Identifying Important Characteristics in the KDD99 Intrusion Detection Dataset by Feature Selection using a Hybrid Approach

Nelcileno Araújo[1]
[1]Institute of Computing
Federal University of Mato Grosso
Cuiabá, MT, Brazil
nelcileno@yahoo.com.br

Ruy de Oliveira[2]
Ed'Wilson Ferreira[4]
[2,4]Department of Informatics
Federal Institute of Mato Grosso
Cuiabá, MT, Brazil
ruy@cba.ifmt.edu.br
ed@cba.ifmt.edu.br

Ailton Akira Shinoda[3]
[3]Department of Electrical Engineering
State University Júlio de Mesquita Filho
Ilha Solteira, SP, Brazil
shinoda@dee.feis.unesp.br

Bharat Bhargava[5]
[5]Department of Computer Science
Purdue University
West Lafayette, IN, USA
bb@cs.purdue.edu

Abstract— **Intrusion detection datasets play a key role in fine tuning Intrusion Detection Systems (IDSs). Using such datasets one can distinguish between regular and anomalous behavior of a given node in the network. To build this dataset is not straightforward, though, as only the most significant features of the collected data for detecting the node's behavior should be considered. We propose in this paper a technique for selecting relevant features out of KDD99 using a hybrid approach toward an optimal subset of features. Unlike existing work that only detect attack or no attack conditions, our approach efficiently identifies which sort of attack each register in the dataset refers to. The evaluation results show that the optimized subset of features can improve performance of typical IDSs.**

Keywords: KDD99. Feature Selection, Hybrid Approach, K-Means, Information Gain Ratio

I. INTRODUCTION

Over the past ten years, the number of security related incidents registered at CERT.br (Center for Studies, Answers and Handling of Security related Incidents in Brazil) has increased about 100-fold [1]. This demonstrates the inherent vulnerability of the Internet, which calls for permanent development of efficient security mechanisms. As a result, various security tools, such as firewall, cryptography, and Intrusion Detection Systems (IDSs) have been developed rendering computing systems more reliable.

In particular, the IDSs have received great attention from researchers all over the globe because of their ability to keep track of the network behavior, so that abnormal behavior can be detected quickly. The detection can occur in two distinct ways. One technique uses previously known attack patterns to infer intrusions. This technique is normally called *misuse detection*. Another way of detection is called *anomaly detection*, in which there are no known attack patterns, but only regular patterns. Everything that is not regular is taken as anomalous and consequently may be linked to an intrusion [2].

Comparing the two IDS approaches, one can say that the *misuse detection* provides accurate results in recognizing patterns, but it is limited to the known attacks. This means new attacks that are not included in the signature database cannot be detected. On the other hand, the *anomaly detection* based approach provides good performance in detecting new forms of attacks, but gives high false positive rates (false alarms), due to the difficult of characterizing a practical normal behavior pattern for the nodes in the network. In fact, regardless of the IDS approach in place, for the sake of the reliability, it is needed to choose appropriate detection metrics to either represent the attack pattern efficiently or define the regular behavior expected for the network.

In order to choose proper intrusion detection metrics, several training datasets for IDSs have been created. One of the most popular such a dataset is the *Knowledge Discovery and Data Mining* – KDD99 [3], which was developed, by the *Massachusetts Institute of Technology* - MIT, during the international competition on data mining in 1999. In this dataset each connection (TCP Connection) is represented by 41 features, but experiments have shown that using all these features does not guarantee efficiency for attacks based on the package contents [4].

With that in mind, this paper proposes optimizing the existing metrics in the KDD99 training dataset through a feature selection technique using a hybrid approach, which will generate an optimal dataset of features. Differently from existing work, our approach takes into account all the categories of connections in KDD99 (attacks or no attacks), i.e., Normal, DoS, *Probe*, U2R, R2L. It is also a purpose of this paper to check the impact of using such a dataset on the IDS' accuracy.

The remainder of this paper is organized as follows. In Section 2, we describe the KDD99 intrusion detection dataset, where the procedures to generate the dataset are discussed. Section 3 addresses the selection of the most relevant features in KDD99 through a hybrid approach that combines the information gain ratio and the *k-means* classifier toward the optimized dataset. Yet in this section, it is shown comparative results on detection accuracy for ten distinct datasets generated from the so called "10%KDD99" training dataset. In Section 4, we conclude the work and outline suggestions for future work.

