Detection & Classification of Internet Intrusion Based on the Combination of Random Forest and Naïve Bayes

Younes Chihab [#1], Abdelah Ait Ouhman [#2], Mohammed Erritali [*3], Bouabid El Ouahidi [*4]
[#] Faculty of Sciences Semlalia , Cadi Ayyad University, Marrakech, Morocco
[1] chihabyounes@gmail.com
[2] Ouhman@uca.ma
[*] Faculty of Sciences, Mohamed V Agdal University, Rabat, Morocco
[3] mederritali@yahoo.fr
[4] ouahidi@fsr.ac.ma

***Abstract*— The Use of internet renders a network packets susceptible to attacks ranging from passive eavesdropping to active impersonation, message replay and message distortion. There is no clear description as to what packets can be considered normal or abnormal. If the intrusions are not detected at the appropriate level, the loss of system may sometimes be unimaginable. Although many intrusion detection system (IDS) methods are used to detect the existing types of attacks within the network infrastructures, reducing false negative and false positive is still a major issue. In this work we present a comparative study between five data mining algorithms to come up finally with the proposition of a hybrid classifier based on Random Forest and Naïve Bayes algorithm. This method provides an effective distinction between different types of intrusions which allows us to customize the treatment given to each type of intrusion. These methods are tested using the KDD'99 database.**

Keywords -- IDS, Data Mining, Random Forest, Naïve Bayes, KDD Cup, Network Security.

I. INTRODUCTION

The increase in network interconnections made them accessible by a different population of users which is increasing, these users are not all full of good intentions vis-a-vis these networks. Indeed they may try to access, read, modify or destroy sensitive information or simply to undermine the proper functioning of the networks. As soon as these networks have emerged as potential targets of attacks, securing them has become a circumvented issue.

Intrusion detection is a mechanism that allows discovering or identifying the use of a system for purposes other than intended.

There are currently two main approaches used to develop systems for intrusion detection: the scenario approach [1] and the behavioral approach [2]. The behavioral approach assumes that normal activity is different from an intrusive activity.

It is sufficient to develop a profile of normal activity and a mechanism that allows comparing the current activity to developed profile to detect significant differences that will be considered as a possibility of intrusions.

Such profile can be obtained by observing, for a sufficient time, a normal activity within a network. A behavioral intrusion detection system uses artificial intelligence methods to develop this normal profile.

In this work we present a comparative study between five datamining algorithms to come up finaly with the proposition of a hypride classifier based on Random Forest and Naïf Bayes algorithm. These methods are tested using the KDD'99 database.

II. DATA MINING ALGORITHM

In this article we operate on five data mining algorithm. Each one of these algorithms has its own characteristic that can be explored in intrusion detection and classification:

- ID3
- C4.5
- Rnd Tree
- Multilayer perceptron
- Naive Bayes continuous

A. The ID3 Algorithm

ID3 is a supervised algorithm [3] developed by Ross Quinlan whose purpose is to build decision trees from a data set. Decision trees are very efficient as they classify new cases from the training data and test data to properly assess the quality of the tree constructed.

The decision tree is built recursively. The ID3 calculates, among the remaining attributes, the ones which will generate the most information (information gain), which will classify examples of any level of the decision tree.

B. The C4.5 Algorithm

Among the disadvantages of the ID3 algorithm that may be mentioned is that it is unable to process discrete attributes which is the case in our paper, and it is also incapable of solving problems related to missing attributes. The algorithm C4.5 developed by Quinlan [4],[5] is essentially based on the Functioning of the ID3 algorithm by providing improvements [6] including the calculated gain information

The C4.5 uses the expansion phase to calculate the correct decision tree recursively dividing the training set by using the entropy function. During this phase, first we will be interested mainly in the decision, whether a node is terminal or not, then the selection of test to associate with a node by calculating the test that maximizes the amount of information gain and finally the assigning the majority class to a leaf.

To prune the constructed tree, the C4.5 uses the training set; the pruning criterion is based on a heuristic to estimate the actual error on a given sub-tree.

C. Random Forest

A random forest is an ensemble of decision trees which will output a prediction value. That operates by constructing a multitude of decision trees at training time and outputting.

