Effective Value of Decision Tree with KDD 99 Intrusion Detection Datasets for Intrusion Detection System

Joong-Hee Lee[†], Jong-Hyouk Lee[†], Seon-Gyoung Sohn[‡], Jong-Ho Ryu[‡], and Tai-Myoung Chung[†]

Internet Management Technology Laboratory,
[†] Electrical and Computer Engineering, Sungkyunkwan University,
300 Cheoncheon-dong, Jangan-gu, Suwon-si, Gyeonggi-do, 440-746, Korea,
[‡] Electronics and Telecommunications Research Institute (ETRI),
161 Gajeong-dong, Yusung-gu, Daejeon-si, 305-700, Korea
Email: {jhlee00, jhlee}@imtl.skku.ac.kr, {sgsohn, ryubell}@etri.re.kr, tmchung@imtl.skku.ac.kr

Abstract—A decision tree is a outstanding method for the data mining. In intrusion detection systems (IDSs), the data mining techniques are useful to detect the attack especially in anomaly detection. For the decision tree, we use the DARPA 98 Lincoln Laboratory Evaluation Data Set (DARPA Set) as the training data set and the testing data set. KDD 99 Intrusion Detection data set is also based on the DARPA Set. These three entities are widely used in IDSs. Hence, we describe the total process to generate the decision tree learned from the DARPA Sets. In this paper, we also evaluate the effective value of the decision tree as the data mining method for the IDSs, and the DARPA Set as the learning data set for the decision trees.

I. INTRODUCTION

Intrusion detection system (IDS) is the most essential part of the security infrastructure for the networks connected to the Internet, because of the numerous ways to compromise the stability and security of the network. The IDS is useful to detect, identify and track the intruders. Particularly, network-based IDSs (NIDSs) analyze the network traffic coming into the network to be protected in order to detect and classify the attacks. According to the detection approaches, they can be divided into a misuse detection and a anomaly detection [1]. Most misuse detections are based on signatures of the attacks. The signature has to be defined by the specialist of the security after the attack is recognized and analyzed. After the definition of the signature, all of the incoming traffics into the network are collated with the signature to be judged whether the traffic tries to attack or not. Therefore, the misuse detection based on the signatures of the attacks has effective performance to detect the known attacks. Anomaly detection is an approach to the suspicious traffics compared with the normal traffics in order to cope with the attack. It has the objective originally to block the attack before the successful performing of the unknown attack. To detect the anomaly traffic, IDS has to have own criterion to perceive the traffic to be the attack. The IDS for the anomaly detection should firstly learn the characteristics of normal activities and abnormal activities, then the IDS detects traffics that deviate from normal activities [2]. For learning and

analysis of the traffic, the classification rule among the data set of the traffic has to be discovered [3]. Therefore, various data mining algorithms can be the solution for the purpose of the anomaly detection.

A decision tree is one of the most powerful and effective method among the various data mining methods. It is expected to be suitable solution for the IDS especially the anomaly detection. Therefore, various approaches applying the decision tree to the IDS have been introduced in many researches such as [5], [6], [7]. To use the decision tree as the criterion of the anomaly detection, a training data set and a evaluation dataset are necessary for the learning and the evaluation of the decision tree. DARPA 98 Lincoln Lab evaluation Data Set (DARPA Set) is the data set universally used for the learning and testing data set in the IDS [8], [9]. This data set is also used in KDD Cup 1999 [4].

Even though the decision tree and the DARPA Set are widely used in IDSs, the the process to generate the decision tree using the DARPA Set is not clearly described in regular order. Therefore, we explain the details how to generate the decision tree using the DARPA Set in this paper, and we also evaluate the performance of the decision tree as the anomaly detection method.

The rest of this paper is organized as follows. In Section 2, we introduce the details of the decision tree including the decision tree learning algorithm and DARPA 98 Lincoln Lab evaluation dataset. Then, the process making the decision tree are explained step by step in Section 3. In Section 4, we explain the architecture of the implementation for the decision tree as the criterion of the IDS, and we describe the scenario for the learning process and the evaluation of the decision tree. In Section 5, we conclude the benefits from the study in this paper.

