ISSN 2007-9737

A New High Performances Intrusion Detection System based on Dempster-Shafer Theory

Abdelkader Alem[1], Bendaoud Mebarek[2,*], Sid Ahmed Mokhtar Mostefaoui[2], Hadj Ahmed Bouarara[3]

[1] University of Tiaret,
Laboratoire de Génie Énergétique et Génie Informatique (L2GEGI), Tiaret, Algeria

[2] University of Tiaret,
Laboratoire de Recherche en Intelligence Artificielle et Systèmes (LRIAS), Tiaret, Algeria

[3] University Dr Moulay Tahar of Saida,
Department of Computer Science,
Algeria

abdelkader83.alem@univ-tiaret.dz, bendaoud.mebarek@univ-tiaret.dz

Abstract. Intrusion detection systems (IDS) have become a very important element in securing any network infrastructure. Malicious attacks have resulted in negative impacts on network as before, increasing the need for an effective approach to detect and identify such attacks more effectively. In context of improving performance, this paper presents a new adaptive intrusion detection system based on Dempster Shafer theory. It is a hybrid and multi-levels model. Each level includes two classifiers Naïve Bayes and Support Vector Machine known for their performance in classification. The decision at each level is performed using fuzzy logic and the combination rule of Dempster. The experiments were carried out with KDD'99 data sets. The experimental results show that our approach greatly improves the detection rate with low false alarm rates compared to some recent existing works.

Keywords. Dempster-Shafer theory, intrusion, system, IDS.

1 Introduction

Nowadays, our civilization becomes increasingly numerous and proposes to us formidable opportunities related to the use of new technologies in various fields. Nevertheless, with this evolution computer security has become an important necessity. Network security is a sensitive and serious problem that requires the implementation of several tools and mechanisms such as authentication, cryptography, access control and firewalls [8] However, these mechanisms are not sufficient to protect systems against malicious attacks. Therefore, a second layer of security is required, such as intrusion detection.

An Intrusion detection system (IDS) is defined as a security management system for machines and networks. An IDS monitors a network or a computer for anomalies that could indicate an intrusion. Intrusion detection systems are usually configured to send alerts to an administrator when they detect suspicious activities.

An IDS acts as the last defensive means in the system security but does not influence the use of the preventive mechanism [8]. However, they suffer from several problems. Indeed, some attacks may go as a normal action, and in this case, we speak of "false negatives". And some alerts are generated compared to a normal behavior, which corresponds to "false positives". In this context we distinguish two approaches to detect intrusions: anomaly detection and misuse detection (signature detection). The first consists in searching known signatures of attacks while

the second consists in defining normal behavior of the system and sets a baseline of normal behavior and intrusion. It will deduce anything that widely deviates from its normal profile as possible intrusions [18].

Different data mining techniques are used to detect intrusion and to deal with the difficulty to make boundaries between normal a behavior and an attack. The requirement for continuous adaptation to a constantly changing environment is an additional difficulty those techniques have to tackle. Dempster-Shafer theory is a powerful strategy for data fusion, which can not only provide an explicit estimation of imprecision and conflict from different sources, but also deal with any unions of hypotheses. Aiming at the high false alarm rates of currently used network intrusion detection methods, we propose to develop a multi-levels and hybrid Network Intrusion Detection System, based on Dempster-Shafer theory.

In this paper, we show the contribution of belief functions to reduce the false alarms rate and increase the intrusion detection rate. To build our model, we have used the same classifiers and the same dataset (training and testing) with same parameters as the model proposed by [1,2]. The main idea of this paper is the use of Dempster's rule of combination to perform the overall decision of the IDS.

The remainder of this paper is organized as follows: section 2 presents the related works; Section 3 presents the proposed model. Next, the experimental results are discussed in Section 4. Finally, the last section concludes the paper.

2. Related Work

Intrusion detection systems have become a very important element in securing any network infrastructure. Malicious attacks have never been more damaging to networks. Which increases the need to an effective approach to detect and identify such attacks. To deal with this challenge and improve the performance of IDS, many hybrid data mining techniques were developed, such as Naïve Bayes [16, 3], Support Vector Machines [20, 11], and decision trees [14].