Random forests as defined by Leo Breiman [7]: Is a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. Each decision tree is constructed by using a random subset of the training data.

D. Naive Bayes

One of the strengths of this technique is that it requires a small amount of information during the learning phase. Naive Bayes algorithms, based on Bayes' theorem [8], are widely used in the classification domains. The operating principle of this algorithm is based on the assumption that each class of examples is independent [9]. It allows to estimate the probability of each class among the examples and then sets this example as the class the most likely [10].

E. Multilayer perceptron

A Multilayer perceptron behaves, outside a perspective, as a function f that processes data (inputs) and produces a corresponding response (output).

It is assumed that learning is achieved and the weights are fixed. The neurons perform a simple weighted sum of inputs, compare a threshold value, and provide a response at the output. For example, we can interpret his decision as Class 1 if the value of x is 1 and class 2 if the value of x is -1 [11].

Multilayer perceptron uses neurons bearing the sigmoid activation function that allows the nuances needed for proper use of back-propagation learning algorithm.

III. THE KDD CUP DATASET

The dataset [KDD] contains 494021 records of different kinds of intrusions. There are 4 main categories of attacks in the KDD dataset (Fig.1):

- Denial-of-service attack: occupied computer resources
- Probing: scans for potential vulnerabilities in the network.
- User to Root Attacks: access to a normal user account.
- Remote to User Attacks: test potential vulnerabilities of this system.

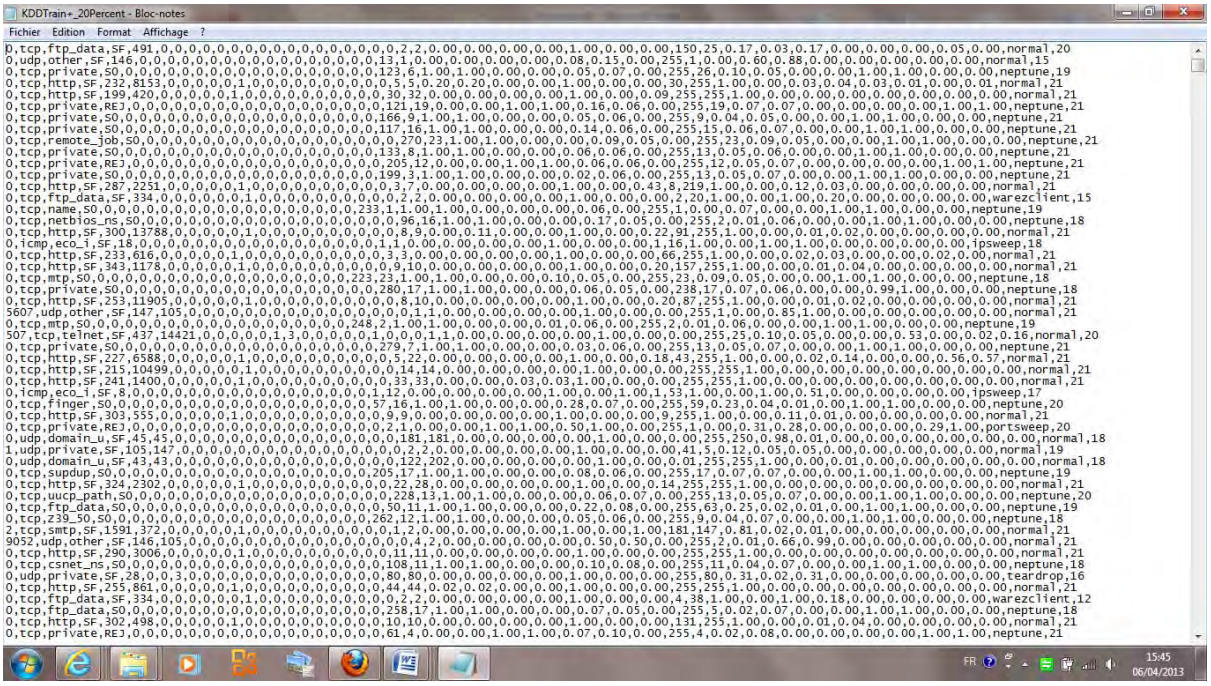

Fig.1: An overview of the KDD dataset

The objective was to survey and evaluate research in intrusion detection. A standard set of data was proposed, which includes a wide variety of intrusions simulated in a military network environment. The raw training data was about four gigabytes of compressed binary TCP dump data from seven weeks of network traffic.

