Volume 6, Issue 11, November 2016 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Evaluating Very Fast Decision Tree (VFDT) Algorithm for Detecting Network Intrusion

K. V. Keerthi, K. Vidhyavathi, M. Swathi, K. Suresh
CSE & JNTUA, Andhra Pradesh,
India

Abstract— With recent advances in network based technology needs protecting computers and networks which becomes a huge problem. Based on information coming from various response teams a computer was attacked or broken into more than once per second. In this paper, two grains levels intrusion detection system (IDS) is suggested fine-grained and coarse-grained. In normal case the intrusions are not detected, to improve the performance the most suitable IDS level is the coarse-grained. Any intrusion is detected by coarse-grained IDS after that the fine-grained is used to detect the possible attack details. Very fast decision tree (VFDT) algorithm is used in both of these detection levels. In order to ensure efficiency of the proposed model, it has been tested on KDD CUP 99 dataset and a real traffic dataset. Experimental results demonstrate that the proposed model is highly successful in detecting known and unknown attacks

Keywords— very Fast Decision Tress, Intrusion Detection, Knowledge Discovery Dataset, Coarse-Grained IDS, Fine-Grained IDS

I. INTRODUCTION

An intrusion detection system (IDS) inspects all network activity and identifies suspicious patterns that may indicate a network or system attack from someone attempting to break into or compromise a system. With the rapid growth in network, intrusions in computers have increased rapidly. Intrusion Detection System is an essential component of a complete defence-in-depth architecture for network security. It collects and inspects packets, looking for evidence of intrusive behaviours. Whenever intrusive event is detected, an alarm is raised giving the security analyst an opportunity to react promptly. Most of designed IDSs cannot cope with fast networks. Although several IDS systems are available, the common objectives of these systems are to reduce the amount of false alarms, and to recognize new attacks in order to increase detection ratio. In this paper, the concentration is on detecting attacks in fast networks in order to mitigate the influence of the attack by reducing the time gap between the real attack and its detection. This paper contributes to build two grains levels IDS in order to detect abnormal behaviour of network traffic and cope with fast networks i.e. fine-grained and coarse-grained. It is well known that the intrusion occurrence in networks with respect to general traffic is rare. These motivate us to build the proposed two grains levels IDS they are fine-grained and coarse-grained. In normal case, where intrusions are not detected, the most suitable IDS level is the coarse-grained to increase performance. At the moment of intrusion is detected by coarse-grained IDS, the fine-grained IDS is used to detect as most as possible of attack details.. The coarse-grained Intrusion Detection System focuses on five packet features while fine-grained Intrusion Detection System works on 20 features. Very Fast Decision Tree (VFDT) algorithm is selected as a fast classifier. The advantages of this system is processing and analysing of high-speed network traffic, discovering and accurately identifying new attacks to reduce the false alarms to an maximum extent, and detecting the intrusion in real time.

DARPA KDD CUP 99 dataset is used as a bench-mark for the proposed IDS, which contains 41 features. we analysed these features and selected 20 features having information gain ratio over the average of the dataset. Then, we trained and tested the proposed system.

II. RELATED WORK

1. Intrusion detection and attack classified on three techniques

In recent times, different soft-computing methods have been proposed for the development of intrusion detection systems. The main purpose of this work is to develop, implement and evaluate an anomaly off-line based intrusion detection system(IDS) using three techniques; data mining association rules, decision trees(ID3 algorithm), and artificial neural network, then comparing among them to decide which technique is better in performing for intrusion detection system. Many methods have been proposed to modify these techniques to improve the classification process. For association rules, the major vote classifier was modified to build a new classifier that can recognize anomalies. By decision trees, ID3 algorithm was modified to deal not only with discreet data, but also to deal with numerical data. For neural networks, a back-propagation algorithm has been used as the learning algorithm with different number of inputs (118, 51, and 41) to initiate the important knowledge about the intruder to the neural networks. Different methods of normalization were applied on the input patterns to speed up the learning process. The full 10% KDD 99 train dataset and the full correct test dataset are used in this work. The proposed techniques results show that there is an improvement in the performance comparing to the standard techniques, further the Percentage of Successful Prediction (PSP) and Cost

Per Test (CPT) of neural networks and decision trees are better than association rules. On the other hand, the training time for neural network takes longer time than the decision trees.