II. BACKGROUND STUDIES

A. What is Decision Tree?

1) Decision Tree: Decision tree is one of the most powerful and simple data mining method. The decision tree is a kind of a
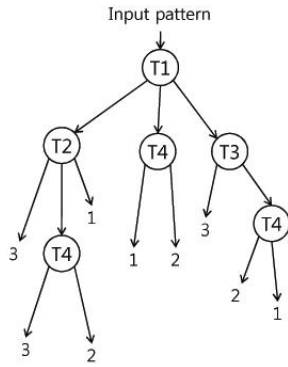
Fig. 1. Simple example of Decision Tree

Fig. 2. Process to make the Decision Tree

tree that consists of branch nodes representing a choice among a number of alternatives, and each leaf nodes representing a class of data. A simple example of decision tree is depicted in Fig. 1. In Fig. 1, branch nodes such as T1, T2, T3, and T4 assign a class number to a input pattern by filtering the pattern down through the tests in the tree. For example, the T3 tests the input pattern down from the T1, and assigns class 3 to the input pattern or passes down to the T4. Finally, any input patterns can be categorized to the class 1, 2, or 3 when the input pattern reaches to the leaf nodes. Therefore, the decision tree is valuable to categorized the data from the large dataset.

2) Learning Algorithm: Learning algorithms for decision tree locates the features to the appropriate position in a decision tree from the learning data set in order to automatically make up the decision tree. There are various decision tree learning algorithms such as ID3, C4.5, and CART [10], [11], [12]. We introduce the ID3 algorithm to make the decision tree in this paper because the ID3 algorithm has a clear concept using Shannon's information theory, and can be simply implemented.

The ID3 algorithm adopts the greedy concept to locate the features in the decision tree, that is, it choices the features from the learning data set according to the correlation between the features and the class. The correlation, that employs the concept of the entropy in the information theory, is calculated by the Eq. 1. Information gain represented shown as the Eq. 2 is to measure the expected reduction in entropy [6].

$$E_S = \sum_{i=1}^{S_{max}} -p_i \log_2(p_i), \qquad (1)$$

where p_i is the proportion of instances in the dataset that take i^{th} value of the target feature

$$Gain(S, A) = E_S - \sum_{\nu \in A} \frac{|S_\nu|}{|S|} E_{S_\nu}, \qquad (2)$$

where ν is a value of feature A, $|S_\nu|$ is the subset of instance of S where A takes the value ν, and $|S|$ is the number of instances
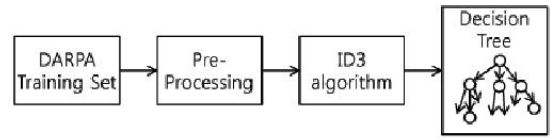
B. DARPA 98 Lincoln Lab Evaluation Data Set

The DARPA Set was defined by the Information Systems Technology Group (IST) of MIT Lincoln Laboratory sponsored by Defense Advanced Research Projects Agency (DARPA ITO) and Air Force Research Laboratory (AFRL/SNHS) [9]. The DARPA Sets provide the data sets for both the learning and the testing. This is widely used in various IDS researches, so it is useful for comparing evaluation results.

The DARPA Sets for the learning consists of 7 weeks data. Each week has five days, and each day has the BSM audit data and the TCP dump data. It also provides TCP dump list file, which labels every flow whether the flow is attack or not. All attacks in DAPA Sets can be categorized into 4 classes of attacks. The classes are summarized as follows.

- $Denial of Service (DOS)$: Attack compromising the availability, e.g., syn flood attack.
- $Remote to Local (R2L)$: Attack trying unauthorized access from a remote machine, e.g., guessing password.
- $User to Root (U2R)$: Attack trying unauthorized access to local superuser (root) privileges, e.g., various buffer overflow attacks.
- $Scan$: Attack to surveille and gain information illegitimately, e.g., port scanning.

In this paper, we only consider the TCP dump datum and TCP dump list files because we target the NIDS. We employ the DARPA Sets to learn our decision trees and to evaluate our decision trees. The details how we use the DARPA Set are explained the next section.