II. KDD99 INTRUSION DETECTION DATASET

This dataset is composed of various training and test data for IDSs. It was developed from a project at MIT Lincoln Labs, in 1999, where comparative evaluations among several distinct methodologies for intrusion detection were conducted. Fig. 1 illustrates the simulated network topology used for KDD99. It is a fictitious military network with three target-machines running various operating systems and services. Moreover, there are three additional machines for generating traffic from different sources. The Sniffer captures the network flow in TCP Dump format. The simulation ran for seven weeks.

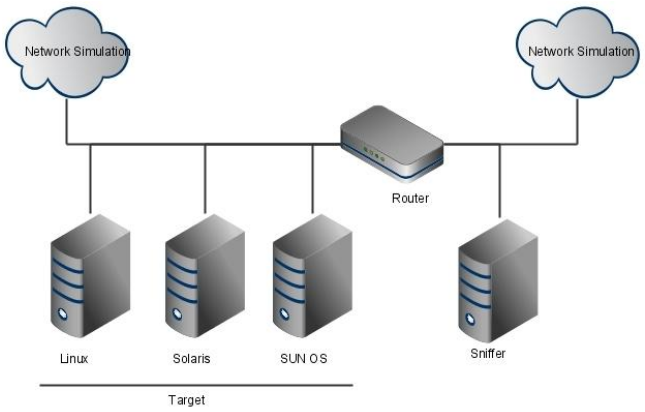


Figure 1. Topology of the simulated network for KDD99

The *logs* from the *sniffer* were divided into five categories [2], [3], [4]:

- **Normal** – connections that fit the expected profile in the military network.
- **Denial of Service** (DoS) – connections trying to prevent legitimate users from accessing the service in the target-machine.
- **Scanning** (*Probe*) – connections scanning a target machine for information about potential vulnerabilities.
- **Remote to Local** (R2L) – connections in which the attacker attempts to obtain non-authorized access into a machine or network.
- **User to Root** (U2R) –connection in which a target machine is already invaded, but the attacker attempts to gain access with superuser privilegies.

The files generated during the data collection were put in a standard format that contains 41 features for each registered connection. A connection here refers to a sequence of TCP packets with well defined time duration and transmitted over a well defined protocol between a source machine and a destination machine [3]. Each connection is labeled as either normal or under a specific sort of attack. Each connection register is about 100 bytes long.

The combination of the 41 features of each connection determines to which of the five connection categories mentioned above the audited connection belongs to. We call this procedure *categorization of the connections*. Accordingly, to better understand the contribution of each of the features within the dataset to this categorization, they were gathered into four groups [2], [3], [4], as follows:

- Basic features – identify the properties in the packet header, which represent critical metrics in a connection.
- Content features – these are information extracted from the packets that are only useful for experts who are able to associate them to known forms of attacks. Example of such metrics is the number of non-authorized access attempts into a given machine.
- Time based traffic features – show the features that occurred in a traffic profile computed during a time interval of two seconds. Crucial information related to some sort of attacks can only be obtained if the time duration is taken into consideration. A good example here is the number of connections to a single machine in a time interval of two seconds.
- Host based traffic features – In this case, the metrics, which show the traffic profile, are calculated from a historical data that is estimated from the last hundred used connections. A metric employed in this group is the number of connections to the same destination machine.

KDD99 is actually composed of three datasets. The largest one is called "Whole KDD", which contains about 4 million registers. This is the original dataset created out of the data collected by the *Sniffer*.

Since the amount of data to be processed is too high, it is interesting to reduce the computational costs involved as much as possible. Thus, a subset containing only 10% of the training data, taken randomly from the original dataset was created. This resulted in the "10% KDD" dataset used to train the IDS.

In addition to the "10% KDD" and "Whole KDD", there is a testing dataset known as "Corrected KDD". This dataset does not have the same distribution of probability of attacks as is the case in the other bases. This happens because the "Corrected KDD" includes 14 new types of attacks aiming at

checking the IDS performance to unknown forms of attacks. Note that in the complete dataset (Whole KDD) and in the training dataset (10% KDD) there are 22 types of attacks in total [4].

It is also important to mention that the KDD's training dataset contains a large number of connections for the categories normal, probe and DoS. They represent approximately 99.76% of the whole dataset.

