Labelling Clusters in an Intrusion Detection System Using a Combination of Clustering Evaluation Techniques

Slobodan Petrović[1], Gonzalo Álvarez[2], Agustín Orfila[3], and Javier Carbó[3]
[1] *NISlab, Department of Computer Science and Media Technology, Gjøvik University College, P.O. box 191, 2802 Gjøvik, Norway*
[2] *Institute of Applied Physics (C.S.I.C.), Serrano 144, 28006 Madrid, Spain*
[3] *Carlos III University of Madrid, 28911 Leganés, Madrid, Spain*
slobodan.petrovic@hig.no, gonzalo@iec.csic.es, {adiaz,jcarbo}@inf.uc3m.es

Abstract

A new clusters labelling strategy, which combines the computation of the Davies-Bouldin index of the clustering and the centroid diameters of the clusters is proposed for application in anomaly based intrusion detection systems (IDS). The aim of such a strategy is to detect compact clusters containing very similar vectors and these are highly likely to be attack vectors. Experimental results comparing the effectiveness of a multiple classifier IDS with such a labelling strategy and that of the classical cardinality labelling based IDS show that the proposed strategy behaves much better in a heavily attacked environment where massive attacks are present. The parameters of the labelling algorithm can be varied in order to adapt to the conditions in the monitored network.

1. Introduction

Intrusion detection systems (IDS) are security tools designed to detect and classify attacks against computer networks and hosts. They can operate in two ways: either by searching for specific patterns in data (misuse based IDS) or by recognising certain deviations from expected behaviour (anomaly based IDS). In anomaly based IDS, clustering algorithms are often used for recognition of ”abnormal” behaviour. They can be applied either directly on incoming data [6, 9, 18] or as a supporting technique in a stage posterior to data classification performed by means of other techniques [12, 24].

Anomaly based IDS classify input data into a number of categories, or classes. This number can be arbitrary, but as the essential goal of these systems is to distinguish between ”normal” and ”abnormal” behaviour, it is very common to partition the incoming resource access requests into two classes that correspond to these two types of behaviour. The data are submitted to the system in the form of lists created at predefined time intervals or alternatively, upon a predefined number of incoming requests. Then the system makes the decision about whether abnormal behaviour occurred or not, based on the obtained classification results.

In this paper, we consider the Denial-of-Service (DoS) attack scenario in which attack resource access requests arrive to the monitored network in bursts. An anomaly based IDS interprets this situation in the following way: If it analyses N resource access requests at a time then more than $N/2$ of these requests correspond to attacks. We call such a scenario a massive attack. Sometimes, other network monitoring tools (firewalls etc.) can detect such attacks, but the advantage of an anomaly based IDS regarding all kinds of attacks (including massive attacks as defined in this paper) is in the capability of detecting a completely new attack.

If clustering is used for classification of resource access requests in an IDS, the main problem is the interpretation of clustering results, so called ”labelling” of clusters. Namely, without additional information (which is, by contrast, always present in the systems with learning) it is difficult to decide whether the data classified in one cluster correspond to ”normal” behaviour in the monitored network or to ”abnormal” behaviour. Cardinalities of clusters are often used as a decision parameter for this purpose (see, for example, [18]) because the mathematical expectation of ”normal” behaviour is considered greater than that of ”abnormal” behaviour. However, this approach has some serious drawbacks and fails to detect massive attacks. Solving this problem requires a more complex clusters labelling algorithm.

We propose a clusters labelling strategy based on a combination of clustering evaluation techniques. The Davies-Bouldin clustering evaluation index and the comparison of centroid diameters of the clusters are combined in order to respond adequately to the properties of attack vectors. We consider the compactness of the corresponding clusters

0-7695-2507-5/06/$20.00 (C) 2006 IEEE

and the separation between them the principal parameters that distinguish "normal" from "abnormal" behaviour in the analysed network. Because of that, clustering evaluation techniques that take into account these parameters are applied in our IDS. We found the Davies-Bouldin index the most convenient measure that can be used for labelling clusters in IDS. In the experiments, we test the response of a multiple classifier IDS (see, for example, [7]) with the new labelling strategy implemented to artificial data and express the IDS quality through Receiver Operating Characteristics (ROC) curves. The effectiveness of our IDS is compared with that of a system of the same kind implementing the classical clusters cardinalities based labelling algorithm.