III. PROCESS TO MAKE THE DECISION TREE

The making process of the decision tree can be simply represented as shown in Fig. 2. As you can see in the figure, the pre-processing procedure receives DARPA Training Set as the input, and manufactures the values of each features appropriately for the input data of the ID3 algorithm. The ID3 algorithm selects the features and makes the decision tree.

A. Classification of DARPA Set

The attacks can be classified into 4 classes in DARPA Training Set. Thus, we extract the TCP dump data for each attack in the whole DARPA Training Set, and we make 4 decision trees for each attack class. In this subsection, we explain how to extract the attack data from the DARPA Training Set.

The TCP dump list file, which is mentioned in previous section, contains the information that identifies each flow, and indicates whether the flow is an attack or not. The table I

1687	07/02/1998	8:16:57	0:00:01	snmp/u	161	1411	192.168.001.001	194.027.251.021	0	-
1688	07/02/1998	8:16:57	0:00:01	snmp/u	1411	161	194.027.251.021	192.168.001.001	0	-
1689	07/02/1998	8:16:59	0:05:28	ecr/i:r1215	-	-	202.077.162.040	172.016.114.050	1	smurf
1690	07/02/1998	8:16:59	0:05:28	ecr/i:r1206	-	-	202.077.162.178	172.016.114.050	1	smurf

TABLE II
EXAMPLE OF EXTRACTION

| tcpdump -r input.dump src host 202.77.162.40 and dst host 172.16.114.50 -w smurf1.dump |
| tcpdump -r input.dump src host 202.77.162.178 and dst host 172.16.114.50 -w smurf2.dump |

represents a part of the TCP dump list file. Every entries consists of the flow identifier number, date, time when the first packet of the flow is arrived, duration, service name, source port number, destination port number, source IP address, destination IP address, attack score, and the name of the attack. With this file, we are able to recognize which flow is an attack and to extract the data from the TCP dump data with the information in the TCP dump list file. The table II shows an example of the smurf attack extraction from the TCP dump file using the information in the table I. With this procedure, we extract all kinds of attacks from the TCP dump files of DARPA Training Set.

B. Pre-Processing

Now, we have TCP dump files for all kinds of attacks. However, these files are not ready to be an input of the ID3 algorithm. The TCP dump files have to be preprocessed to be the suitable data for the ID3 algorithm, because the ID3 algorithm cannot be handled the continuous value. Preprocessing is also helpful to summarize information from the TCP dump files. We do not use all information contained in TCP dump files, but we manufacture the raw packet data to make the information be meaningful. This is called "Selecting Features". The selected features has to well identify the characteristics of the packets. In this paper, the selected features are the attributes that are mainly used to detect attacks in the Snort [14]. We select the features that are 5 tuple, IP TOS, IP length, IP fragmentation, IP TTL, UDP length, TCP flag, TCP window size, TCP urgent pointer, ICMP type, ICMP code, Packets Per Second (PPS), and Bits Per Second (BPS).

With the features, the raw packet data is summarized. This is called "Encoding". The some of encoding rule is represented in table III. The encoding rule has to well identify the characteristics of header fields of a packet.

C. Learning Data

The learning data is the input of the ID3 algorithm, which is the mixing data with the encoded data by the procedure explained in the previous subsection. The learning data has to contain the data for the positive class and the negative class. The data for the positive class is the data of the targeted attack class. The data for the negative class can be any data except the targeted data. We compose the data for the negative class with the normal data which is not attack data, and the attack data

TABLE III
EXAMPLE OF ENCODING RULE

IP header	Encoding Rule	Encoding Code
TOS	$TOSfield == 0$	IP TOS = 1
	$TOSfield > 0$	IP TOS = 2
TTL	$TTLfield < 64$	IP TTL = 1
	$64 < TTLfield >= 128$	IP TTL = 2
	$128 < TTL >= 192$	IP TTL = 3
	$192 < TTL >= 255$	IP TTL = 4
TCP header	Encoding Rule	Encoding Code
Port #	$port == 80$	PORT = 1
	$port == 20$	PORT = 2
	$port == 21$	PORT = 3
	$port >= 49151$	PORT = 24
PPS	Encoding Rule	Encoding Code
PPS	$pps <= 20$	PPS = 1
	$pps < 100$	PPS = 2
	$pps < 300$	PPS = 3
	$pps >= 300$	PPS = 4

