Combining Multiple Techniques for Intrusion Detection

Chaker Katar

Department of Computer Science,
Institut Supérieur de Gestion, Tunis, Tunisia

Summary

Most intrusion detection systems (IDS) are based on a single algorithm that is designed to either model the normal behaviour patterns or attack signatures in network data traffic. Most often, these systems fail to provide adequate alarm capability that reduces false positive and false negative rates. We here propose a double multiple-model approach capable of enhancing the overall performance of IDS. In a first step, every group of identical intrusion detection models are combined independently of the rest of the groups to produce a fused intrusion detection model. Then all the fused models are fused to produce the final intrusion detection model.

Our IDS model adopted three reasoning methods: Naive Bayesian, Neural Nets, and Decision Trees. We used Darpa attack taxonomy and the KDD Intrusion Detection Dataset to demonstrate the working of our IDS model.

.

Keywords: *intrusion detection system, combined detection model, fusion method.*

1. Introduction

Intrusion detection systems have became a critical component of integrated security solution for today organisations. The most used intrusion detection systems taxonomy distinguishes two main classes: misuse and anomaly detection systems. Misuse detection systems called policy-based detection systems dispose of signature-base of known attacks. When log files are analyzed, these systems trigger an alert only if analysed event sequences completely match one of the saved signatures. Knowledge-based systems reach high detection rate of known attacks. However, a small modification in actions sequences of these attacks makes them undetectable by misuse systems. Another drawback of misuse systems is their incapability to detect unknown attacks. Thus anomaly detection or behaviour-based detection systems have been designed. These systems are based on normal or expected behaviour of system or user. They generate an alarm when analyzed activity sequences deviate considerably from learned acceptable behaviour. The main shortcoming of anomaly detection systems is their high false alarm rate [7, 22]

The majority of commercial systems are generally misuse. Multiple research activities, last decade, focus on anomaly detection system trying to circumvent their shortcomings. In these, commercial systems and research prototypes, different analysis techniques have been experimented in modelling acceptable behaviour of systems or users. However, the majority of these works adopt a single algorithm either for modelling normal behaviour patterns and/or attack signatures which insures a lower detection rate and increases false negative rate. In our work, we propose the combination of analysis techniques not only to improve the overall performance of IDS but also to enhance representation of acceptable behaviour patterns and attack signatures. The proposed system will take simultaneously multiple aspects, in representing patterns or signature, which are provided each one by a single detection model.

In this work, we propose the combination of multiple techniques for intrusion detection. Multiple algorithms will be adopted in implementing our intrusion detection system. A rule based, probabilistic and non-linear models will model system normal behaviour patterns and signatures of different attack categories. Two fusion approaches, probabilistic and evidential, will be experimented in combining decisions of these detection models. In all our experiments, training and testing data sets are those of DARP 1998 IDS evaluation data [11].

Our work is organized into 4 sections. In section 2, the proposed architecture of multiple models based IDS will be presented. Selected detection models on which is based our system will be discussed in section 3. Different combination methods and those implemented for pooling decisions of detection models will be examined in section 4. A complete numerical example is given in section 5 to illustrate processing steps performed by our combined detection model over all fusion methods. In the last section, we conclude with the advantages of the proposed approach and its preliminary empirical improvements.

2. Our approach

Multiple attack taxonomies have been proposed based on different criteria. DARPA taxonomy is one of the most used. It distinguishes (defines) 4 main classes based on intruder target. Denial of service (DOS) attacks form the first class of DARPA taxonomy. These attacks make computing resources

and memory of the target system too busy thus they become unavailable and inaccessible by legitimate and authorized users. The second class focuses on user-to-root (U2R) attacks that are mounted by normal users using multiple password guessing techniques to gain super user access to the system. The third class regroups attacks mounted remotely, generally by outsider entities. Remote-to-local (R2L) attacks exploit bugs in network infrastructure to gain unauthorised access to the target machine. The last class of probe attacks allow information gathering on vulnerabilities and possible exploits supported in the target system [11].