2. Multi-Level Intrusion Detection System (ML-IDS)

With the increase in deployment of network-centric systems, there is a proportional increase in intensity as well as complexity of network attacks. Attack detection techniques can be classified as being signature-based, classification-based, or anomaly-based. In this paper we present a multi level intrusion detection system (ML-IDS) which uses autonomic computing to automate the control and management of Multi-Level-IDS. This automation allows ML-IDS to detect intrusion in network attacks and proactively protect against them. ML-IDS inspects and analyses network traffic using three levels of granularities (network traffic flow, packet header, and payload), and employs an efficient fusion decision algorithm to improve the overall intrusion detection rate and reduce the occurrence of false alarms. We have individually evaluated each of our approaches against a huge range of network attacks, and then compared the results of these approaches with the results of the combined decision fusion algorithm.

3. Decision Tree Intrusion Detection (Id3 Algorithm)

Classification algorithms create a decision tree like the one presented in by identifying different patterns in an existing dataset and using that information to create the tree. Decision tree algorithms take pre-classified data as input. They learn the patterns in the data and apply simple rules to differentiate between the different types of data in the pre-classified data set. The ID3 algorithm begins with the original set as the root node. On each iteration of the algorithm, it iterates through every unused attribute of the set and calculates the entropy of that attribute. It then selects the attribute which has the smallest entropy (or largest information gain) value. The set is then split by the selected attribute (e.g. age is less than 50, age is between 50 and 100, age is greater than 100) to produce subsets of the data. The algorithm continues to recourse on each subset, considering only attributes never selected before Recursion on a subset may stop in one of these cases: Every element in the subset belongs to the same class , then the node is turned into a leaf and labelled with the class of the examples There are no more attributes to be selected, but the examples still do not belong to the same class, then the node is turned into a leaf and labelled with the most common class of the examples in the subset

4. An Intrusion Detection System Based on KDD-99 Data using Data Mining Techniques and Feature Selection

Day by day Internet and internet users are increasing. Due to rapid development of internet technology, security is becoming big issue. Intruders are monitoring computer network system continuously for attacks. A sophisticated firewall with efficient intrusion detection system (IDS) is needed to prevent computer network from attacks. A comprehensive study of literatures proves that data mining techniques are more powerful technique to develop IDS as a classifier. Performance of classifier is a crucial issue in terms of its efficiency, also number of feature to be scanned by the IDS should also be optimized. In this paper two techniques C5.0 and artificial neural network (ANN) are utilized with feature selection. Feature selection techniques will discard some irrelevant features while C5.0 and ANN acts as a classifier to classify the data in either normal type or one of the five types of attack.KDD99 data set is used to train and test the models ,C5.0 model with numbers of features is producing better results with all most 100% accuracy. Performances were also verified in terms of data partition size.

III. SYSTEM ARCHITECTURE

The proposed system consists of four processing stages, which are data collection, pre-processing, classification, and response. Both IDS levels require adequate connection information to train the proposed model. Therefore, the system is doing update the information at any time to obtain a sufficient number of connections which enable building a decision tree for attacks. After connection/packet information is available, a VFDT algorithm is applied in one of the IDS levels in order to do a classification and make the decision (either normal or attack). In case of an attack is detected, a report is generated providing information of the attacks, e.g. IP addresses, time … etc.