III. Optimizing the KDD 99 Intrusion Detection Dataset Using a Hybrid Approach for Feature Selection

In general, it is not a good idea to feed the IDSs' learning mechanisms with the originally collected dataset. It needs to be optimized, since there are features that are either irrelevant or redundant for the learning algorithm. Without a proper treatment of the dataset, the detector accuracy is degraded and the test and training procedures may get really slow [5]. Hence, it is important to determine an optimal set of features that accurately represents the characteristics of the traffic being evaluated. Experiments have shown that proper set of features results in up to 50% of time reduction for the IDS' test and training phases [6].

A. Feature Selection

Feature selection is crucial for designing the intrusion detection models. In this process, only the most relevant features are extracted from the whole dataset. This prevents the irrelevant features from causing noise in the *categorization of the connections*.

Currently, there exist two main approaches to carry out feature selection: *filter* and *wrapper*. In the former, an independent metric, such as correlation and PCA [6], is used to compute the relevance of a set of features, resulting in the optimal subset of features that contains the important features classified in accordance with the measured values of the used metric. The latter uses machine learning algorithms for rating the importance of one or more features in order to build an optimal subset of features with the most representative features. *Wrapper* is more complex in terms of computing than the *filter* approach, but gives better results [6], [7], [8], [9].

These approaches have some drawbacks. For instance, the classifier input using random features can result in biased outcomes, and the search for the optimal set of features can result in thousands of combinations in the classifier, which leads too high computational costs. For example, the KDD99 dataset encompasses 41 features, and considering all possible combinations in the classifier to verify which set best contributed to the detection models, we will have hundreds of billions of feature combinations that can render the use of the dataset unviable.

Different techniques have been employed to mitigate the feature selection problem. In [10], the authors used classification algorithms to reduce the set of features out of the KDD99 dataset (originally with 41 parameters) into an optimal subset having 6 features only. They used the *Support Vector Machines (SVM)*, *Multivariate Adaptive Regression Splines* (MARS) and *Linear Genetic Program* (LGP) algorithms to associate a weight to each feature. The *Sequential Backward Search technique was employed* in [11] and [12] to indentify the subset of relevant features. In their approach the whole dataset is initially used and after each iteration a feature is removed from the dataset, until the desirable precision for the classifier is reached. Another popular approach, known as *hybrid approach*, combines the both techniques: *filter* and *wrappe*r. The work in [6] shows the efficiency of the hybrid approach with large datasets, in which the calculation demand of the optimal subset of features is similar to the one of the *filter* approach. In [5], the authors use the hybrid approach over a dataset obtained in an infrastructure wireless network based on the IEEE 802.11 model. They applied the Gain Information Ratio metric to classify the original dataset of features on the basis of the reached grade, and a so-called *k-means* classifier to build an optimal subset of features that increases the detector accuracy and at the same time reduces the learning time.

B. Proposed Model

Our proposed scheme for feature selection is based on the hybrid approach published in [5]. Nevertheless, while the work in [5] evaluates the quality of the optimal subset of features considering only whether the connection is either normal or under attack, our evaluation takes into account all the categories of connections in KDD99, i.e., Normal, DoS, *Probe*, U2R, R2L. Besides, the captured data in our evaluations were not collected in an infrastructure wireless network but in a wired military-like network. We also used the two metrics to evaluate the capability of detection of the IDS: the detection ratio of the whole dataset and the acknowledgment accuracy ratio of each connection category.

The feature selection algorithm proposed here is shown in Fig. 2. Initially, the information gain ratio for each of the 41 features of KDD99 are computed, and then ranked in accordance to their values. In the sequence, after each iteration, the k-*means* classifier extracts the feature with the highest IGR from the dataset and assesses the detection rate of the optimal subset of features. Additionally, the accuracy level in detecting the right category for the connection in the optimal subset is verified. The selecting process stops when either the classifier accuracy is above the adjusted threshold or the accuracy value is below the previous calculated value.

The IGR metric was used here mainly because of its good results shown in the *filter* approach, as well as its low computational cost [4], [5], [13]. This metric is computed as shown in (1) [14].

$$IGR(D, A) = \frac{Gain(D, A)}{SplitInformation(D, A)} \quad (1)$$

where,

D – training data with N features.

A – set of features in the dataset.