The DARPA taxonomy was used in simulation of data sets for IDS evaluation. It will be adopted in this work. Moreover, simulated network traffic will be analyzed by our system after it is preprocessed and subdivided into three data sets according to defined feature categories. In fact, network traffic features can be grouped into three main categories: basic, content and time-based features. Basic features in logged network traffic are extracted from packet header. They provide information on intrinsic characteristics of exchanged packets such as connection duration, protocol types and flags. Content features, extracted from packet content within a connection, allow information at access level. They provide different indicators on connections status such as the number of root and access control files access, the identity of logged entity and others. Traffic features, called time-based attributes, provide different statistics in the past two seconds on similar connections that have the same host or service.

In our approach, we propose a hierarchical combination scheme for combining multiple decisions of heterogeneous intrusion detection models (Figure 1). Logged network data will be broken into three data sets according to defined feature categories. In each category, heterogeneous ID models will process the associated data set. Their decisions will be fused in the first combination level, within the same feature category. In the second level, the fused decisions by feature category will be forwarded to the final combination step or inter-categories combination in order to assign the given example to one of the four attack classes or normal class.

In the proposed architecture, a set of modelling techniques will process each data log associated with a specific features category. The set of models consists of three heterogeneous classifiers: Decision Tree (DT), Naïve Bayes (NB) and Neural Network (ANN) classifiers. The decisions of these classifiers are combined locally, within the same feature category, then with others in different categories to assign the given example to the most likely class.

3. Intrusion Detection Models

Multiple algorithms have been applied in modelling attacks signatures or expected behaviour of the system. In this work, the three following models will be adopted.

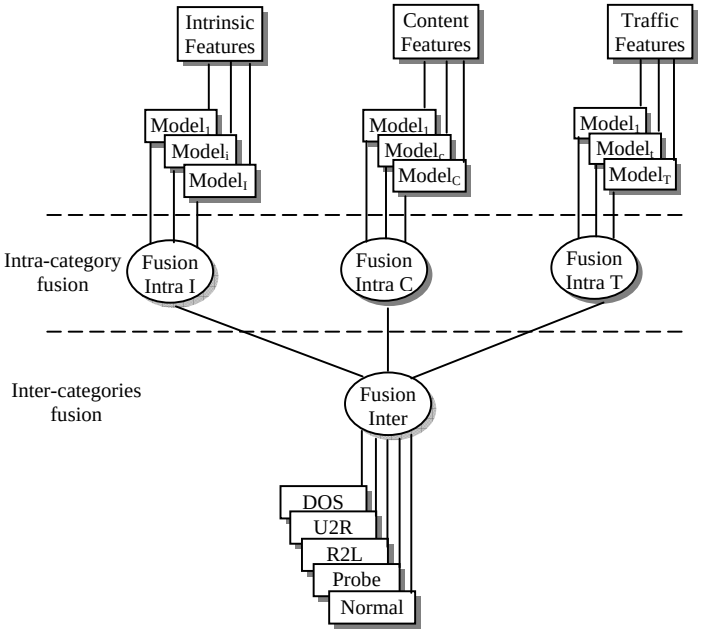

Figure 1: System Architecture

3.1. Naïve Bayes Model

Naïve Bayes is one of the most practical and most used learning methods when dealing with large amount of data as in intrusion detection. The naïve Bayes classifier simplifies learning task relying on the assumption that features are independent given the class. Moreover, it ensures an accuracy level comparable to more sophisticated classifiers (and preserves a lower computation cost and complexity than these).