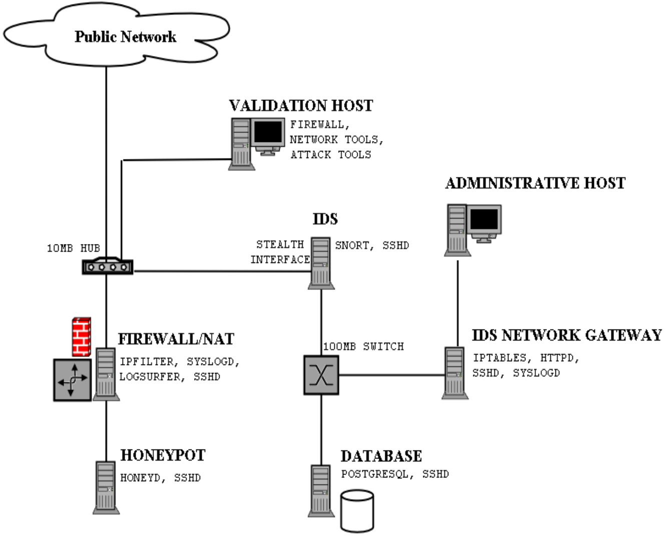


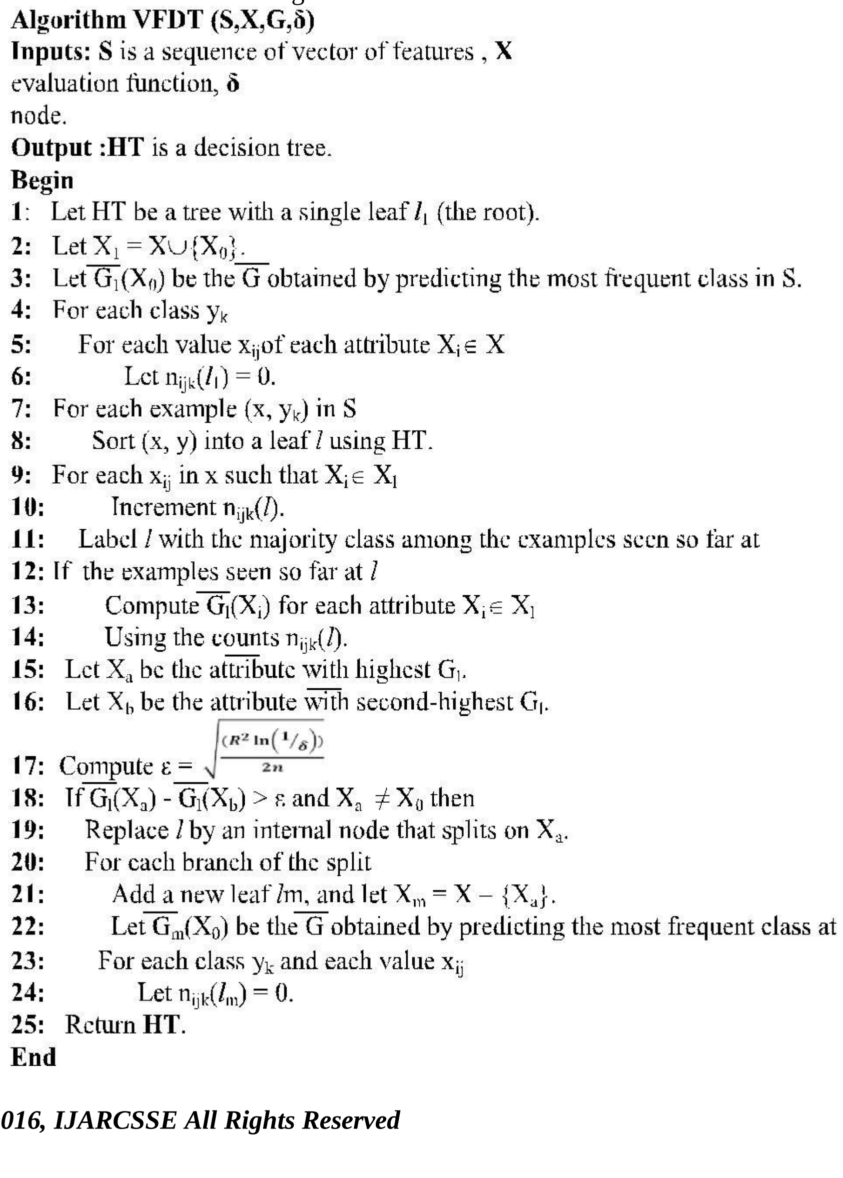
Fig 1: system architecture

The main task of the two grains level IDS is to identify intrusion patterns by considering the features that are extracted from packets. The VFDT detector constructs a decision tree by using constant memory and constant time per sample. The tree is built by recursively replacing leaves with decision nodes. Sufficient statistics of attribute values are stored in each leaf. Heuristic evaluation function is used to determine split attributes converting from leaves to nodes. Nodes contain the split attributes and leaves contain only the class labels. The leaf represents a class that the sample labels. When a sample enters, it traverses the tree from root to leaf, evaluating the relevant attribute at every single node. After the sample reaches a leaf the existing statistics are updated. At this time, the system evaluates each possible condition based on attribute values: if the statistics are sufficient to support the one test over the other, a leaf is converted to a decision node. The decision node contains the number of possible values for the chosen attribute of the installed split test. A decision tree to classify the attacks may become large. However, the error rate is high because it becomes very complicated tree. Therefore, decision tree is pruned to minimize error rate and becomes more simply and easily understandable.

The frequency of computer intrusions has increased rapidly during the last two decades. Intrusion Detection Systems (IDSs) are an essential component of a complete defence-in-depth architecture for network security. They collect and inspect packets, looking for evidence of intrusive behaviours. As soon as an intrusive event is detected, an alarm is raised giving the security analyst an opportunity to react promptly. Unfortunately, most of designed IDSs cannot cope with fast networks. Although several IDS systems are available, the common objectives of these systems are to reduce the amount of false alarms, and to recognize new attacks in order to increase detection ratio. In this paper, the concentration is on detecting known and unknown attacks in fast networks in order to mitigate the influence of the attack by shrinking the time gap between the real attack and its detection.