In the experiments, we tested our labelling algorithm on the well known KDD CUP artificial data set [5, 13], which was used as the traffic source. Although this source has been criticized in the literature (see, for example, [15, 23]), it is still being used for IDS benchmarking [1, 8, 17]. We found it convenient as a source of massive attacks, against which we have tested our labelling strategy. The experimental results show that the labelling strategy proposed in this paper works well even in a heavily attacked environment to which the KDD CUP data set corresponds and because of which (as one of the reasons) it has been criticized as being "unrealistic".

The structure of the paper is the following: In Section 2, a general description of the analyzed intrusion detection system is given. In Section 3, the new clusters labelling method combining the Davies-Bouldin index and the centroid diameters comparison is described in detail. In Section 4, the experimental work is described and the results of the experiments are given. Finally, Section 5 concludes the paper.

2. General description of the system

The multiple classifier IDS, whose elements we analyze in this paper, consists of the following components (Fig. 1):

1. $\mathcal{K}$ sensors, $\mathcal{P}_1, \ldots \mathcal{P}_{\mathcal{K}}$, which operate in parallel on the same data set $\mathbf{X}_\tau$, $\tau = 0, 1, 2, \ldots$ We limit ourselves to the case in which every sensor is merely a clustering algorithm that classifies the input data set into K clusters, without any interpretation of clustering results.

2. $\mathcal{L}$ assessors, $\mathcal{A}_1, \ldots, \mathcal{A}_{\mathcal{L}}$, whose task is to "label" the clusters obtained from the sensors, upon processing the current data set $\mathbf{X}_\tau$. For this to be carried out, every assessor calculates the value of its own criterion function for every sensor over the data set $\mathbf{X}_\tau$. A local extreme value (maximum or minimum) of this function determines the decision of the assessor on the following: an element of $\mathbf{X}_\tau$ belongs to a cluster that is interpreted as one of the "normal" clusters or it belongs to a cluster that is interpreted as the "abnormal" one.

3. The manager of the system adjusts the parameters of the sensors and the assessors in order to maximize the effectiveness of the system as a whole.

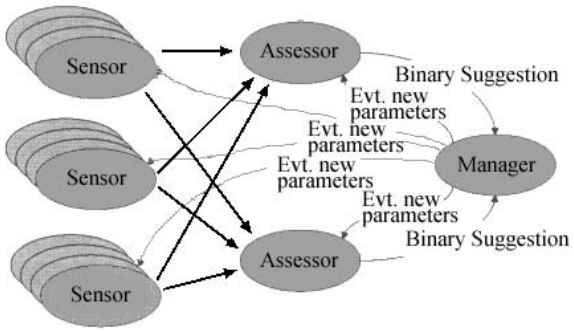

Figure 1. A multiple classifier IDS

In this paper, we concentrate on the basic sensor-assessor structure. The former actually performs the clustering of the incoming resource access requests, whereas the latter performs the clustering quality evaluation.

We have selected the well known K-means algorithm (see for example [11]) for implementation in the sensors of the IDS, because we consider this algorithm the best trade-off between accuracy and efficiency. The K-means algorithm is presented in the Fig. 2.

1. Initialization: Randomly choose *K* vectors from the data set and make them initial cluster centers.

2. Assignment: Assign each vector to its closest center.

3. Updating: Replace each center with the mean of its members.

4. Iteration: Repeat steps 2 and 3 until there is no more updating.

Figure 2. The K-means algorithm

The input resource access requests are encoded in such a way that vectors of the same length are produced. The Euclidean metric, given by the following expression, is used in our system as a distance measure between vectors.

$$d(\mathbf{X}, \mathbf{Y}) = \sqrt{\sum_{i=1}^{n} \mid x_i - y_i \mid^2} \qquad (1)$$

where n is the dimension of the vectors $\mathbf{X}$ and $\mathbf{Y}$.

3. The new labelling algorithm

Having obtained clusters from the sensors, the task of the assessors is to label them, i.e. to determine which clusters correspond to "normal" behaviour, and which to "abnormal". Since there is no learning on labelled data in the system, the assessors must use other criteria to decide on this. A common assumption is that few anomalies are expected in the clustering results, so a significant difference in cardinalities of the clusters naturally labels the cluster with the greatest cardinality as that corresponding to "normal" behaviour. However, there are at least two problems related to such a strategy [18]: first, normal data transmitted by means of a less frequently used protocol (such as *ftp* or *telnet*) might produce clusters of very different cardinalities, which could mislead such an assessor. Second, there are some Denial-of-Service attacks, such as *syn-Flood*, that can mislead this labelling strategy by making the mathematical expectation of the attack much greater than that of a "normal" behaviour. To overcome the problems related to the labelling strategy described above, we propose a combination of clustering evaluation techniques to be used in the assessors of the IDS.