Naïve Bayes classifier is based on probabilistic model for assigning the most likely class to given instance. Probabilistic model (approach) in classification field allows (model or looks for) the estimation of conditional probability of classes given instance, $p(C/A_1,\ldots, A_N)$ where $C\in\{C_1,\ldots,C_M\}$ the classes and A_i, i=1..N, a set of features describing dataset examples. Given a valued example, the most appropriate class to be assigned to is the class with the upper a posterior probability,

$$\text{Argmax}_c\ p(C=c/A_1=a_1,\ldots, A_N=a_N) \quad (1)$$

Bayesian approach splits a posterior distribution into a priori distribution and likelihood,

$$\text{Argmax}_c\ p(C=c/A_1=a_1,\ldots, A_N=a_N) = \text{Argmax}_c\ \alpha\ p(A_1=a_1,\ldots, A_N=a_N /C=c)\ p(C=c) \quad (2)$$

Where α is normalization factor to ensure that sums of conditional probabilities over class labels are equal to 1. The distribution of features given class label is more complex to estimate. Its estimation is exponential in attribute number and requires a complete training dataset with sufficient examples

for each class. Such problem can be avoided, if we assume independence of features given class, and likelihood estimation uses the following formula.

$$p(A_1=a_1,\ldots, A_N=a_N /C=c)=\Pi_i\, p(A_i=a_i /C=c) \quad (3)$$

This assumption is called Naïve Bayes assumption. It means that attributes have no joint impact within the data set of single class [10].

3.2. Neural Network Model

ANN is one of pattern recognition technique that has the capacity to adaptively model user or system behaviour. This algorithmic technique can built a useful model of user or system behaviour relying on a reduced amount of log data. Thus, it is useful for IDS where experimented hacker can sometimes alter system or applications log files to hide their mounted attacks. Moreover, ANN technique has been employed in modelling anomalous data and detection of attacks signs in intrusive data. In [3, 15] ANN was capable to autonomously learn attack signature. In addition, it is able to detect learned attacks (encountered in training data) and relying on its generalization capacity it is able to identify and learn new unseen attacks

ANN is a powerful technique for modelling complex relationship between input and output data. It consists of a network of computational units that implement a mapping function to approximate the desired output relying on a training data set. The network units or neurones are highly interconnected. Each unit receives weighted inputs to compute its activation and feeds a single output to other neurones that perform the same task. Each connection between two processing units has a weight which can be updated from iteration to another to adapt the network to the desired outputs [17].

In neural network, processing units are organized into layers. The input layer is the first layer the network structure. Neurons in this layer don't perform any task rather than feeding input data to other neuron layers. The number of neurons of this layer depends on the dimensionality of logged network traffic data. The structure of ANN disposes a single input layer which is connected to the first hidden layer of neurons and may be to other layers in specific architectures (Recurrent Neural Network). The neural network can be formed by single or more hidden layers. Processing units of this layer process input data and give their weighted outputs to neurons of either the next hidden layer or the output layer. The last is the final neuron layer in the network structure. It returns the decision of the network to the given problem. The neurons of the output layer are connected either to those of input layer or the hidden layer. Their number depends on the treated problem. It can be a single neuron when dealing with function prediction problem or multiple neurons in the case of classification problem such for intrusion detection.

3.3. Decision Tree Model

Decision tree (DT) is one of the most used machines learning technique, the last decades, in intrusion detection field. This machine learning technique builds a tree structure of attack signature using anomalous log data as in [14]. Moreover, the normal behaviour of a system or a user can be traduced in a tree structure as in [24]. The decision tree technique was applied both for misuse and anomaly detection either for network or single host [25].

The DT classifier consists of decision and leaf nodes. Each decision node corresponds to a test over a single attribute of the given instances. It has different branches on other decision or leaf nodes that represent the possible values of the actual feature. Leaf nodes represent the possible attack and normal class labels that can serves as an output when classifying a new example.

Generally, the DT classifier is generated relying on two phase's process. The dimensionality reduction is the first phase in DT building process. In this phase, the appropriate decision nodes are selected. This phase is required in every learning problem and it aims at reducing the complexity of learning process and optimizing the decision process of the learner when dealing with high dimensionality feature space. Multiple techniques can be used to extract relevant features to the actual learning task (selection criterion, GA). In the literature, information gain measure is one of the most used selection criterion [9, 25, 28]. It evaluates the effectiveness of an attribute in classifying training examples and serves for ranking features according their computed relevance. In the second phase, the most relevant feature is taken as root node of the tree structure. The braches of this node are defined from training examples. Each branch defines a new sub-tree. The root node of the sub-tree is selected from the remaining set of feature, of the first phase. Moreover, it should be less relevant than the tree root node. This process is repetitively performed for all selected attributes, with respect to their relevance, until connecting branches on leaf nodes. Then, the train data examples are completely processed and the DT classifier is generated.