VFDT(Very Fast Decision Tree) Algorithm

VFDT is a very high-performance data mining system based on decision trees. So many of classification learning methods have been proposed, of which one of the commonly used method is decision tree learning method. Because it is fast and the classifiers description that it derives is easily understood. VFTD is one of the data stream algorithms that support the decision tree learning method. As if any data arrives, this data stream grows gradually while the data is classified. VFDT allows the attribute evaluation measure as use of either information gain or the Gini-index. It contains a number of refinements to the algorithm.



```
Algorithm VFDT (S,X,G,δ)
Inputs: S is a sequence of vector of features , X
evaluation function, δ
node.
Output :HT is a decision tree.
Begin
Let HT be a tree with a single leaf $l_1$ (the root).
Let $X_1 = X \cup \{X_0\}$.
Let $\overline{G}_1(X_0)$ be the $\overline{G}$ obtained by predicting the most frequent class in S.
For each class $y_k$
    For each value $x_{ij}$of each attribute $X_i \in X$
        Let $n_{ijk}(l_1) = 0$.
For each example $(x, y_k)$ in S
    Sort (x, y) into a leaf $l$ using HT.
For each $x_{ij}$ in x such that $X_i \in X_l$
    Increment $n_{ijk}(l)$.
    Label $l$ with the majority class among the examples seen so far at
If  the examples seen so far at $l$
    Compute $\overline{G}_l(X_i)$ for each attribute $X_i \in X_l$
    Using the counts $n_{ijk}(l)$.
Let $X_a$ be the attribute with highest $G_l$.
Let $X_b$ be the attribute with second-highest $G_l$.
Compute $\varepsilon = \sqrt{\frac{(R^2 \ln(1/\delta))}{2n}}$
If $\overline{G}_l(X_a) - \overline{G}_l(X_b) > \varepsilon$ and $X_a \neq X_0$ then
    Replace $l$ by an internal node that splits on $X_a$.
    For each branch of the split
        Add a new leaf $l$m, and let $X_m = X - \{X_a\}$.
        Let $\overline{G}_m(X_0)$ be the $\overline{G}$ obtained by predicting the most frequent class at
    For each class $y_k$ and each value $x_{ij}$
        Let $n_{ijk}(l_m) = 0$.
Return HT.
End
```