TABLE IV
EXAMPLE OF LEARNING DATA

| 1 1 2 3 1 1 3 23 5 0 1 1 0 0 0 0 0 1 1 yes |
| 1 1 2 3 1 1 3 23 5 0 1 1 0 0 0 0 0 2 1 yes |
| 1 1 2 2 3 1 6 0 0 0 0 0 0 0 0 0 0 1 1 yes |
| 1 1 2 3 1 1 3 23 5 0 1 1 0 0 0 0 0 2 2 yes |
| 1 1 2 3 1 1 3 5 23 0 1 1 0 0 0 0 0 2 2 no |
| 1 1 1 1 1 2 3 5 23 0 1 1 0 0 0 0 0 1 1 no |
| 1 1 1 3 1 1 3 23 22 0 1 1 0 0 0 0 0 2 1 no |
| 1 1 1 3 1 1 3 22 23 0 1 1 0 0 0 0 0 2 1 no |
| 1 1 2 3 1 1 3 23 5 0 3 1 0 0 0 0 0 1 1 no |

which is not included in the targeted attack class. For example, in case of composing the learning data for DoS attack, the learning data contains every encoded data of DoS attack such as ping of death (POD), smurf, land, teardrop, and etc. The learning data for DoS attack also contains the encoded normal data that is attack-free data, and the encoded non-DoS attack data that may be belonged in R2L, U2R, and Scan class. The table IV shows an small part of the learning data. As you can see in the table, every entry in the learning data has to have the value for each feature, and to indicate that the entry is classified into whether the positive class or the negative class. If the learning data for every attack classes, the decision trees for each attack class can be made by simply input the learning to ID3 algorithm.

D. Generated Tree

Due to the insufficient space of the paper, we cannot place the figure of the every decision tree. So, we enter only the decision tree for the U2R and R2L attack. The decision tree for the U2R attack is depicted in the figure 3, 4.

IV. EVALUATION AND DISCUSSION

In this paper, we generate the decision trees for each attack class using ID3 algorithm with the DARPA Training Set. We evaluate the decision trees in this section. The testing data set is the DARPA Testing Set. The DARPA Testing Set is same as the DARPA Training Set in form, only except that the DARPA Testing Set contains more kinds of attacks than the DARPA Training Set. Thus, the decision tree can be tested with new kinds of attacks, which means the decision tree can be tested as the anomaly detection method. The detection rates for each attack class are represented in next subsections.

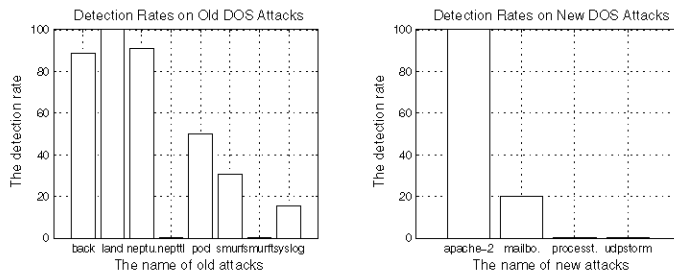## A. Detection rate for DoS Attacks

Fig. 5.   Detection rate of the detection tree for DoS attack

Between the old DoS attacks, back attack, land attack, and neptune attack have the detection rate of more than 90%, as you can see in the left graph of the figure 5. However, other attacks has the detection rate of less than 50%.

The pod attack, smurf attack, and smurfttl attack are the attacks mainly using ICMP. The packets using ICMP are included as smaller part in the DARPA Training Set than the DoS attacks using UDP or TCP. Thus, the information about the packets using ICMP cannot sufficiently influence the decision tree because the ID3, which generates the decision tree, adopts the concept of the information entropy.