Horng et al. [10] developed an IDS that integrates a Hierarchical clustering algorithm as well as an SVM technique.

In 2012 Natesan used a Naïve Bayes and Decision Tree as weak classifiers to propose an Adaboost based algorithm [13].

NFPHIDS (A New Fast and High-Performance Intrusion Detection System) by [1] is hierarchical IDS composed of two levels. A first one that includes four fast classifiers Random Forest, Simple Cart, Best first decision tree and Naive Bayes. They are used for their excellent performance. As inputs, the second level uses outputs of the first one that contains Naïve Bayes as final classifier.

Sandeep et al. [22] suggested an IDS-based Support Vector Machine (SVM) and belief functions in which intrusion detection takes place with the help of Dendritic Cell Algorithm and Dempester belief theory along with SVM classification algorithms.

Horng et al. [10] were developed an IDS that integrate a Hierarchical clustering algorithm and the SVM technique.

An Adaboost based algorithm was proposed in [22] using Naïve Bayes and Decision tree as weak classifiers.

Ahmim et al. [2] have proposed multi-level ids based on the intersection of two different classifiers, fuzzy unordered rule induction algorithm and random forests.

XM-RF [17] is a hybrid IDS based on X-Means clustering and Random Forest classification. Firstly, the approach analogous data instances based on their behaviors are grouped using X-Means clustering. Secondly, Random Forest classifier is utilized to rearrange the misclassified clustered data to apropos group.

Moorthy and Sathiyabama [12] have proposed a hybrid fuzzy IDS for wireless local area networks, based on Fuzzy logic. In this IDS a misuse detection module is connected to the anomaly detection module and the overall decision is performed by fuzzy rules.

In [4], An IDS based belief function was proposed which contain three levels. At the first two modules (SVM detection module and Naïve

Table 1. Confusion matrix

		Predicted class	
		Normal	*Attack*
Actual class	Normal	True negative	False positive
	Attack	False negative	True positive

bayes detection module) were used. In the second, outputs of the first level are fuzzified using fuzzy logic. The last level use belief function to perform the final decision of the system.

2.1. Overview of Dempster-Shafer Theory

Dempster-Shafer theory (DST) is a mathematical theory of evidence [15]. It can be interpreted as a generalization of the probability theory. The significant innovation of this framework is the fact that it allows the allocation of probability into either sets or intervals. In traditional probability theory, the evidence is associated with only one possible event. However, in DST, evidence can be associated with multiple possible events.

An important aspect of this theory is the combination of evidence obtained from multiple sources and modelling of conflict between them.

In DST, a set of disjoint hypotheses of interest, ɵ= {Normal, Abnormal} is called a frame of discernment.

The basic probability assignment {bpa} function is also called mass function mɵ. The term “basic probability” does not refer to probability in the classical sense.

The bpa function is defined as:

$$m_\theta: 2^\theta \rightarrow [0,1]$$
$$m_\theta(\phi) = 0 \; and \sum_{A_i \in \theta} m_\theta(A_i) = 1\,, \tag{1}$$

m is interpreted as a measure of the total belief committed to A_i.

DST is a useful strategy for data fusion. The goal of the combination is to fuse the evidence for a hypothesis from multiple sources and calculate an overall belief for that hypothesis. In general, mass distribution from different sources is combined with Dempster’s rule:

$$m_1(A) \oplus m_2(A) = \frac{1}{1-k} \sum_{A_1 \cap A_j = A} m_\theta(A_i) \cdot m_\theta\left(A_j\right), \tag{2}$$
$$k = \sum_{A_1 \cap A_j = \phi} m_\theta(A_i) \cdot m_\theta\left(A_j\right).$$

3 Proposed Approach for Intrusion Detection

The aim of our work is to show the use of belief functions in intrusion detection and build an efficient hierarchical and Hybrid IDS based on the Dempster-Shafer theory that will provide good results.

There are different methods and techniques to distinguish the various types of classifiers in data mining. Each classifier can sort every network connection as a normal behavior or an attack with a varying error rate.