Algorithm **Feature Selection based on IGR/K-means**
Input:
D – Training data with *N* features
IGR – Information Gain Ratio
C – k-*means* Classifier
AC – Current Accuracy
AP – Previous Accuracy
Threshold – Gain accuracy threshold
Output:
$S_{optimum}$ – Optimal subset of features
Begin
//*Filter Approach*
For each feature *f* compute IGR(*f*)
Classify the features in D based on IGR(*f*)
//*Wrapper Approach*
Initialize S = EMPTY and AC = 0
Repeat
AP = AC
f = getNext(D)
$S_{optimum} = S_{optimum}$ U {f}
D = D – {f}
AC = ACCURACY(C,$S_{OPTIMUM}$)
UNTIL (AC-AP) < threshold or AC<AP
End

Figure 2. Feature Selection Algorithm based on IGR/k-means

The information gain ratio is a quantitative measure used to grade the relevance of the features based on the values of such features in the dataset [15]. Nonetheless, before computing the information gain ratio, it is necessary to check the noise (misclassification) inserted in the training set. This checking is called Entropy and is computed using (2).

$$Entropy(P) = -\sum_{i=1}^{n} p_i \log_2 p_i \quad (2)$$

where,

P_i – probability of a given feature (or attribute) value to be in the sampled set of the dataset.

n – maximum value assigned to a feature.

After computing the entropy of D, the formula for the information gain ratio in (3) is used to determine the best feature to be used as root.

$$Gain(D, A) = Entropy(D) - \sum_{v \in Attribute(A)} \frac{|D_v|}{|D|} Entropy(D_v) \quad (3)$$

where,

D_v – amount of samples of the dataset that contain repetitions of the evaluated feature.

D – total samples of the training dataset.

The Entropy gives us information about the probability of a given feature value to be in a dataset (p_i). The split information represents the potential information to be generated by dividing the base D into *m* subsets, as defined in (4).

$$Split\ \inf(P) = -\sum_{v \in Attributes(A)} \frac{|Dv|}{|D|} \log_2 \frac{|D_v|}{|D|} \quad (4)$$

The *K-Means* algorithm [16] is one of the oldest and more important algorithms available in the literature for performing grouping. Although it has been published over forty years ago, it is still largely used these days. The main reasons for this popularity include its simplicity and high performance. *K-Means* complexity is *O(nK)*, being *n* the cardinality of original dataset and *K* is the amount of groups [9]. Besides, *K-Means* is of ease implementation and has been evaluated quite a lot in recente years, which leveraged the development of various novelties in the way it works. Because of these characteristics, noting that *K-Means* performance over similar tools is much better, we have adopted it in our scheme.

C. Experimental Evaluations

In order to evaluate the efficiency of the hybrid approach, using IGK/K-means, on optimizing the KDD99, we used for the experiments the parameters setup shown in Table I. The subset "10% KDD99" was chosen because it was created exactly to be used in training IDS'learning modules [3]. This subset is composed of approximately 490.000 samples including all kinds of connection categories defined in KDD99 (Normal, DoS, Probe, U2R, R2L).

The feature selection was carried out by the data mining tool called WEKA [17]. This tool performed efficiently in related work such as [5], [6], [10], [11], [12], [13], and so we adopted it here as well.

TABLE I. PARAMETERS SETUP FOR THE EXPERIMENTS

Components	Configuration
Dataset	10% KDD99
Programming tools	WEKA, MS-Excel 2007
Computer	Notebook processor Intel Celeron M 440, 1.86GHz, 2GB RAM, 250 GB of hard disc
Operating system	Microsoft Windows XP Professional (SP2)

Regarding the evaluated scenarios, two distinct scenarios were considered. The first one was used to optimize the KDD dataset and the second one to check the optimization's effects on the performance of an IDS, as follows:

- Scenario 1: the IGR was applied to the dataset "10%KDD99" to measure the relevance of the 41 features, resulting in a sorted classification. Then, the K-*means* classifier is used to compute the optimal subset of features;
- Scenario 2: the dataset "10%KDD99" was divided into ten subsets, containing about 49000 registers of connections each. Subsequently, each subset was processed by an IDS based on the "decision trees" algorithm, and for this, the optimal subset of features was used.

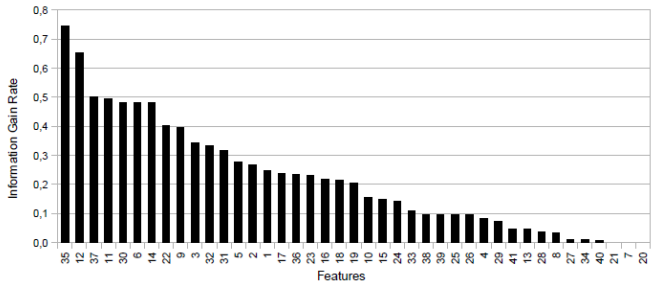


Figure 3. Descending classification of the IGR for the features in the dataset "10%KDD99".