Similarly, the information of the neptunettl attack is not enough to be reflected to the decision tree because most packets in neptunettl attack are destined to the telnet port, but the large number of packets destined to the telnet port are also contained in the data of negative class.

The right side of the figure 5 represents the detection rates of the new kinds of DoS attacks. As you can see in the graph, the apache-2 attack is detected for 100% detection rate in spite of new kinds of attack. That is because the patterns of the encoded data for apache-2 attack have similar patterns to the old DoS attacks. However the other attacks such as the mailbomb attack, process table attack, and UDP storm are rarely detected because the patterns of the encoded data are very different from the patterns of the old DoS attacks.
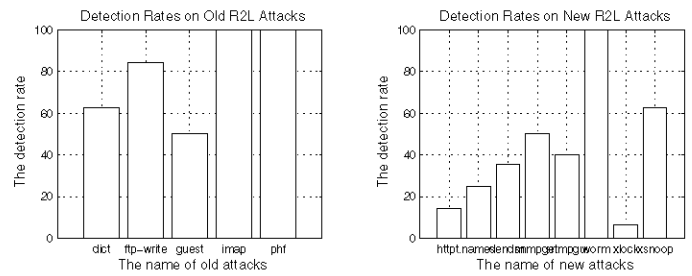
B. Detection rate for R2L Attacks

Fig. 6.   Detection rate of the detection tree for R2L attack

As you can see in the figure 6, the imap attack and the phf attack are detected for 100% detection rate. The ftp-write attack has also high detection rate, but the dict attack and the guest attack have near 50% detection rate. The encoded pattern of the dict and the guest attack are hardly characterized against the normal data. Thus, many features such as TCP port numbers, IP length, IP TTL, and TCP window size are examined to classify these attacks. However these attack are detected with less detection rate compared to the imap and the phf attack. To deal with this problem, more feathres are defined and examined than now, but this also leads to more complex decision tree and less effective performance.

For new kinds of R2L attack, attacks are not well detected except the worm attack and the xsnoop attacks. The worm attack and the xsnoop attack are well characterized with the features defined in this paper, while the others are not. The attacks with low detection rate has the encoded pattern similar to the U2R attack or Scan attack rather than R2L attack. To overcome, we need more features that can examine the contents of the packet.

C. Detection rate for U2R Attacks
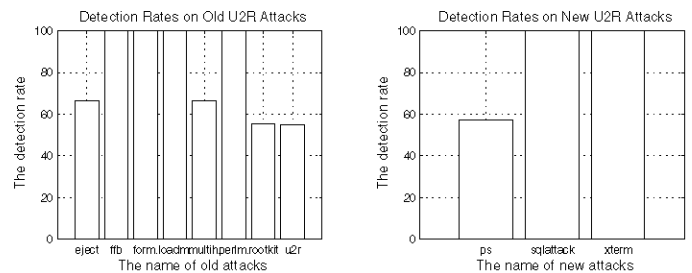
Fig. 7.   Detection rate of the detection tree for U2R attack

The ffb, format, loadmodule and perlmagic attacks are detected for 100%. The other attacks such as the eject, u2r, and rootkit attack has the detection rate of more than 50%. Even new kinds of attacks such as the sql attack, xterm attack , and ps attack are also detected well. The almost U2R attacks are judged with relatively few features such as TCP destination port number, TCP window size, and IP length. The

Fig. 3.   Decision Tree for U2R attack

Fig. 4.   Decision Tree for R2L attack

values of these features are similar pattern in U2R attacks. This is the reason the U2R attack are detected with the high detection rate. However this has the potential possibility that the packets, which is not in U2R attack class, can also be detected. For more precise detection, the features representing the characteristics of the contents in packets are also needed as well as the information in the packet headers.

D. Detection rate for Scan Attacks

Fig. 8.   Detection rate of the detection tree for Scan attack

For Scan attack, both old and new kinds of attacks are detected with relatively high detection rate. The data size of the scan attacks are bigger than the attacks included in the

TABLE V
PERFORMANCE COMPARISON

Category	Our Model		Comlumbia Model	
	Old	New	Old	New
DoS	82.6	60.0	79.9	24.3
R2L	78.1	24.2	60.6	5.9
U2R	58.6	58.8	75.0	81.8
Scan	75.5	81.4	97.0	96.7
Overall	77.6	55.5	80.2	37.7

other attack classes, so that the patterns of Scan attack class are more various than the others. This means that numerous patterns of data can be provided as the learning data set. This is the reason why Scan attacks can be detected with the high detection rate.