General clustering evaluation measures defined in the literature can be implemented in intrusion detection systems relatively easily. All these measures opt for detecting well separated and compact clusters. The following clustering evaluation measures are often used: Silhouette index, Dunn's index, Davies-Bouldin index, etc. [2, 10]. In order to present the definition of the Davies-Bouldin index, we first have to define the *inter-cluster* and *intra-cluster* distance.

Inter-cluster distance

The inter-cluster distance is a measure of separation between clusters. The most frequently used measures of this kind are: single linkage, complete linkage, average linkage, centroid linkage, average of centroids linkage, Hausdorff metrics, etc.

Intra-cluster distance

The intra-cluster distance is a measure of compactness of clusters. The following are the most frequently used intra-cluster distances: complete diameter, average diameter and centroid diameter.

□

Our choice for the IDS assessing algorithm is the Davies-Bouldin index combined with the centroid diameters comparison between clusters. Besides, in the computation of the Davies-Bouldin index, the centroid linkage is used as the inter-cluster distance. The centroid inter-cluster and intra-cluster measures are selected for compatibility with the K-means clustering algorithm used in the sensors (which essentially computes centroids of clusters at each iteration).

The Davies-Bouldin index is selected because of the following advantages over other measures:

1. Stability of results: this index is less sensitive to the position of a small group of data set members (so called outliers) than other measures, such as for example, the Dunn's index [10].

2. In the case of more than 2 clusters and the need to rank them, some measures (for example the Silhouette index) behave unpredictably, whereas the expected behavior of the Davies-Bouldin index in those cases is good [2].

Let $\mathbf{X}_\tau = \{\mathbf{X}_1, \ldots, \mathbf{X}_N\}$ be the data set and let $\mathcal{C} = (C_1, \ldots, C_K)$ be its clustering into K clusters. Let $d(\mathbf{X}_k, \mathbf{X}_l)$ be the distance between $\mathbf{X}_k$ and $\mathbf{X}_l$. Then the Davies-Bouldin index is defined in the following way [2, 10]:

$$DB(\mathcal{C}) = \frac{1}{K}\sum_{i=1}^{K} \max_{i \neq j} \left\{ \frac{\Delta(C_i) + \Delta(C_j)}{\delta(C_i, C_j)} \right\}. \quad (2)$$

where $\Delta(C_i)$ is the intra-cluster distance and $\delta(C_i, C_j)$ is the inter-cluster distance. In the observed IDS, the centroid diameter is used for $\Delta(C_i)$. It is defined in the following way [2]:

$$\Delta(C_i) = 2\left(\frac{\sum_{\mathbf{X}_k \in C_i} d(\mathbf{X}_k, s_{C_i})}{\mid C_i \mid} \right), \quad i = 1, \ldots K, \quad (3)$$

where $s_{C_i} = \dfrac{1}{\mid C_i \mid} \displaystyle\sum_{\mathbf{X}_k \in C_i} \mathbf{X}_k$.

The centroid linkage inter-cluster distance is used for $\delta(C_i, C_j)$. It is defined in the following way [2]:

$$\delta(C_i, C_j) = d(s_{C_i}, s_{C_j}), \quad (4)$$

where $s_{C_i} = \dfrac{1}{\mid C_i \mid} \displaystyle\sum_{\mathbf{X}_k \in C_i} \mathbf{X}_k$ and $s_{C_j} = \dfrac{1}{\mid C_j \mid} \displaystyle\sum_{\mathbf{X}_k \in C_j} \mathbf{X}_k$.

For the remainder of this paper, we shall limit ourselves to studying the 2 clusters case, of which one corresponds to "normal" and the other to "abnormal" behaviour in the corresponding network. This is done primarily because of an easier comparison with the cardinality based labelling algorithm during the experiments. In addition, whatever the number of clusters we use in the sensors, we must finally decide which of them will be considered "normal", leading us to a case with 2 "superclusters".