The performance of the different type of classifiers is measured by its ability to classify each connection in the right category. The table 1, known as the confusion matrix, shows the four possible cases:

- True positive (TP): an attack data identified as an attack.
- True negative (TN): a normal data identified as normal.
- False positive (FP): a normal data identified as an attack.
- False negative (FN): an attack data identified as normal.
- The performance of IDS is evaluated in term of accuracy, detection rate and false alarms rate:
- Accuracy = (TP+TN) / (TP+TN+FP+FN).
- Detection Rate = (TP) / (TP+FP).

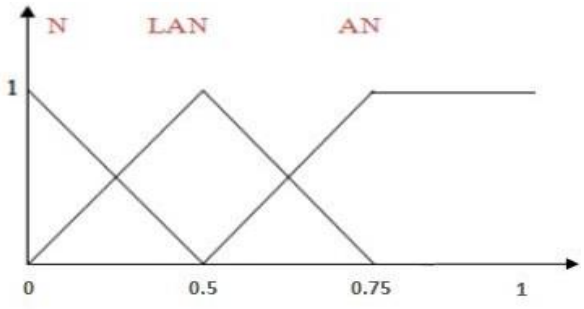

Fig.1. Fuzzy predicates for classifier outputs

– False Alarm Rate = (FP) / (FP+TN).

In this paper, a new approach is proposed. It consists in combining two different classifiers in each level, the number of iterations needed to build the model is the number of levels.

3.1 General Structure of our Model

This section presents the different steps needed to build our model which is based on the TDS theory to merge the predictions of two classifiers. Our adaptive model is composed of N levels, each level contains two classifiers, the predictions of the classifiers are fuzzified using the trapezes associate to each classifier integrating linguistic variables (N: for the normal behavior, AN: for the abnormal behavior and LAN for slightly abnormal (indecision)) as illustrated by figure 1.

Our choice of using Fuzzy Logic was based on two main reasons: (1) No clear boundaries exist between normal and abnormal events (2) the initialization of the belief masses for the fusion step

In each level, the belief function was used for the decision making based on the input from the two classifiers level (fuzzification of classifiers results). After the combination of the results, there are three possible outputs (Abnormal, Normal, and Slightly Abnormal).

Let us consider an example with given sets: Classifier 1= Fuzzy Set (0/AN, 0.05/SAN, 0.995/N) We can see that the possibility for N has the highest membership function. Which means that the possibility N is most likely to be detected than the other two possibilities.

Classifier 2= Fuzzy Set (0.883/AN, 0.117/SAN, 0/N).

Now we can see that the situation is the same for AN, meaning it is the most likely to be detected.

The outputs of Classifier 1 and Classifier 2 are combined using Dempster's rule. The result obtained in our case is Normal (N), since it has the maximum bpa. Hence, that Normal appears to be the best decision for the given constraint.

3.2 Steps in Building our Model

The main idea for the construction of our model is the use of fuzzy logic to properly model the behavior of the system and of belief functions as a merge tool for classifier outputs.

At first, we build the first level based on the initial training dataset. Then, we use all the records that have been classified by the belief function as slightly abnormal (LAN). The records that present a total conflict between the two classifiers are used as new Learning data to train another new level.

In the merging step of each level the Dumpsters rule is used. The stop condition is the absence of conflict between the two sources (classifiers). Inputs with conflict or indecision (LAN) are used as new learning data of a new level. This process is repeated until the conflict measurement is stable and in this case this connection is considered as new attack (all classifiers of all levels are in conflict with one another).

We can summaries the building process of the initial model by the following algorithm:

Model Algorithm

```
Input: Training Data (TD);
Output: Adaptive Intrusion Detection Model
(AIDM) K=0; //measure of conflict between
classifiers
I=1; // number of
level AIDM= null;
Begin
While (C= = False) do
begin
A= subset of D;
B= subset of D // B different from A;
        M1= Model of classifier 1 using A
        dataset; M2= Model of classifier 2
        using B dataset;
AIDM= add(M1 and M2 as classifiers of level I);
//set the two trained model as classifier of
level I I++;
D1= test TD with M1; // Test all records with
M1 D2= test TD with M2; // Test all records
with M2
    SAN= all connections declared as slightly
      Abnormal from fusion module;
CON= all conflicted connections generated
      from fusion module;
Ki= conflict degree between Classifiers of
level I; TD= SAN union CON
If (Ki = =Ki-1) then C=
True; End;
DLL=TD // the training dataset of the last
    level; Return AIDM;
    END.
```

3.3 Testing and Adaptation Procedure

To test the different records, the top-down propagation is applied to all levels. For each level, we test if the decision of the fusion module is Completely Normal (N) or Completely Abnormal (AN), then we stop the process, and we classify the record. Otherwise, we test the record by the next level. If every two classifiers of all levels of the built model give the same conflict degree, this record is labelled as a new attack and added to a new cluster of attacks.