E. Discussion

We discuss the result for each attacks in the previous subsections. In this subsection, we compare the detection rate for each attack class, and compare the result to Columbia Data Mining Framework for IDS Model [15]. As you can see the table V, the detection rate of our model has higher detection rate than the Columbia model for new attacks. That means the our model achieves better performance for the anomaly detection.

V. Conclusion

In this paper, we generate the decision trees for DoS attack, R2L attack, U2R attack, and Scan attack. The ID3 algorithm is used as the learning algorithm to generate the decision tree automatically, and the DARPA Set is adopted for the training data. These method are widely used in the anomaly detection for the NIDS, but there are lack of description for the whole process making the decision tree. We describe the process generating the decision tree step by step, and the decision tree is evaluated by DARPA Set Testing Data. The proposed model achieves the improvement in detecting new kinds of attacks, the anomaly detection in other words. For the future research, the more detailed features, that are able to characterize the contents of the packets as well as the header information, should be defined for improvement the performance. Other data mining method such as neural network should also be researched in the further research.

Acknowledgment

References

[1] O. Depren, M. Topallar, E. Anarim, and M. Kemal Ciliz, "An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks", Expert Systems with Applications 29 (2005), pp. 713-722, May 2005.

[2] R. A. Kemmerer, and G. Vigna, "Intrusion detection: A brief history and overview", IEEE Security and Privacy Maganize (supplement to Computer, vol. 35, no. 4), pp. 27-30, April 2002.

[3] J. R. Quinlin, "Decision Trees and Decision Making", IEEE Transactions on System (Man and Cybernetic, vol. 20, no. 2), pp. 339-346, April 1990.

[4] Web page of ACM KDD Cup: http://www.sigkdd.org/kddcup/index.php, Accessed on November 2007.

[5] T. Abbes, A. Bouhoula, and M. Rusinowitch, "Protocol Analysis in Intrusion Detection Using Decision Tree", Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04, vol. 1), pp. 404-408, April 2004.

[6] C. Kruegel, and T. Toth, "Using Decision Trees to Improve Signature-Based Intrusion Detection", RAID 2003, LNCS 2820, pp. 173-191, Feburary 2004.

[7] V. H. Garcia, R. Monroy, and M. Quintana, "Web Attack Detection Using ID3", IFIP International Federation for Information Processing (vol. 218), pp. 323-332, October 2006.

[8] Web page of KDD Cup 1999 Data: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, Accessed on October 2007.

[9] Web page of MIT Lincoln Laboratory - DARPA Intrusion Detection Evaluation: http://www.ll.mit.edu/IST/ideval/index.html, Accessed on October 2007.

[10] K. Jearanaitanakij, "Classifying Continuous Data Set by ID3 Algorithm", 2005 Fifth International Conference on Information, Communications and Signal Processing, pp. 1048-1051, December 2005.

[11] S. Ruggieri, "Efficient C4.5" IEEE Transactions on Knowledge and Data Engineering (vol. 14, no. 2), pp. 438-444, April 2002.

[12] S. R. Safavian, and D. Landgrebe, "A survey of decision tree classifier methodology", IEEE Transactions on Systems, Man and Cybernetics (vol. 21, no. 3), pp. 660-674, June 1991.

[13] H. G. Kayacik, A. N. Zincir-Heywood, and M. I. Heywood, "Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets", Proceedings of the Third Annual Conference on Privacy, Security and Trust, October 2005.

[14] Web page of Snort: http://www.snort.org, Accessed on October 2007.

[15] W. Lee, S. J. Stolfo, and K. W. Mok, "A data mining framework for building intrusion detection models", Proceedings of the 1999 IEEE Symposium on Security and Privacy, May 1999.