In the experiments, we compare the results obtained with our assessing algorithm with the results obtained with the clusters cardinalities criterion, a common measure for assessing IDS clusters (see for example [18]). We define this criterion in the following way:

Let $\mathbf{X}_\tau = \{\mathbf{X}_1, \ldots, \mathbf{X}_N\}$ be the current data set and let $\mathcal{C}_{k,\tau} = \{\mathbf{Y}_{k,\tau}, \mathbf{Z}_{k,\tau}\}$ be the partition of $\mathbf{X}_\tau$ into 2 clusters, obtained in the sensor $\mathcal{P}_k$. Let $\lambda_j \in \{1, 2\}$ be the label of the vector $\mathbf{X}_j$ in the data set $\mathbf{X}_\tau$, where $\lambda_j = 1$ is interpreted as "normal" behaviour. If $\mid \mathbf{Y}_{k,\tau} \mid \geq \mid \mathbf{Z}_{k,\tau} \mid + \mathcal{D}_C$, where $\mathcal{D}_C$ is a threshold given in advance then $\lambda_j = 1$ for $\mathbf{X}_j \in \mathbf{Y}_{k,\tau}$ and $\lambda_j = 2$ otherwise. If $\mid \mathbf{Z}_{k,\tau} \mid \geq \mid \mathbf{Y}_{k,\tau} \mid + \mathcal{D}_C$, then $\lambda_j = 1$ for $\mathbf{X}_j \in \mathbf{Z}_{k,\tau}$ and $\lambda_j = 2$ otherwise.

The main idea of our clusters labelling algorithm, which uses the Davies-Bouldin index and centroid diameters of the clusters is the following:

The attack vectors are often mutually very similar, if not identical. For example, the corresponding cluster in the case of a massive attack is extremely compact and the Davies-Bouldin index of such a clustering is either 0 (when the non-attack cluster is empty) or very close to 0. Having in mind the expected mutual similarity among attack vectors, the centroid diameter of the attack cluster is expected to be smaller than that of the non-attack cluster. In the exceptional case in which one of the clusters is empty, relabelling is performed if the Davies-Bouldin index of the clustering is equal to 0 and the centroid diameter of the cluster labelled with "2" is equal to 0. Thus lower values of the Davies-Bouldin index indicate the existence of a massive attack, whereas small values of the centroid diameter in these cases indicate the attack cluster.

By contrast, when the Davies-Bouldin index takes higher values, i.e. where massive attacks do not exist, the centroid diameter of the cluster labelled with "1" is expected to be smaller than that of the cluster labelled with "2", because in those cases the non-emptiness of the latter cluster is a rare event and it can even be a consequence of an error caused by the very clustering method implemented in the system.

Example 1: In the KDD CUP data set, many attack vectors correspond to the so called "smurf" attack, which is a sort of DoS attack. Table 1 shows the differences between the coordinates of two attack vectors that correspond to the "smurf" attack. Table 2 shows the differences between two "normal" vectors. In this particular example it is easy to see that the difference between two attack vectors is much smaller than the difference between two "normal" vectors.

The study above gives rise to the following labelling algorithm:

Algorithm 1
Input:

- A clustering $\mathcal{C}$ of N vectors into 2 clusters, C_1 and C_2, in which the vectors belonging to the "non-attack" cluster C_1 take the label "1", and those belonging to the "attack" cluster C_2 take the label "2".
- The Davies-Bouldin index threshold, Δ_{DB}.
- The centroid diameters difference thresholds, Δ_{CD_1} and Δ_{CD_2}.

Table 1. The differences between two attack vectors in the KDD CUP data base (records 7635 and 7636 of the reduced (10%) data set). The rest of 41 coordinates are equal to 0.

Coord. id.	Rec. 7635	Rec. 7636
protocol_type	2	2
service	50001	50001
flag	10	10
src_bytes	1032	1032
count	511	511
srv_count	511	511
same_srv_rate	100	100
dst_host_count	228	238
dst_host_srv_count	83	93

Table 2. The differences between two "normal" vectors in the KDD CUP data base (records 6 and 7 of the reduced (10%) data set). The rest of 41 coordinates are equal to 0.