In both scenarios, the validation of the results was conducted with the so-called "10-fold-cross" technique [18]. The idea here was to obtain low error rates and find out the intrusion detection rate.

1) Results for the Scenario 1

Fig. 3 shows the classification of the 41 features of the dataset "10%KDD99" sorted in a descending order through the information gain ratio. Most of the features have IGR under the average of the dataset, ($IGR_{average} = 0{,}22$). In fact, only 18 features are above the average. This shows that the original database has data concentration in a small group of values. Features that result in a convergence of connection categories within a small group of values are little significant to describe a node behavior. This indicates that the original dataset may contain irrelevant data for the IDS and so needs to be optimized.

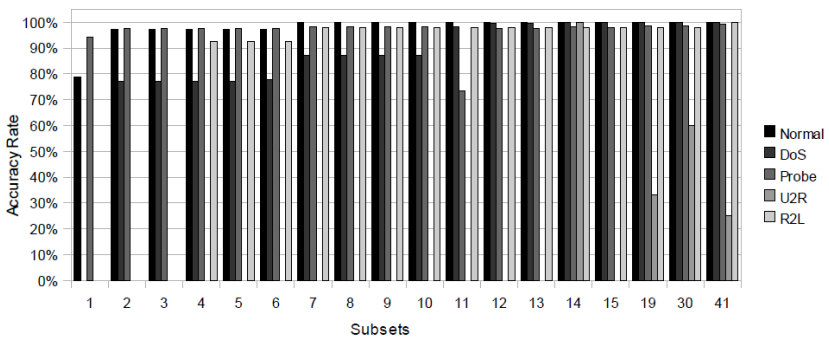
After obtaining the ranked set of features through the IFG the optimal subset of features were determined by the k-*means* classifier. After each iteration of the classifier the most relevant feature, in accordance to the IGR, was added to the optimal subset of features. The classifier keeps track of the accuracy rate of the connection categories in the new subset, and once either the accuracy reaches 90% or it is lower than the value calculated in the previous iteration the classification process ends and the optimal subset of features is determined.

In Fig. 4(a) one can notice that the best results are obtained when the optimal subset of features has the 14 most important features of the evaluated dataset. With less features than that, the U2R class has accuracy close to zero, which means that despite the high detection rate depicted in Fig 4(b), the algorithm does not provide enough accuracy in recognizing U2R connections. Hence, whenever the optimal subset of features contains categories of connections with large percentage of samples, the detection rate is not a good criterion to use to evaluate the quality of such a subset. For this evaluated dataset, the DoS category accounts to 80% of the whole sampled connections. The optimal subset of features "10%KDD99" comprises the following features: *dst host diff srv rate*, *logged in*, *dst host srv diff host rate*, *diff srv rate*, *destination bytes*, *root Shell*, *is guest login*, *urgent*, *service*, *dst host count*, *srv diff host rate*, *source bytes* e *protocol type*.

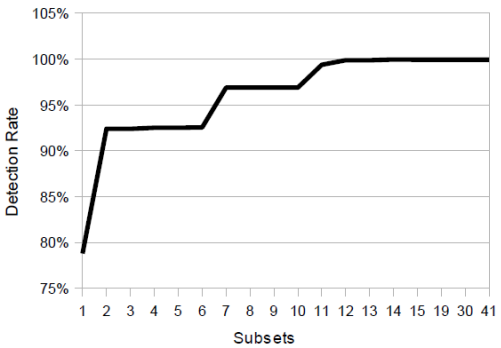
2) Results for the Scenario 2

The purpose of the second scenario is to provide us with good insights into the effects of an optimized dataset on the performance of an IDS. The dataset "10%KDD99" was divided into 10 subsets, as depicted in Fig. 5. Each subset has its own distribution of categories of connections but the subsets 5 and 6. It is possible to distinguish a pattern in most subsets, since there are a lot more DoS connections registers than registers of the other connections. This can be interesting to evaluate our previous statement that subsets with a strong prevalence of a single connection category might render the adjusted detection rate unfeasible.