To ensure the adaptation of the proposed model and if the conflict between sources reaches the threshold (K = 1, the case of unreliable sources) we create a new level.

Otherwise, we cluster the new attacks with the rest of the training dataset of the last level (DLL) to build a new one.

4 Experiments and Results

This section is divided into three parts. The first one describes the data set used in our experiments. The second part describes the parameters and the classifiers used to train our model. The last part represents a comparative study between our model and related works.

4.1 . Data Set Description

Our method has been evaluated using two datasets:

The first one involves a real web traffic collected on a university campus. As for Web attacks, several recent Web attacks are simulated targeting either server-side Web applications or client-side ones. Attacks are simulated on a simulation network involving exactly the same environment as the one where real HTTP traffic was collected. In particular, the Web server software, Web site content, operating system and Web clients. Our dataset contains 98995 connections with 24 features and a test dataset with 69015 connections; Table 2 shows the distributions of connections categories.

The second dataset is the one by Ahmim et al. [2]. To make a reliable comparison we used the same learning (KDD99) and test datasets with exactly the same parameters used in the reference work. The KDD99 Cup was derived in 1999 from DARPA-Lincoln98 dataset by MIT's Lincoln laboratory.

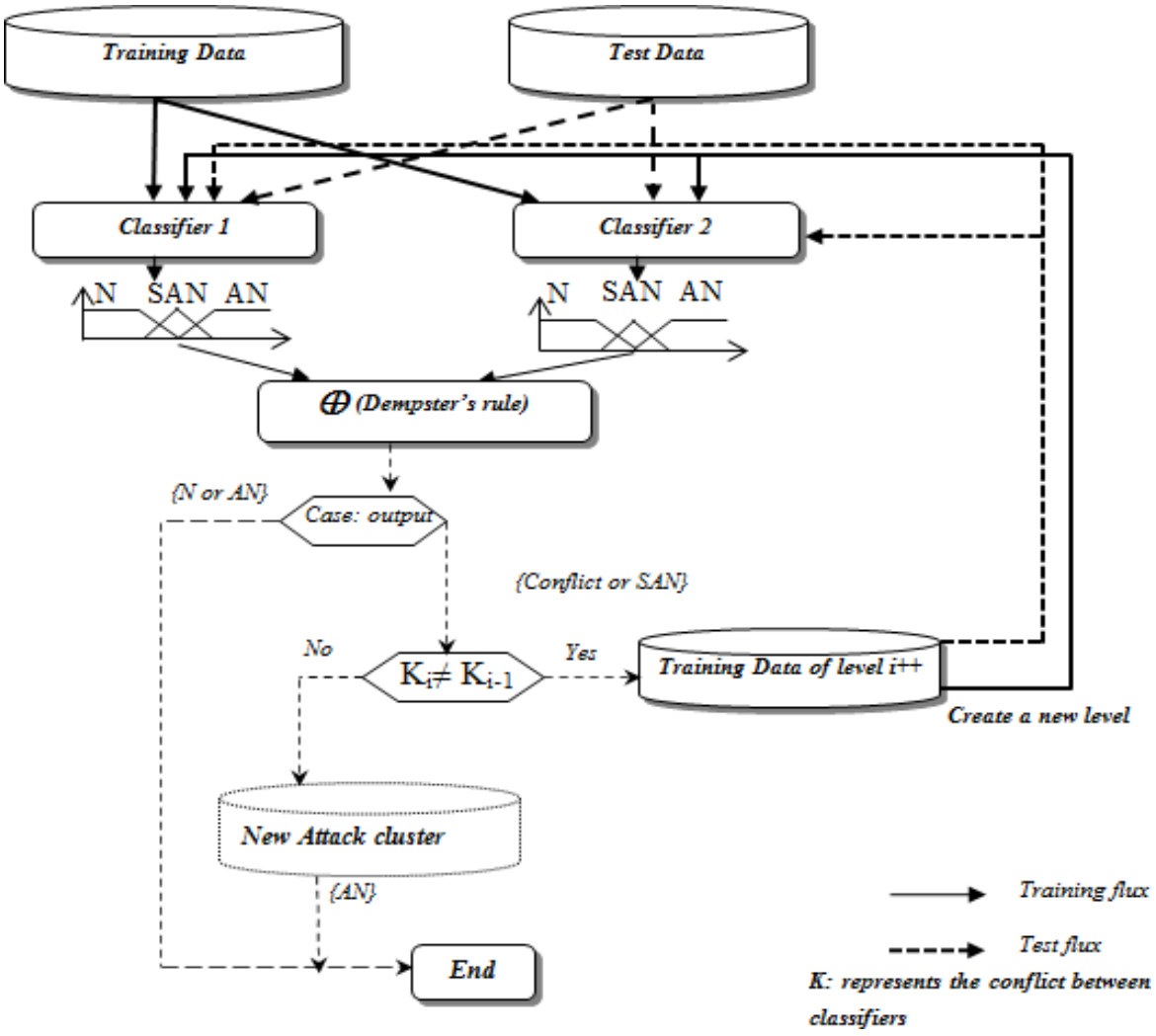

Fig. 2. General structure of our Model

The KDD99 contains 39 attacks and normal behaviors. It is organized into five classes: DOS attack, U2R attack, PROB attack, R2L attack and Normal behavior [10].

Each record of the KDD99's has 41 features: 34 numeric and 7 symbolic.

According to Sundus et al. [17], the KDD99 Training Data Set contains normal behaviors and 22 attacks with 4,940,000 data records. As for the Test Data Set, it contains 311,029 data records, covering normal behaviors and 37 attacks. It should be noted that 17 of the Test Data Set attacks don't exist in the Training Data Set.