Coord. id.	Rec. 6	Rec. 7
service	80	80
flag	10	10
src_bytes	212	159
dst_bytes	1940	4087
logged_in	1	1
count	1	5
srv_count	2	5
same_srv_rate	100	100
srv_diff_host_rate	100	0
dst_host_count	1	11
dst_host_srv_count	69	79
dst_host_same_srv_rate	100	100
dst_host_same_src_port_rate	100	0

□

Output:

- The eventually relabelled input clustering, if relabelling conditions are met.

begin
$db \longleftarrow DaviesBouldinIndex(\mathcal{C})$;
$cd_1 \longleftarrow CentroidDiameter(C_1)$;
$cd_2 \longleftarrow CentroidDiameter(C_2)$;
/** Condition 1 **/
if $(db = 0)$ **and** $(cd_2 = 0)$ **then**

$Relabel(\mathcal{C})$
/** Condition 2 **/
else if $(db < \Delta_{DB})$ **and** $(cd_1 + \Delta_{CD_2} < cd_2)$ **then**
$Relabel(\mathcal{C})$
/** Condition 3 **/
else if $(db > \Delta_{DB})$ **and** $(cd_1 > cd_2 + \Delta_{CD_1})$ **then**
$Relabel(\mathcal{C})$;
end.
procedure $Relabel$(Clustering $\mathcal{C}$)
begin
forall vectors in $\mathcal{C}$
if label of the vector is '1' set it to '2'
and vice versa ;
end.

□

Example 2: The Algorithm 1 applied to the first 20000 records of the reduced (10%) KDD CUP data set, upon clustering by the 2-means algorithm where $N = 1000$, produces no labelling errors at all in spite of a very bad initial (i.e. as-clustered) labelling on average. The parameters of the Algorithm 1 are the following: $\Delta_{DB} = 0.45$, $\Delta_{CD_1} = 500$ and $\Delta_{CD_2} = 0$. The results are summarized in the Table 3.

Table 3. Application of the Algorithm 1 to the first 20000 records of the reduced (10%) KDD CUP data set. The first row of the table corresponds to the records 1-1000 of the KDD CUP data base, the second row corresponds to the records 1001-2000 and so on.

DB index	CD1	CD2	No. of attacks	Relab. cond.*
1.13	32759.24	7108.57	0	3
0.93	17273.10	5216.10	0	3
0.88	16233.92	4771.84	0	3
0.80	50253.55	6979.75	2	3
1.47	76357.90	7273.81	0	3
0.96	4344.63	14158.54	0	0
0.74	7488.44	85018.59	0	0
0.14	69.60	7096.34	376	2
0.00	25.19	0.00	1000	1
0.00	11.38	0.00	1000	1
0.00	0.02	0.00	1000	1
0.15	190.27	7302.86	321	2
1.19	33783.03	5843.60	0	3
1.01	5063.81	22257.14	0	0
0.97	243757.28	8973.01	0	3
1.33	18070.60	8785.56	21	3
1.03	18289.63	5227.67	0	3
1.16	4863.57	22460.80	0	0
1.03	36749.24	6909.28	0	3
0.17	8149.70	423.70	99	0

* See Algorithm 1; 0 means that the initial (as clustered) labelling is correct so no relabelling is performed by the Algorithm 1

The behaviour of the Algorithm 1 depends on the choice of the parameters Δ_{DB}, Δ_{CD_1} and Δ_{CD_2}. These should be determined in advance. One of the ways to do that is to use a network/dataset with known relevant characteristics, of which the most important ones are the base rate, i.e. the probability of the attack (which influences the Davies-Bouldin index of the clustering) and the type of the attack (which influences the centroid diameters of the clusters during the attacks). For example, the KDD CUP data set has the base rate of $\approx 80\%$ and of all the attacks in that dataset $\approx 99\%$ are denial-of-service (DoS) attacks [20]. For a dataset of such characteristics, the best performance of the Algorithm 1 has been obtained by setting the values of the parameters to those used in the Example 2. In a real network, one could start with the parameters of the Algorithm 1 obtained in the controlled network scenario (e.g. with those obtained with the KDD CUP database) and then try to fine tune the parameters over time. However, the choice of the parameters suitable for a particular network may be a real challenge, since we cannot completely control the base rate and the type of attacks.

4. Experimental work

Extensive simulation of the basic sensor-assessor structure of a multiple classifier IDS has been carried out in order to study its response to the attack data. To this end, the following instance of this structure has been built:

1. In the sensor, the 2-means clustering algorithm has been implemented.
2. Two types of assessors have been tested:
   - 2.1 Cardinality based assessor: the cluster of greater cardinality is considered "normal" and is labelled with '1'. The minimum difference $\mathcal{D}_C$ between clusters' cardinalities needed to relabel the clustering is used as a parameter of this assessing algorithm.
   - 2.2 The assessor implementing the Davies-Bouldin index of the clustering and the clusters' diameters, according to the Algorithm 1. The Davies-Bouldin index threshold, Δ_{DB}, and the centroid diameters difference thresholds, Δ_{CD_1} and Δ_{CD_2}, have been used as parameters of this assessing algorithm.