Subsequently, we used the features inside each generated optimal subset of features to feed an IDS based on a decision trees algorithm. The outcome is shown in Fig. 6 through three parameters: detection rate, accuracy rate and true positive rate. The false positive parameter was ignored because the assumed values are too close to zero, which does not contribute to the evaluation of the quality of the optimal subset of features.



(a) Accuracy rate



(b) Detection rate

Figure 4. Performance of alternate subsets (optimal subsets) of "10%KDD99" by two stop criteria.

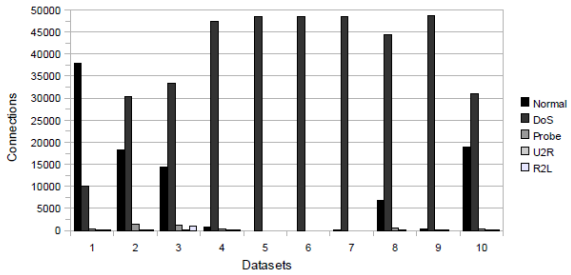


Figure 5. Composition of datasets generated from the "10%KDD99".

From the results we find out more about the quality of the optimal subset of features in terms of connection categories detection. As shown in Fig. 6, the detection rate for all subsets surpasses 99%. To ensure reliability in our evaluations, we also included the accuracy rate in our evaluations. Fig.6 shows that all connections categories provided high accuracy (over 90%) except the U2R category that in the best scenario gave 60% of accuracy. This is a result of the low relevance of such a category in the sample space of each subset, which corroborates our finding that the detection rate parameter does not impact the evaluation process of the quality of the optimal subset of features.

Finally, the high values for the true positive rate strenghtens the viability of the our proposal, as this indicates that the IDS is capable of recognizing the connection categories efficiently. It is important to note that in some cases the rate is below 80%, which occurred again due to the low impact of the evaluated category on the sample space of the dataset. As an example, we have in the category 7 a total of 11 connection of the category Normal but none of them is recognized by the IDS.

IV. Conclusions and Outlook

We have proposed the use of a hybrid approach to select the best features from the training dataset KDD99 toward a reduced dataset to improve an IDS efficiency. The hybrid approach combines the information gain ratio (IGR) and the k-*means* classifier. The former is responsible for classifying the features on the basis of IGR measure. The latter generates an optimal subset of features by evaluating the features accuracy from the ranked data provided by the IGR.

The evaluation results suggest that the detection rate on its own does not provide reliability in detecting intrusions. The main reason lies in the differences found in the weight that each category of connection in the training dataset has on the proposed mechanism. Categories with low weight face problems of detection despite the detection rate remains high, which occurs due to the "Giant" categories of connections in place.

To address this problem, we propose here using jointly the detection rate and the accuracy rate. By using the dataset features in a fairer way, without favoring any category, the accuracy rate corrects the distortions caused by the "Giant" categories of connections.

Since the computational cost for large dataset are non-negligible, and the results here showed that the optimized dataset provided similar outcome to the original dataset (with 41 features), we can say that our proposal is indeed worthwhile. By using it, an IDS will be trained much faster than it would do with the original dataset.

The following tasks are left for future work. Application of the technique used here for feature selection in dataset collected from other network environments such as sensor, mesh, and WiMax wireless networks. Alternate programming tools, such as C and FORTRAN, for conducting the feature selection, as the WEKA [7] algorithm, that was used here for feature selection, is based on JAVA and so demanded too much both memory and processing capabilities of the machine used in the experiments. And finally, the use of Metaheuristics (genetic algorithms, tabu search, and simulated annoling) to perform feature selection through the computation of the optimal subset of features.

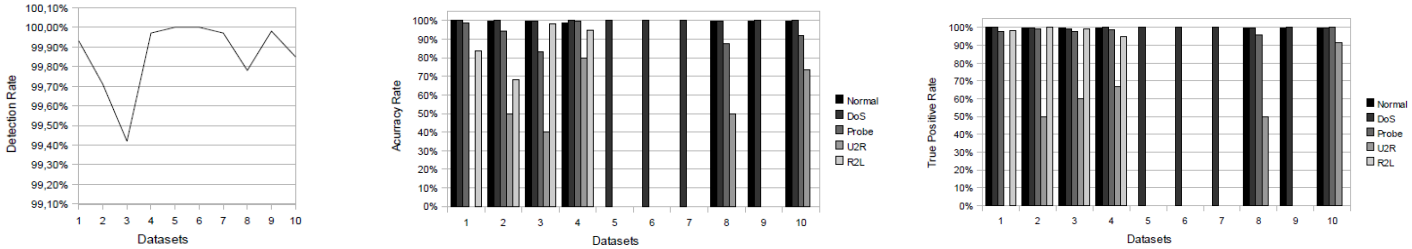


Figure 6. Results obtained by the decision tree based IDS on the 10 datasets generated from the "10% KDD99".