IV. Comparative Study Between Data Mining Algorithms

In this part we present a comparative study between the presented algorithms applied to predict intrusion from the KDD dataset.

The experimental steps can be summarized as follows: First, the raw data samples are transformed into Tables recognized by Tanagra. Second, the converted Tables are presented separately to each one of the presented algorithms to achieve the training & the classification process.

A. ID3 Results

For the ID3 parameters, size before split was 200, after split 50, max depth of leaves: 10 & goodness of split threshold: 0.03. Table I, presents prediction rate obtained by ID3 algorithm.

TABLE I
Prediction rate obtained by ID3 algorithm on KDD data set.

Prediction Values		
Value	**Recall**	**Precision**
normal.	0.9997	0.0054
buffer_overflow.	0	1
loadmodule.	0	1
perl.	0	1
neptune.	0.9912	0.0037
smurf.	1	0
guess_passwd.	0	1
pod.	0	1
teardrop.	1	0.2143
portsweep.	0.45	0
ipsweep.	0.9605	0.0094
land.	0	1
ftp_write.	0	1
back.	1	0
imap.	0	1
satan.	0	1
phf.	0	1
nmap.	0.7615	0
multihop.	0	1

The produced decision tree is composed of 79 nodes and 73 leaves (Computation time: 3682 ms). It's clear that ID3 cannot detect a big number of intrusions but we can notice that from 39297 normal tread only 2 are detected intrusions.

B. C4.5 Results

For the C4.5 parameters, we use a min size of leaves=5 and a Confidence-level for pessimistic=0.25. Table II gives the experimental results obtained for the C4.5 algorithms

TABLE III
Prediction rate obtained by C4.5 algorithm on KDD data set.

Prediction Values		
Value	Recall	1-Precision
normal.	0.9998	0.0005
buffer_overflow.	1	0.5
loadmodule.	0	1
perl.	0	1
neptune.	1	0
smurf.	1	0
guess_passwd.	0.9245	0
pod.	1	0
teardrop.	1	0
portsweep.	1	0
ipsweep.	0.9985	0.0076
land.	0	1
ftp_write.	0	1
back.	1	0
imap.	0	1
satan.	0	1
phf.	0	1
nmap.	0.9615	0.0157
multihop.	0.4	0.6

The produced decision tree is composed of 201 nodes and 159 leaves. (Computation time : 3682 ms.)

The C4.5 presents very good classification rate, more than 99% precision for more than 99% of the data set. But the problem still persists with (loadmodule, perl, land. , ftp_write., imap., satan. & phf.) intrusions.

C. Random Forest results

The Rnd tree or Random Forest presents the best performance from all decision tree categories with recognition rate of 99.99%, with a computation time of 1872ms. (We can only note a 60% recognition rate for the multihop. And 75% for the ftp_write.

TABLE IIIII
Prediction rate obtained by C. Random Forest algorithm on KDD data set.

Prediction Values		
Value	Recall	1-Precision
normal.	0.9995	0.0002
buffer_overflow.	1	0.4
loadmodule.	0	1
perl.	1	0
neptune.	0.9999	0
smurf.	1	0.0011
guess_passwd.	1	0
pod.	1	0
teardrop.	1	0
portsweep.	1	0
ipsweep.	0.9954	0.0076
land.	1	0
ftp_write.	0.75	0.1429
back.	1	0
imap.	1	0
Satan.	1	0
phf.	1	0
nmap.	1	0.0076
multihop.	0.6	0

D. Naive Bayes Results

Parameters used for Naïve bayes training are: Lambda for laplacian=0 & Homoscedasticity assumption=1;

TABLE IVV
Prediction rate obtained by NAIVE BAYES algorithm on KDD data set.