Test data examples are classified by DT starting at the root node. The value of the root feature is tested and the convenient branch leads to other nodes is selected. By moving down to next root node of the new sub-tree, the same decision process is recursively performed until branching on a leaf node. The last is considered the most appropriate class associated to the given example.

In our work, C4.5 DT induction algorithm is used to generate classifier both for normal and intrusive data sets. C4.5 of Quilan is based on information gain as a feature selection criterion. The information gain ratio of C4.5 allows the selection of the feature with maximal information gain and

minimal partitioning (minimal information split) at each level of tree building. The C4.5 algorithm can deal with data sets that have examples with missing values in specific attributes. Moreover, it can process features with continuous values such data size and connection duration attributes [26].

In our experiments, we will use weak 3.4.4 a java implementation of machine learning tools. Weka supports J48 an improved implementation of C4.5 DT learner called release 8. NB detection model adopted for our experiments uses the kernel density estimator rather than normal distributions for numeric attributes. Numeric estimator precision values are estimated using training data sets [8, 26]. Moreover, Weka implementation of BPNN algorithm with 500 epochs (iterations for each data fold), .2 learning rate and variable hidden layer node numbers (7-9-12) was used. In experiments, detection models were trained with 10-folded cross validation [26, 27].

4. Combination approaches

Different combination methods have been presented in [28, 20]. They can be classified into three types based on base models outputs. Output information of base classifier can be assigned to one of the three levels: abstract, rank and measurement. Type I classifier outputs abstract information that is the most probable class label for the input. Output information of type II classifier is a partial or complete ranked list of class labels. The most likely output class of this classifier is the top of the list. Type III classifier allow soft outputs that give its confidence on each class for the given input.
Methods such as majority voting and Behaviour-Knowledge space allow fusion of type I classifiers outputs. Combination methods of type II classifiers outputs are based either on reduction or reordering approaches. They aim at improving the rank of the true class of the given input either by reducing or resorting class labels over all lists. The largest class of combination methods focus on classifiers output information at the measurement level. They thought of returned confidence values by each classifier as probability, possibility or belief measures the can reduce uncertainty level of the combined decision. In this work, both methods of first and third classes are used.

4.1. Bayesian fusion

Bayesian approach has been extensively studied and already applied in decision fusion. Bayes combination rule computes probabilities of hypotheses using evidences provided by classifiers, simultaneously. It allows the computation of posterior probabilities of hypotheses using both prior and conditional probabilities

Consider that we dispose of $p(C_1)$, the prior probability of an attack class C_1. At a given point, we obtain more knowledge in form of piece of evidence E that informs us on the state of the network. So it is more appropriate to express the new belief on C_1 using conditional probability. According to Bayes theorem [5, 23]

$$p(C_1/E)=p(C_1,E)/p(E)$$
$$= [p(E/C_1)p(C_1)] / [\Sigma_j\, p(E/C_j)p(C_j)] \quad (4)$$

If we have multiple evidences $E_1,\ldots, E_K$, the posterior of C_1 became:

$$p(C_1/ E_1,\ldots, E_K)=$$
$$[p(E_1,\ldots, E_K /C_1)p(C_1)] /\ p(E_1,\ldots, E_K) \quad (5)$$

If all evidences are independent

$$p(C_1/ E_1,\ldots, E_K) = [p(E_1 /C_1)\, p(E_k /C_1)..\, p(E_K /C_1)\, p(C_1)] / [\Sigma_j\, p(E_1 /C_j)\, p(E_k /C_j)..\, p(E_K /C_j)] \quad (6)$$