KDD99_10% represents 10% of KDD99 Training Data set with the same distribution of attacks and normal behaviors. The following table shows the distribution of the attacks and normal behaviors in both the KDD99 training_10% and the KDD99 test.

We reduced the size of KDD99_10% by removing all redundant records, thus creating our own training data set containing 40,000 records. A random selection is used to select

Table 2. Distribution of connection categories in the dataset

	Learning data set		Test data set	
Class	**Number**	**%**	**Number**	**%**
Normal Connections	55344	55,90	61378	88,93
Abnormal connections attack	43651	44,10	7637	11,07
Total	98995	100	69015	100

Table 3. Dataset description

Number of records	**KDD99_10% Training data set**		**KDD99 Test data set**	
Category of connection	**ALL**	**Distinct**	**ALL**	**Distinct**
Normal	97278	87832	60593	47913
DOS	391458	54572	229853	23568
Probe	4107	2130	4166	2678
R2L	1126	999	16189	2913
U2R	52	52	228	215
ALL	494021	145585	311029	77287

Table 4. Comparison betw

	Ahmim et al. [2]			Our Approach		
Corpus	**Accuracy**	**DR**	**FAR**	**Accuracy**	**DR**	**FAR**
KDD'99	95.41%	94.84%	2.23%	96.88%	96.88%	0.61%
Web attacks	98.91%	95.90%	0.68%	99.94%	99.84%	0.04%

Normal and Attack records. The KDD99 test data set KDDcup [21] is used to evaluate the performance of our model.

4.2 Classifiers and Parameters used to Build and Train our Model

To implement our approach, we have chosen the same data mining techniques used in [2].

These two techniques are very different in their operating process and provide a high detection rate and a good heterogeneity in the classification of the different network connections.

The first classifier is Fuzzy Unordered Rule Induction Algorithm (FURIA) that represents a fuzzy rule-based classification method. FURIA is based on the well-known RIPPER algorithm [18]. It preserves their advantage and introduces some modifications, by using fuzzy rules instead of conventional rules and unordered rule sets instead of rule lists [6].