Prediction Values		
Value	*Recall*	*Precision*
normal.	*0.9803*	*0.0059*
buffer_overflow.	*0.6667*	*0.5*
loadmodule.	*1*	*0*
perl.	*1*	*0.9167*
neptune.	*0.9998*	*0.0006*
smurf.	*0.9963*	*0.0043*
guess_passwd.	*0.9623*	*0.0893*
pod.	*1*	*0.0476*
teardrop.	*0.9899*	*0*
portsweep.	*0.475*	*0*
ipsweep.	*0.9605*	*0.304*
land.	*1*	*0.8889*
ftp_write.	*0.25*	*0*
back.	*0.9361*	*0.0079*
imap.	*1*	*0.5*
satan.	*1*	*0.9918*
phf.	*1*	*0.9286*
nmap.	*0.7923*	*0.055*
multihop.	*0.8*	*0.9767*

The Naive Bayes continuous algorithm present a very good prediction rate by detecting the majority of intrusion but there are some problems with buffer_overflow , portsweep and ftp_writ intrusion the prediction rate is still under 70%.

E. Multilayer perceptron (MP)

In the experimental part we will use an MP with a single hidden layer consisting of ten neurons. Table V shows the obtained results. The MP is Unable to detect ten kinds of intrusions.

TABLE V
Prediction rate obtained by NAIVE BAYES algorithm on KDD data set.

Prediction Values		
Value	*Recall*	*1-Precision*
normal.	*0.9979*	*0.0028*
buffer_overflow.	*0*	*1*
loadmodule.	*0*	*1*
perl.	*0*	*1*
neptune.	*1*	*0.0015*
smurf.	*0.9996*	*0.0001*
guess_passwd.	*0*	*1*
pod.	*0*	*1*
teardrop.	*1*	*0.1681*
portsweep.	*0*	*1*
ipsweep.	*0.9574*	*0.0187*
land.	*0*	*1*
ftp_write.	*0*	*1*
back.	*0.996*	*0.0599*
imap.	*0*	*1*
satan.	*0*	*1*
phf.	*0*	*1*
nmap.	*0.7615*	*0.0198*
multihop.	*0*	*1*

V. THE PROPOSED SYSTEM

The proposed system is composed of the combination of two algorithms Random Forest and Naïve Bayes (Fig.2). We chose these two algorithms for two reasons: First, they showed the best recognition rate. Second, they present the fast execution times of the two algorithms (ten times less than the time required by the Multilayer Perceptron).

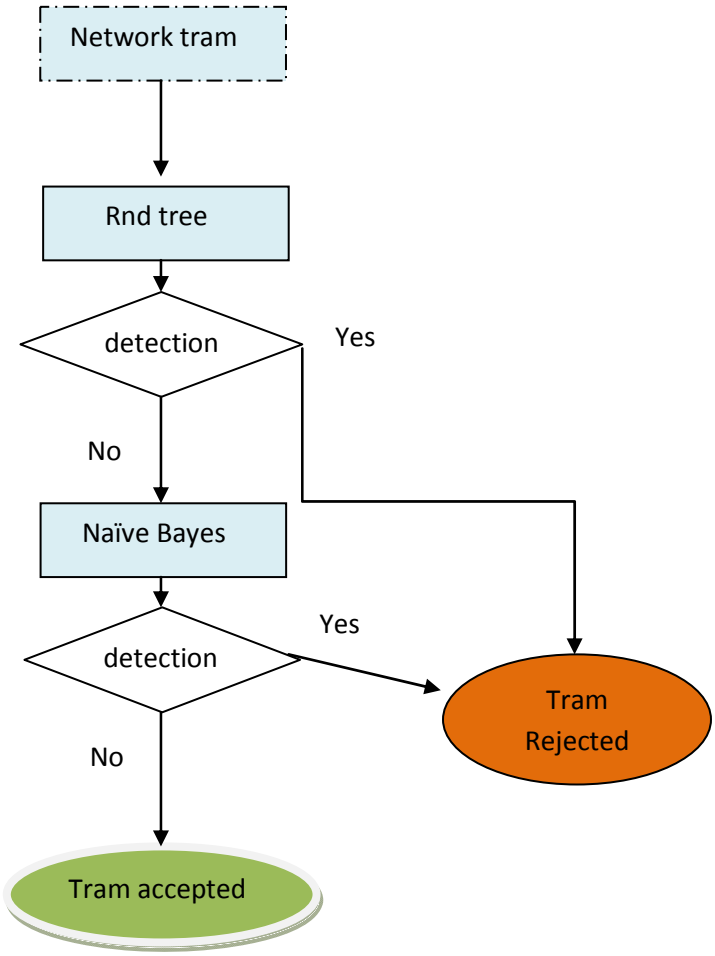

Fig.2. The proposed IDS, based on the combination of Random Forest and Naïve Bayes

TABLE VI
Prediction rate obtained by NAIVE BAYES, Random Forest & the proposed system.