This posterior probability collects all evidences of different classifiers and integrates their impacts on the given hypothesis for making the final decision. The last is based on Bayes decision rule defined by the following equation

$$p(C/ E_1,\ldots, E_K) = \max_j p(C_j/ E_1,\ldots, E_K) \quad (7)$$

Bayes decision rule stated that the final decision of the most probable attack for the given example is the class to which is associated the grater posterior probability.
In Bayes combination scheme, the decisions of classifiers are considered statistically independent. Moreover, the set of attack classes $\{C_j\}$, j=1..M, is supposed composed by mutually exclusive and exhaustive classes. And before performing combination, each class should have a priori probability, $p(C_j)$. In addition, Bayesian combination scheme does not provide any information neither on the quality of computed probability nor on the existence of conflicting evidences that can influence Bayes decision criterion [12, 13, 23].

Bayesian fusion methods used in this work are based on average and product rules. The average rule [28] computes the posterior probability of combined decision based only on confidence values returned by each classifier. However, product rule takes into account prior probability of each class in estimation of combined evidences support. The prior probability of each class is, generally, estimated from training data. Another variant of the product rule which incorporates both prior probability and information on feature categories in evidences fusion [16] is also considered.

4.2 Evidential fusion

The mathematical theory of evidence is a generalisation of probability theory to simply and directly represent ignorance.

The Dempster-Schafer theory (DST) of evidence is a powerful tool for representing knowledge, updating beliefs and combining evidences relying on Dempster's combination rule. Thus, it becomes attractive for modelling complex systems and practical for multiple applications in different domains such as classification, information fusion, medical diagnosis and others.

DST is based on Ω the frame of discernment. It is a set of mutually exclusive and exhaustive hypotheses $\Omega=\{C_1,\ldots, C_M\}$. All possible subsets ($C\subseteq\Omega$) of Ω are also hypotheses and they form superset of 2^M. The impact of evidence or a subset of the power set can be measured by the mass function or the basic probability assignment (BPA). BPA is a mapping function of the power set to the interval [0,1]. Formally, its prosperities are the following:

$$m: 2^M \rightarrow [0,1]$$
$$m(\varnothing)=0 \text{ and } \Sigma_{C\subseteq\Omega}\, m(C)=1$$

The evidences with not null mass are called focal elements. They represent the only elements in Ω taken into account in computing belief values. The belief function is based on mass function to evaluate the total belief committed to a given hypothesis C via its all subsets as given by following formula

$$Bel(C)= \Sigma_{B\subseteq C}\, m(B) \qquad (8)$$

The plausibility relies also on BPA. It is the sum of all masses associated to a subset B that intersect with C

$$Pl(C)= \Sigma_{B\cap C\neq\varnothing}\, m(B) \qquad (9)$$

Bel and Pl represent respectively the lower and upper bound that locate the probable impact of evidence on the hypothesis C. They fix respectively the minimum and the maximum extents to which current evidence allows to belief C [1, 2, 21]

Dempster's combination rule

Dempster's rule allows the pooling of two or more independent evidences within the same frame of discernment and from different sources into a single belief function that expresses the support of the proposition in both evidences bodies.

Consider Bel_1 and Bel_2 two belief function and m_1 and m_2 their respective BPA associated to independent evidences defined in the same frame Ω. The combined BPA that represents the aggregated impact of different pieces of evidences on the hypothesis is defined as follow

$$\forall\, C\subseteq\Omega\ m(C)= m_1 \oplus m_2(C)$$
$$= K\, \Sigma_{(A,B\subseteq\Omega;\, A\cap B=C)}\, m_1(A)\, m_2(B) \qquad (10)$$

Where

$$K=1/(1-\Sigma_{(A\cap B=\varnothing)}\, m_1(A)\, m_2(B))$$

K is a coefficient of normalisation. It expresses the degree of agreement between sources. If it is null, it means the complete conflict between sources and the combination is impossible [1, 9, 6]

The corresponding belief function $Bel(C)= Bel_1 \oplus Bel_2(C)$ can be computed using the combined masses by (10) and equation (8).