The second classifier is Random forests (RF) with combined tree predictors. Each tree depends on the values of a random vector sampled independently and also on the same distribution for all other trees in the forest. The classification error of the forest of tree depends on the robustness of the different trees in the forest and the correlation between them [5].

To increase the conflict between the two classifiers of our training data set, we have used

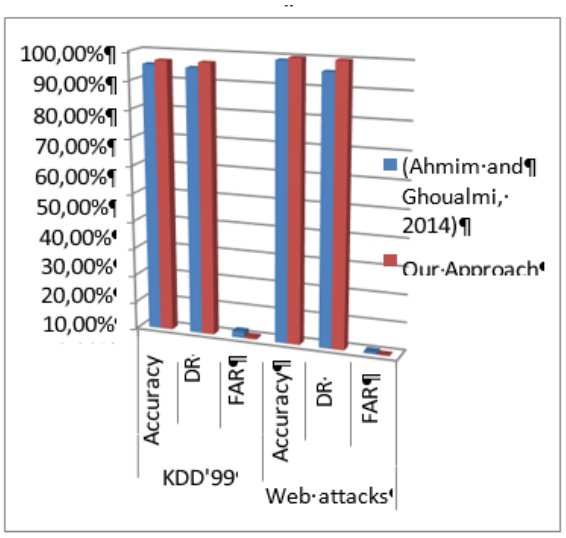

Fig. 3. Comparison between Ahmim et al. [2] and our model

in our experiments two different sub-sets (A and B). Those sets ate used to train respectively FURIA and RF. This heterogeneity of classifiers and training dataset allows us to build a model with multi-levels.

5. Results and Discussion

In our work, WEKA Data Mining Tools [19] has been used to implement both classifiers. The results were obtained using a Pc operating on Microsoft Windows OS and equipped with a Core i5 2,4 GHZ CPU and 4 GB RAM.

To evaluate the performance of the proposed approach as well as to improve the belief functions in intrusion detection utility, we have compared the obtained results with those mentioned in [2] In the chosen reference work all KDD99 Test dataset (KDD99) are used for the validation.

In this study, we have used the parameters mentioned in detail in the previous section to build and to train the model. Then, all KDD99 Test Dataset were used as a test dataset. To add more confidence in the performance of our model we added another factor on a recent web attacks data set. The results are shown in Table 4.

As illustrated in figure 3, our approach (DSIDS) Ives the lowest false alarms rate with the highest accuracy without losing a good detection rate compared to Ahmim et al. approach [2] using KDD dataset.

Our model demonstrates high performance across all three evaluation measures Detection Rate (DR), Accuracy, and False Alarm Rate (FAR) when tested on the web attack dataset.

6. Conclusion

In this paper; we have presented an efficient hybrid and multi-level intrusion detection system based on the belief function and the fuzzy logic. To build a high performed and speed model, we have used Fuzzy Unordered Rule

Induction Algorithm and Random Forests as classifiers. The decision-making system is based on Dempster-shafer theory. This methodology of finding the intrusion in the system is suitable since both classifiers outputs are separately performed and the overall system status is cross checked by the decision-making module processed by Dempsters' rule of combination.

This hierarchical and hybrid technique is very scalable and accurate. The accuracy is due to the fact that incorrect interpretation is very unlikely thanks to the double-checking technique in all system levels. Hence, the use of belief functions technique in intrusion detection is very efficient to maintain the system security.

The experiment results illustrate that the proposed approach gives a good performance. While it gives a very low false alarm rate it also preserves a high detection rate and accuracy. This is true even when compared to well-known works in the literature using exactly the same parameters and the same training and test dataset.

References

1. **Ahmim, A., Ghoualmi-Zine, N., (2013).** A New Fast and High Performance Intrusion Detection System. International Journal of Security and Its Application, Vol. 7, No. 5, pp. 67–80.