Type of intrusion	Prediction Values		
	(Bayes Naive)	Random Forest	Proposed System
normal.	0.9803	0.9995	0.9999
buffer_overflow.	0.6667	1	1
loadmodule.	1	0	1
perl.	1	1	1
neptune.	0.9998	0.9999	0.9999
smurf.	0.9963	1	1
guess_passwd.	0.9623	1	1
pod.	1	1	1
teardrop.	0.9899	1	1
portsweep.	0.475	1	1
ipsweep.	0.9605	0.9954	0.9998
land.	1	1	1
ftp_write.	0.25	0.75	0.8125
back.	0.9361	1	1
imap.	1	1	1
satan.	1	1	1
phf.	1	1	1
nmap.	0.7923	1	1
multihop.	0.8	0.6	0.92

The proposed system has allowed us to obtain a remarkable improvement in recognitions rates. It allows a concrete detection of many intrusions that escaped from the majority of classifiers. Also it allowed the detection of new intrusions (not present in the training phaseas like peurl. & loadmodule.). Finally it permits the improvement of predictions rate for the entire data set presented.

VI. CONCLUSION

In this paper we have presented a comparative study between five datamining algorithms (ID3, C4.5, Random Forest, Multilayer Perceptron, Naive Bayes) applied to the classification of network intrusions. The Algorithms that have shown the highest prediction rate are the Random Forest and Naive Bayes Algorithms.

It has been noted experimentally that the complementarily of the two algorithms, Random Forest arrives to detect intrusions that escapes Naive Bayes and vice versa. For this we have proposed a hybrid system composed of two algorithms. As expected, the proposed system has led to a remarkable improvement in prediction with a reduced calculation time.

REFERENCES

[1] Faroq Anjum,Dhanant Subhadrabandhu and Saswati Sarkar, “Signature Intrusion Detectio for Wireless Ad Hoc Networks: A Comparative study of various routing protocols”, in 2003.
[2] P C Kishore Raja, Dr.Suganthi.M, R.Sunder, “WIRELESS NODE BEHAVIOR BASED INTRUSION DETECTION USING GENETIC ALGORITHM”, Ubiquitous Computing and Communication Journal, 2006.
[3] J. Ross Quinlan, Machine Leaming, 1986, « Induction of decision trees », p. 81-106.
[4] Comparative Analysis of Serial Decision Tree Classification Algorithms
[5] Surbhi Hardikar et al “Comparison between ID3 and C4.5 in Contrast to IDS “ / VSRD International Journal of CS & IT Vol. 2 (7), 2012
[6] NRSA, Méthodologie projet décisionnel. "Les arbres de décision.".
[7] LeoBreiman,AdeleCuttler,RandomTrees,http://www.stat.berkeley.edu/users/breiman/RandomForests/
[8] Nakache, D. (2007). Extraction automatique des diagnostics à partir des comptes rendus médicaux textuels. Laboratoire CEDRIC - équipe ISID. Paris, Conservatoire National des Arts et Métiers: 219.
[9] Han, J., M. Kamber Concepts d'exploration de données et des techniques [M] X. Meng, Pékin: Appuyez sur l'industrie mécanique, 2005 196 201

[10] H. Debar, M. Becker, and D. Siboni, "A neural network component for an intrusion detection system," Proceedings of 1992 IEEE Computer Society Symposium on Research in Security and Privacy, Oakland, California, pp. 240 – 250, 1992.
[11] DARPA, DARPA Neural Network Study, Chap. 8. AFCEA International Press, 1988.M. Minsky and S. Papert, Perceptrons, Expanded Edition. MIT Press, 1988.