ACKNOWLEDGMENT

This material is based on a research project funded by the Foundation for Research Support of Mato Grosso (FAPEMAT) on the supervision of the Network and Security Research Group (GPRS). GPRS is managed by the Federal Institute of Mato Grosso (IFMT) in conjunction with the Federal University of Mato Grosso (UFMT), State University Júlio de Mesquita Filho (UNESP) and Federal University of Uberlandia (UFU). The authors acknowledge the facilities and equipment provided by IFMT for the development of this work.

REFERENCES

[1] CERT.br – Computer Emergency Response Team Brazil. http://www.cert.br/stats/incidentes/, Last Access: August 2009.

[2] P. Souza, Study about anomaly based intrusion detection systems: an approach using neural networks, M.Sc. Thesis, Salvador University/Salvador, 2008.

[3] R. Lippmann, J. W. Haines, D. J. Fried, J. Korba & K. Das, "The 1999 DARPA off-line intrusion detection evaluation," Computer Networks, vol.34, n.4, pp. 579-595, 2000.

[4] H. G. Kayacik, A. N. Zincir-Heywood & M. I. Heywood, "Selecting features for intrusion detection: a feature relevance analysis on KDD 99," in Proceeding of third annual conference on privacy, security and trust, 2005.

[5] M. Guennoun, A. Lbekkouri & K. El-Khatib, "Optimizing the feature set of wireless intrusion detection systems," International Journal of Computer Science and Network Security, vol. 8, n. 10, pp. 127-131, 2008.

[6] Y. Chen, Y. Li, X. Cheng & L. Guo, "Survey and taxonomy of feature selection algorithms in intrusion detection system," Lecture Notes in Computer Science, vol. 4318, pp. 153-167, 2006.

[7] H. Liu & H. Motoda, Feature selection for knowledge discovery and data mining, Kluwer Academic, 1998.

[8] R. A. M. Horta & F. J. dos S. Alves, "Data mining techniques in feature selection for prediction of insolvency: implementation and evaluation using recent brazilian dataset," in Proceedings of the XXXII Meeting of ANPAD, 2008, pp. 1-15. [Digests XXXII Encontro da ANPAD, 2008, p. 152.]

[9] J. de A. Soares, Preprocessing data in data mining: a comparative study in inputation, D.Sc. Thesis, Federal University Rio de Janeiro/Rio de Janeiro, 2007.

[10] H. Sung & S. Mukkamala, "The feature selection and intrusion detection problems," in Proceedings of the 9th Asian Computing Science Conference, Lecture Notes in Computer Science, 2004, vol. 3321, pp. 468-482.

[11] H. Sung & S. Mukkamala, "Identifying Important Features for Intrusion Detection Using Support Vector Machines and Neural Networks," in Proceedings of the 2003 Symposium on Applications and the internet, 2003, pp. 209-217.

[12] G. Stein, B. Chen, A. S. Wu & K. A. Hua, "Decision tree classifier for network intrusion detection with GA-based feature selection," in Proceedings of the 43rd Annual Southeast Regional Conference, 2005, vol. 2, pp. 136-141.

[13] Bsila, S. Gombault. & A. Belghith, "Improving traffic transformation to detect novel attacks," in Proceeding of 4th International Conference: Sciences of Eletronic, Technologies of Information and Telecommunications, 2007.

[14] O. Maimom & L. Rokach., Decomposition methodology for knowledge discovery and data mining – theory and applications, World Scientific Publishing Co, 2005.

[15] T. M. Mitchell, Machine Learning, McGraw Hill, 1997.

[16] J. Mcqueen, "Some methods for classification and analysis of multivariate observations," in Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1967, pp. 281–297.

[17] R. R. Bouckaert et al., WEKA manual for version 3-7-0. http://www.cs.waikato.ac.nz/ml/weka/, Last Access: August 2009.

[18] Y. Bengio & Y. Grandvalet, "No unbiased estimator of the variance of *k*-fold cross validation," *Journal of Machine Learning Research*, vol.5, pp. 1089-1105, 2004.