs an iterative approach, where k-itemsets are used to explore (k+1)-itemsets. First, the set of frequent 1-itemsets is found.

This set is denoted as L1. L1 is used to find L2, the frequent 2-itemsets, which is used to find L3 and so on, until no more frequent k- itemsets can be found. The finding of each LK requires one full scan of the database.

5. Modified Signature Apriori Algorithm

The steps of how to find out frequent k-item sets will be as follow. At the first step, all of the frequent items will be found. And then we use a simple way to scan the database in order to find the frequency of occurrence of each item, and decide which one meets the minimum support. Secondly, we generate the candidate n-item sets by checking all of the possible combinations of the frequent items with already known signatures, if they meet the minimum support requirement. Then, append this n-itemsets from right. We can first append the backward, until the minimum support is unsatisfied. Then, we append forward, and stop when the same condition occurred. Finally, the maximum length of frequent-item set can be mined by our method. A simple way to find frequent item set is we read one transaction each time from database and then count the support of each different item. If an item occurs twice in the same transaction, the support count of this item will increase once. Repeat until no transactions available in database. Finally, we will check all items in candidate 1-itemset and append the item that meet the minimum support into L1. After all frequent items have been mined; we will stop generating all possible candidate 2-itemsets we generate the candidate itemsets only related the known signatures. Then, all of the frequent items will be concatenated to the known signature and put them into candidate n itemsets. After that, check all item sets in the candidate n-itemset. Then, add the itemsets that meet the minimum support into L1. The improvement in the algorithm is we know that Ci+1 is generated from Li* Li. Clearly, a Ci'generated from Ci * Ci, instead of from Li* Li, will have a greater size than | Ci | where Ci is generated from Li* Li. However, if | Ci'| is not much larger than | Ci |, we may save one round of database scan. This technique is called scan-reduction [9].

Next, we take an example below to show how the proposed algorithm works. We assume that the transactions in the database are {{A B C D E F G Q}, {M N A B C D E F G}, {J A B C D E F G}, {P Q I}}. The attack signature we have already known is {C D E}. Let the minimum support be 0.7. Applying the proposed algorithm, we can firstly get the frequent items L1= {A B C D E F G}. In order to find out the derived attack signature we expanded the known signature by each frequent item, and we then we have Cn= {{C D E A}, {C D E B}, {C D E C}, {C D E D}, {C D E E}, {C D E F}, {C D E G}} at the first stage. After we have Cn candidate itemsets, we scan the database to find out the Ln={C D E F}. Then we let the Ln be the new attack signature that we have already known. Repeating the step until the minimal support is no longer satisfied. We win get the Ln = {C D E F G} in this example. Next, we expand the Ln in the inversed direction. Finally, we will get the possibility attack signature Ln = {A B C D E F G} [2].

REFERENCES

[1] R. Bace and P. Mell, "NIST Special Publication on Intrusion Detection Systems", 800-31, 2001

[2] Yang,X.R.,Song,Q.B.and Shen, J.Y,"Implementation Of Sequence Patterns Mining In Network Intrusion Detection System", in Proceeding of ICII,2001. Pp.323- 326.

[3] Hu Zhengbing,Li Zhitang, "A Novel Network Intrusion Detection System(NIDS) Based on Signatures Search of Data Mining",10-16,2008

[4] S. Peddabachigari, A. Ajith, C. Grosan, and J. Thomas, "Modeling intrusion detection system using hybrid intelligent systems," Journal in Network Computer Applications, vol. 30, no. 1, pp. 114-132, 2007.

[5] Carlos Domingo, Osamu Watanabe "MadaBoost: A Modification of AdaBoost" Thirteenth Annual Conference on Computational Learning Theory Pages: 180-189 Year of Publication: 2000 ISBN:1-55860-703-X

[6] Han, H., Lu, Lu, X.L., and Ren, L.Y.,"Using Data Mining to Discover Signatures in Network-Based intrusion detection", in Proceeding of IEEE Computer Graphics and Applications,2002. pp.212-217

[7] MIT Lincoln Laboratory. http://www.ll.mit.edu/IST/ideval/

[8] KDD cup 99 Intrusion detection data set. Web site of the data set is as bellow http://kdd.ics.uci.edu/databases/kddcup99/kddcup.data_10_percent.gz

[9] Rakesh, A., and Srikant, R.,"Fast Algorithm For Mining Association Rules", in Proceeding of the 20th international Conference on VLDB, 1994

[10] Park, J.S., Chen, M.S., and Yu, P.S., "Using a Hash – Based Method With Transaction Trimming For Mining Association Rules", Knowledge and Data Engineering, IEEE Transaction, 1997