2. **Ahmim, A., N. Ghoualmi-Zine, (2014).** A new adaptive intrusion detection system based on the intersection of two different classifiers. International Journal of Security and Network, Vol. 9, No. 3, pp. 125–132.
3. **Amor, N.B., Benferhat, S., Elouedi, Z. (2004).** Naïve Bayes vs. decision trees in intrusion detection systems. In Proc. of ACM Symposium on Applied Computing, pp. 420–424.
4. **Alem, A., Dahmani, Y., Hadjali, A. (2016).** On the use of Belief Functions to improveHigh Performance Intrusion Detection System. 12th International Conference on Signal-Image Technology & Internet-Based Systems. Doi: 10.1109/SITIS.2016.50.
5. **Breiman, L. (2001).** Random forests. Machine Learning, Vol. 45, No. 1, pp. 5–32.
6. **Cohen, W. (1995).** Fast effective rule induction, Proceedings of the 12th International Conference on Machine Learning (ICML'95), pp. 115–123.
7. **Dempster, A.P. (1967).** Upper and Lower probabilities induced by a multivalued mapping. Annals of Mathematical Statistics, Vol. 38, No. 2, pp. 325–339.
8. **Evangelista, T. (2004).** Les IDS - Les systèmes de détection d'intrusions informatiques. Dunod.
9. **Hühn, J., Hüllermeier, E. (2009).** FURIA: an algorithm for unordered fuzzy rule induction. Data Mining and Knowledge Discovery,Vol. 19, No. 3, pp. 293–319.
10. **Horng, S., Su, M., Chen, Y., Kao, T., Chen, R., Lai, J., Perkasa, C. D. (2011).** A novel intrusion detection system based on hierarchical clustering and support vector machines. Expert Systems with Applications, Vol. 38, No. 1, pp. 306–313.
11. **Mukkamala, S., Janoski, G., Sung, A. (2002).** Intrusion Detection Using Neural Networks and Support Vector Machines. Proceedings of the 2002 International Joint Conference on Neural Networks, Honolulu, pp. 1702–1707.
12. **Moorthy, M., Sathiyabama, S. (2011).** Hybrid Fuzzy Based Intrusion Detection System for Wireless Local Area Networks. European Journal of Scientific Research, Vol. 53, No. 3, pp. 431–446.
13. **Natesan, P., Balasubramanie, P., Gowrison, G. (2012).** Improving the Attack Detection Rate in Network Intrusion Detection using Adaboost Algorithm. Journal of Computer Science, Vol. 8, No. 7, pp. 1041–1048.
14. **Paek, S., Oh, Y., Lee, D. (2006).** sIDMG: Small-Size Intrusion Detection Model Generation of Complimenting Decision Tree Classification Algorithm. Proceedings of the 7th International Workshop, WISA 2006, Jeju Island, Korea, Springer Berlin Heidelberg, Vol. 28-30, pp. 83–99.
15. **Shafer, G. (1976).** A Mathematical Theory of Evidence. Princeton, NJ, Princeton. University Press.
16. **Scott, L.S. (2004).** Bayesian paradigm for designing intrusion detection systems. Computational Statistics and Data Analysis, Vol. 45, No. 1, pp. 69–83.
17. **Sundus, J., Zaiton, M., Warusia, Y., (2014).** Reducing false alarm using hybrid intrusion detection based on x-means clustering and random forest classification. Journal of Theoretical and Applied Information Technology, Vol. 68, No. 2, pp. 249–254.
18. **Wu, S. Xiaonan, Banzhaf, W. (2010).** The use of computational intelligence in intrusion detection systems. Applied Soft Computing, Vol. 10, No. 1, pp. 1–35.
19. **Witten, I, Frank, E., Hall, M. (2011).** Data Mining: Practical Machine Learning Tools and Techniques. Elsevier Inc.
20. **Zhang, Z. and Shen, H. (2005).** Application of online-training SVMs for real-time intrusion detection with different considerations. Computer Communications, Vol. 28, No. 12, pp. 428–1442.
21. **KDD (2017).** The KDD CUP 1999 Data (KDD99) [online]http://kdd.ics.uci.edu/databases/kddcup 99/kddc up99.html.
22. **Sandeep, S., Jasvinder. P., Gaurav, S. (2013).** A hybrid artificial immune system for IDS based on SVM and belief function. Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), Doi: 10.1109/ICCCNT.2013.6726 835.

Article received on 02/12/2023; accepted on 14/12/2025.
**Corresponding authors is Bendaoud Mebarek.*