Next, an input data source had to be selected. This task is considered difficult. For example, according to [16], this is one of the challenges of IDS testing. In [16], 4 approaches to this problem are defined, according to the use of background traffic in the test data:

1. testing using no background traffic at all [21];
2. testing using real traffic/logs [22];
3. testing using sanitized traffic/logs [14];
4. testing by generating traffic on a tested network [3, 4, 13, 19, 21].

Each of these approaches has its advantages and disadvantages. The main advantage of testing by generating traffic artificially is the possibility of accurate determination of the number of false alarms, since no unknown attacks can appear in the test data. The quality of such a simulated data source is a separate question. For example, the well known and widely used KDD CUP source [5, 13] has been criticized by various authors (see [15, 23], among others). However, the KDD CUP data set contains many massive attacks (which is typical for a military environment to which it corresponds) and this is a decisive characteristic needed for testing the labelling strategy proposed in this paper.

Thus, we have selected the KDD CUP database as the traffic source for our experiments. The aim was to compare the results obtained by applying the proposed labelling strategy with those obtained by applying the cardinality based labelling strategy, with and without the presence of massive attacks. Because of that, the attacks from the KDD CUP database were filtered out in the same way as in [18]. The filtering percentage of 0%, 98% and 99% was used over all the resource access requests records of the database. Without filtering out the attacks (0%), the database simulates many massive attacks, whereas if the filtering of 98% and 99% of attacks is applied it simulates a "realistic" situation, in which attacks are rare events. The effectiveness of the system was measured by means of the ROC (Receiver Operating Characteristic) curve for the filtered data set mentioned above. The ROC curve depicts the relationship between false positive rate FPR and true positive rate TPR, where:

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \qquad \mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \qquad (5)$$

In the equations above, FP is the number of false positive outcomes of the intrusion detection on a fixed data set, i.e. the number of decisions in which a non-existing attack is signalled, TP is the number of true positive outcomes, i.e. successful detections, TN is the number of true negative outcomes, i.e. the number of decisions, in which a non-existing attack is not signalled, and FN is the number of false negative outcomes, i.e. the number of decisions, in which an existing attack is not signalled.

The results concerning the effectiveness of the IDS using the Algorithm 1 are compared with those obtained with clusters labelling according to their cardinalities. The comparative results are presented in the Fig. 3. By varying the parameters Δ_{CD_1} and Δ_{CD_2}, the best results with the Algorithm 1 over the KDD CUP database are obtained with $\Delta_{CD_1} = 500$ and $\Delta_{CD_2} = 0$.