Two Level IDS Algorithm

A fine-grained IDS system working on SF set of features. Coarse-grained IDS system working on *BF* set of features. The two-level system is composed of two grains levels IDS which allows the system to analyse network traffic on different granularities. The two levels IDS differs from available IDSs in which it adapts with network situation when it is under attack or not. Its detection levels are Coarse-grained IDS and fine-grained IDS. In normal case, where network intrusions are not detected, the most suitable level is the Coarse-grained IDS in which five features are monitored to increase IDS performance. At the moment where intrusion is detected by grained Coarse- IDS, the fine-grained IDS is activated where 20 features are monitored to detect as more as possible of attack details. VFDT algorithm is selected as classifier to achieve this goal because VFDT is capable of processing and analysing of high-speed network traffic, and detecting the intrusion in real time.

```
Algorithm two grain
Begin
    Mode = H_Level
    While (P=packet capturing ()) {
        Preprocess(P0
        if (Mode=H_Level)
        //coarse-grained IDS
        E=BF (P)
        Classify E using VFDT model
        if (any attack is detected) {
        Mode=C_Level
        Generate Alerts}
        else{
            //fine-grained IDS
        E=SF(P) where E is build from a set of p
        classify E using VFDT model
        if(any attack is detected)
        Generated Alerts
        if (no attack is detected within a specific period )
        Mode=H_Level
    }
End
```

IV. CONCLUSIONS

Coarse-grained IDS and Fine grained IDS is proposed allowing the system to analyse network traffic on different granularities. It is different from the available IDS in which it adapts with network situation when it is under attack or not. Its detection levels are coarse-grained IDS and fine grained IDS. It improves the accuracy of the generalization tree and new attacks detection

REFERENCES

[1] R. Perdisci, G. Giacinto, F. Roli, *Alarm clustering for intrusion detection systems in computer networks*, J. Eng. Appl. Artif. Intell. 19 (2006) 429e438.

[2] D. Pedro, H. Geoff, *Mining high speed data streams, in: ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2000, pp. 71e80.

[3] Mohammed M. Mazid, M. Shawkat Ali, Kevin S. Tickle, *A comparison between rule based and association rule mining algorithms*, in: 3rd IEEE International Conference on Network and System Security, 2009, pp. 452e455.

[4] G. Radhika, S. Anjali, C.J. Ramesh, *Parallel misuse and anomaly detection model*, Int. J. Netw. Secur. 14 (4) (2012) 211e225.

[5] P. Mrutyunjaya, R.P. Manas, *Evaluating machine learning algorithms for detecting network intrusions*, Int. J. Recent Trends Eng. 1 (1) (2009).

[6] Dewan M. Farid, H. Nouria, B. Emna, Z.R. Mohammad, M.R. Chowdhury, Attacks *classification in adaptive intrusion detection using decision tree*, World Acad. Sci. Eng. Technol. (2010) 27e44.

[7] A.N. Huy, D. Choi, Application *of data mining to network intrusion detection: classifier selection model, in: Asia-Pacific Network Operation and Management Symposium*, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 399e406.

[8] M. Adnan, B. Abdulazeez, S.I. Adel, *Intrusion detection and attack classifier based on three techniques, A Comp. Study*. Eng. Technol. J. 29 (2) (2011) 233e254.

[9] M.F. Kamel, B. Aoued, *Securing network traffic using genetically evolved transformations*, Malays. J. Comput. Sci. 19 (2006) 3e23.

[10] S. Staniford, J.A. Hoagland, J.M. McAlerney, *Practical automated detection of stealthy portscans*, J. Comput. Secur. 10 (1e2) (2002) 105e136.

[11] E. Eskin, A. Arnold, M. Preraua, L. Portnoy, S.J. Stolfo, *A geometric framework for unsupervised anomaly detection: detecting intrusions in unlabeled data*, in: D. Barbar, S. Jajodia (Eds.), Data Mining for Security Applications, Kluwer Academic Publishers, Boston, 2002.

[12] A. Honig, A. Howard, E. Eskin, S.J. Stolfo, *Adaptive model generation: an architecture for the deployment of data mining based intrusion detection systems*, in: D. Barbar, S. Jajodia (Eds.), Data Mining for Security Applications, Kluwer Academic Publishers, Boston, 2002.