DST is useful when dealing with incomplete and possibly contradictory information. it does not require a priori knowledge on probability distribution of attack classes for performing evidence combination as in Bayesian scheme. However, DS combination scheme is similar to Bayesian scheme in that evidences are assumed to be statistically independent [23]

Selected Combination Methods

RSR method

Xu et al. evidential combination method is based on detection model global information. Recognition, Substitution and Rejection rates (RSR) of attack classes and normal behaviour are used in this method [28]. These two measures are computed using confusion matrix of each detection model in testing phase. They will serve in computing belief mass (m_k) of each hypothesis for each detection model.

In this method, the detection model (e_k) decision for each given example x will be

$$e_k(x)=C_j^k$$

where $e_k \in \{ e_1, \ldots, e_K \}$, the set of detection models .

- $C_j^k = C_{M+1}$, x is not recognised by the classifier e_k, in this case we have a single focal element or it is the complete ignorance case and $m(\Omega)=1$
- $C_j^k \in \Omega$, we have two focal elements (C_j^k and $\neg C_j^k = \Omega - C_j^k$) and

$$m_k(C_j^k)=r_j^k$$
$$m_k(\neg C_j^k)= s_j^k$$
$$m_k(\Omega)=1- r_j^k - s_j^k \qquad (11)$$

Where are r_j^k and s_j^k are respectively recognition and substitution rates of detection model k and class j. The BPA of K detection models decisions will be fused using the orthogonal combination rule to assign of the given instance to the most appropriate class [13, 18, 4].

Predictive rate method (PRM)

As in Xu et al's method, Parikh et al. combination scheme is based on classifier level information. The predictive rates instead of recognition, substitution and rejection rates are used in hypotheses masses estimation. Belief masses are estimated for PRM method using the confusion matrix of each classifier. The predictive rate of each class takes into account misclassified instances of other classes. It measures to which extent the detection model can recognise this class.

The predictive rate of class j, p_j^k, is used as BPA in Parikh et al PRM method when detection model k outputs C_j and the given example x is not rejected. The disbelief on C_j is $(1- p_j^k)$ if it is not the correct class of the given example according to the detection model k [18]

Class Level method
Rogova combination method is based on class level information. This method is based on a distance measure to estimate belief on hypothesis C_j for detection model k. For each model k and class C_j, it computes the reference vector R_j^k that characterizes them form class specific training data set. The distance measure $d_j^k = \Phi(y_k, R_j^k)$ is computed between the output of the classifier k ($e_k(x) = y^k$, y^k is a vector of confidence values, one for each class) and the reference vector of class j, R_j^k. The distance d_j^k serves to estimate BPA per-class-per-classifier as follow

$$m_j^k(C_j)= d_j^k$$
$$m_j^k(\Omega)= 1- m_j^k(C_j) \qquad (12)$$

To combine belief on different hypothesis, Rogova's method takes into account classifiers votes not pro-hypothesis C_j , the disbelief on C_j is:

$$m_{\neg j}^k(\neg C_j)=1-\Pi_{i\neq j}\, d_i^k$$
$$m_{\neg j}^k(\neg \Omega)= 1- m_{\neg j}^k(\neg C_j) \quad (13)$$

Both BPAs $m_j^k(C_j)$ and $m_{\neg j}^k(\neg C_j)$ computed for all classifiers are combined using Dempster orthogonal sum rule to provide the final belief on each hypothesis ($Bel_j=m(C_j)$ **,** C_j is an atomic hypothesis). Rogova has given a simplified formula to compute belief masse of each hypothesis, formula (14). The combined belief on each hypothesis in Ω is computed using (15). The hypothesis with the max belief is taken as the appropriate decision of combined detection models [19, 4].

$$m_m(C_j)=1- \Pi_i\, d_j^i \quad \text{if } j = m$$
$$m_m(C_j)=0 \quad \text{if } j \neq m$$
$$m_m(\Omega)= 1- \Pi_i\, (1-d_j^i) \qquad (14)$$

$$m(C_j)=[p_j\, \Pi_{i\neq j}\, (1- p_i)]/[\Sigma_j\, p_j\, \Pi_{i\neq j}\, (1- p_i)+ \Pi_i\, (1- p_i)] \quad (15)$$

where $pj=m_j(C_j)$.