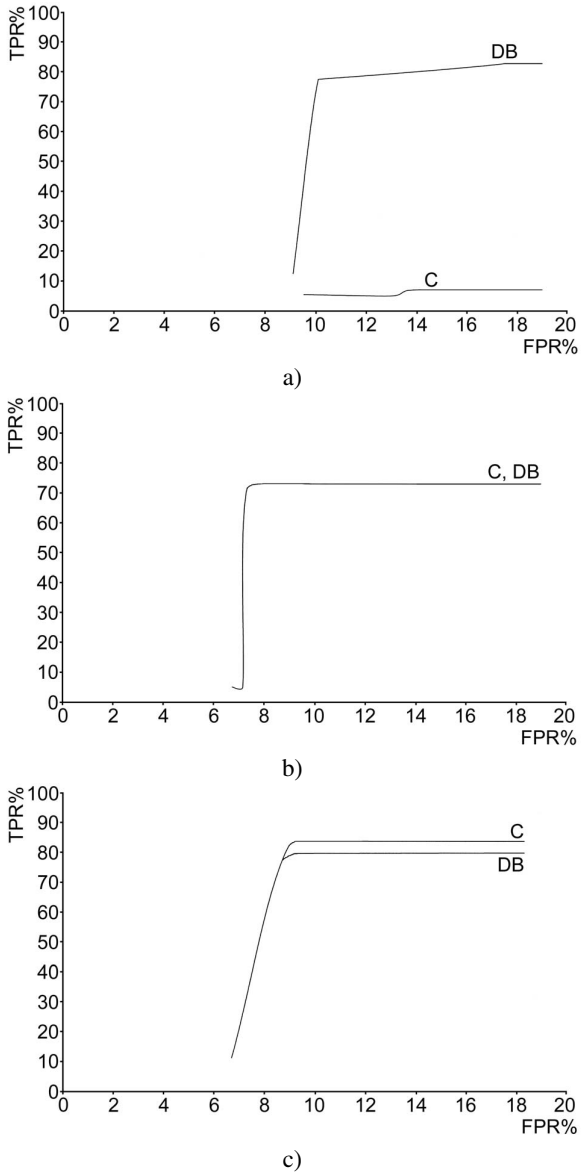

Figure 3. ROCs of the IDS. C - labelling clusters according to their cardinality; DB - labelling clusters using a combination of clustering evaluation techniques (see text). Attack filtration: a) 0%, b) 98%, c) 99%

The ROC curves labelled with DB from the Fig. 3 are obtained by setting $\Delta_{CD_1} = 500$ and $\Delta_{CD_2} = 0$ and by varying only the threshold Δ_{DB} between 0.2 and 0.45, whereas the ROC curves labelled with C are obtained by

varying the parameter $\mathcal{D}_C$ in the clusters cardinalities labelling algorithm between 500 and 0. The cardinality of the data set for clustering was $N = 1000$ in all the experiments.

From the Fig. 3, it can be observed that without attack filtering (Fig. 3a), the Algorithm 1 gives very good results, whereas the clusters labelling algorithm based on cardinalities of the clusters fails completely. With 98% of the attacks from the KDD CUP database filtered out (Fig. 3b), the behaviour of these two labelling algorithms is exactly the same, whereas the clusters labelling algorithm based on cardinalities is somewhat better than the Algorithm 1 if even more attacks (99%, Fig. 3c) are filtered out from the KDD CUP database.

The results for FPR and TPR include not only the errors caused by the labelling algorithm, but also the errors due to clustering itself. Because of this fact, the minimum values of FPR in the ROC curves from the Fig. 3 are approximately 6%. This is expected, having in mind that the IDS is based on clustering only. It is possible to improve the correctness of the decisions made by the IDS by implementing a more sophisticated clustering algorithm, but in that case the time complexity of the overall IDS algorithm would also increase, reducing the overall system efficiency.

5. Conclusion

In this paper, a new clusters labelling strategy has been proposed for application in a multiple classifier intrusion detection system (IDS). That strategy combines the computation of the Davies-Bouldin index of the clustering and the comparison of centroid diameters of the clusters. The aim of the labelling algorithm is to detect compact clusters containing very similar vectors that are highly likely to be attack vectors. By using the proposed labelling algorithm in the assessors, the response of such an IDS to a massive attack (for example, a Denial-of-Service attack) is significantly improved. In the experiments, the KDD CUP database has been used as the traffic source, in spite of all the criticism, because it is a good source of massive attacks. It has been shown experimentally, via ROC curves obtained by applying the IDS over the KDD CUP database, that the proposed labelling strategy is much better than the classical clusters labelling algorithm that uses cardinalities of clusters, when a massive attack is present. Besides, in the absence of massive attacks, i.e. with the vast majority of attacks (98% and 99%) filtered out from the database, the behaviour of the proposed assessor is similar to that of the classical assessor based on cardinalities of clusters. By selecting properly the parameters of the proposed clusters labelling algorithm, such an IDS can be adapted to the properties of any network.

References

[1] Ben Amor N., Benferhat S. and Elouedi Z., "Naive Bayes vs. Decision Trees in Intrusion Detection Systems", *Proceedings of the 19th ACM Symposium on Applied Computing (SAC2004), ACM*, Nicosia, Cyprus, 2004, pp. 420-424.

[2] Bolshakova N. and Azuaje F., "Cluster Validation Techniques for Genome Expression Data", *Signal Processing*, 83, 2003, pp. 825-833.

[3] Debar H., Dacier M., Wespi A. and Lampart S., "An Experimentation Workbench for Intrusion Detection Systems", *IBM Research Report RZ2998*, March 1998.

[4] Durst R., Champion T., Witten B., Miller E. and Spagnuolo L., "Testing and Evaluating Computer Intrusion Detection Systems", *Communications of the ACM*, Vol. 42, No. 7, 1999, pp. 53-61.

[5] Elkan C., "Results of the KDD'99 Classifier Learning", *ACM SIGKDD Explorations*, Vol. 1, No. 2, 2000, pp. 63-64.

[6] Frank J., "Artificial Intelligence and Intrusion Detection: Current and Future Directions", *Proceedings of the 17th National Computer Security Conference*, Baltimore, USA, 1994.

[7] Giacinto G. and Roli F., "Pattern Recognition for Intrusion Detection in Computer Networks", D Chen and X. Cheng (Eds.) *Pattern Recognition and String Matching*, Kluwer Academic Publishers, Dordrecht, 2002, pp. 187-209.

[8] Gomez J., Gonzalez F. and Dasgupta D., "An Imuno-Fuzzy Approach to Anomaly Detection", *Proceedings of the 12th IEEE International Conference on Fuzzy Systems (FUZZIEEE)*, Vol. 2, St. Louis, USA, May 2003, pp. 1219-1224.

[9] Guan Y., Ghorbani A. and Belacel N., "Y-Means: a Clustering Method for Intrusion Detection", *Proceedings of Canadian Conference on Electrical and Computer Engineering*, Montreal, Canada, 2003.

[10] Günter S. and Bunke H., "Validation Indices for Graph Clustering", J. Jolion, W. Kropatsch, M. Vento (Eds.) *Proceedings of the 3rd IAPR-TC15 Workshop on Graph-based Representations in Pattern Recognition*, CUEN Ed., Italy, 2001, pp. 229-238.

[11] Jain A., Murty M. and Flynn P., "Data Clustering: A Review", *ACM Computing Surveys*, Vol. 31, No. 3, 1999, pp. 264-323.

[12] Julisch K., "Clustering Intrusion Detection Alarms to Support Root Cause Analysis", *ACM Transactions on Information and System Security*, Vol. 6, No. 4, 2003, pp. 443-471.

[13] Lippman R., Haines J., Fried D., Korba J. and Das K., "The 1999 DARPA off-line intrusion detection evaluation", Computer Networks, 34, 2000, pp. 579-595.

[14] McGregor A., Braun H. and Brown J., "The NLANR Network Analysis Infrastructure", IEEE Communications Magazine, Vol. 38, No. 5, May 2000, pp. 122-128.

[15] McHugh J., "Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory", *ACM*

Trans. on Information and System Security, Vol. 3, No. 4, November 2000, pp. 262-294.

[16] Mell P., Hu V., Lippman R., Haines J. and Zissman M., "An Overview of Issues in Testing Intrusion Detection Systems", *NIST interagency report 7007*, June 2003.

[17] Ozyer T., Ahlaj L. and Barker K., "A Boosting Genetic Fuzzy Classifier for Intrusion Detection Using Data Mining Techniques for Rule Pre-Screening", *Design and Application of Hybrid Intelligent Systems*, 2003, pp. 983-992.

[18] Portnoy L., Eskin E. and Stolfo S., "Intrusion Detection with Unlabeled Data Using Clustering", *Proceedings of the ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001)*, Philadelphia, PA, November 5-8, 2001.

[19] Puketza N., Chung M., Olsson R. and Mukherjee B., "A Software Platform for Testing Intrusion Detection Systems", *IEEE Software*, Vol. 14, No. 5, 1997, pp. 43-51.

[20] Sabhnani S. and Serpen G., "Application of Machine Learning Algorithms to KDD Intrusion Detection Dataset within Misuse Detection Context", *Proceedings of the International Conference on Machine Learning, Models, Technologies and Applications (MLMTA-2003)*, Las Vegas, USA, June 2003, 209-215.

[21] Shipley G., "Intrusion Detection, Take Two", *Network Computing*, No. 15, November 1999.

[22] Shipley G. and Mueller P., "Dragon Claws its Way to the Top", *Network Computing*, No. 20, August 2001, pp. 45-67.

[23] Taylor C. and Alves-Foss J., "An Empirical Analysis of NATE - Network Analysis of Anomalous Traffic Events", *Proceedings of the 2002 workshop on New security paradigms*, Virginia Beach, Virginia, September 2002, pp. 18-26.

[24] Tölle J. and Niggemann O., "Supporting Intrusion Detection by Graph Clustering and Graph Drawing", *Proceedings of 3rd International Workshop on Recent Advances in Intrusion Detection (RAID 2000)*, Toulouse, France, 2000.