5. Illustrative example

Logged network traffic records processed by our combined model are valued vector of 41 features illustrated by examples in figure 2. The explanation and complete list of features used in these examples can be found in [11].

Each record in training or testing data sets is broken into three fragments according to defined feature categories as depicted by following sample records. The data sets are then given to the appropriate set of models**.**

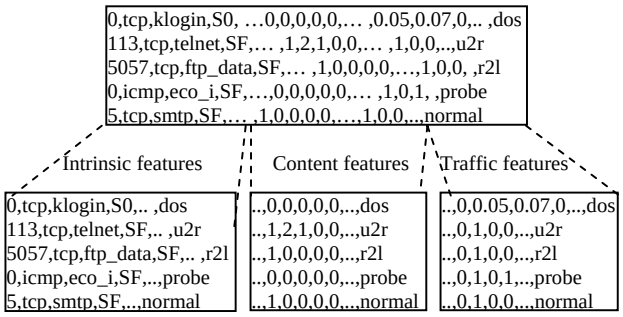

Figure 2: Sample data sets for feature categories

In training phase, each detection model is trained with normal and intrusive data set using 10-fold cross validation. On each training fold generated models is revised and updated then it is tested on remaining folds. A sample rule-based model (RBM) built by our DT detection model using traffic features is given in figure 3.

1. (dst_host_diff_srv_rate <= 0.61 ∧ dst_host_srv_diff_host_rate <= 0.4 ∧ srv_count <= 2 ∧ dst_host_srv_count <= 4 ∧ dst_host_serror_rate <= 0.1 ∧ dst_host_same_src_port_rate <= 0.75 ∧ count <= 1) ⇒ u2r (19.0/4.0)
2. (dst_host_diff_srv_rate <= 0.61 ∧ count <= 2 ∧ dst_host_srv_diff_host_rate <= 0.4 ∧ dst_host_srv_count > 86 ∧ dst_host_srv_diff_host_rate > 0) ⇒ normal (239.0)
3. (dst_host_diff_srv_rate <= 0.61 ∧ count <= 2 ∧ dst_host_srv_diff_host_rate <= 0.4 ∧ dst_host_srv_count > 86 ∧ dst_host_srv_diff_host_rate <= 0) ⇒ dos (62.0)

Figure 3: Sample rules generated for attacks and normal behaviour using traffic feature category

Each rule has one or two numbers after the output class or the consequent part that serve in computing the support. These numbers define respectively the correctly and incorrectly classified instances by this rule.
The probabilistic model (PM) is based on distribution functions of features. Therefore, in training phase attributes distributions are estimated using training examples. For instance, NB models allows following probabilistic models for attack classes and normal behaviour of the system.

Class dos: Prior probability = 0.37
count: Normal Distribution. Mean = 168.0187 StandardDev = 211.7242
WeightSum = 875…..
Class u2r: Prior probability = 0.01
count: Normal Distribution. Mean = 0.6589 StandardDev = 1.2327
WeightSum = 27…
Class normal: Prior probability = 0.22
count: Normal Distribution. Mean = 3.8501 StandardDev = 11.0605
WeightSum = 583 ….

Figure 4: Probabilistic model for attacks and normal behaviour using traffic features

In this model, DOS attack prior probability is estimated from training data set. For this attack, the mean and standard deviation of each continuous attribute are estimated using 875 data examples. The normal distributions of features and prior probabilities of classes are used in conjunction with formula (1, 6) to assign each given example to the most likely output class.

Nonlinear model (NM) of attacks and system normal behaviour uses back propagation neural network (BPNN) with five output neurones and variable node numbers in input and hidden layers. The number of features in each category, intrinsic, content and traffic corresponds to the input nodes in first layer of each BPNN. The BPNNs hidden layers for three categories have respectively 7, 9 and 12 neurons. The nodes in these neural networks are all sigmoid. Parameters of all the three BPNNs were initialized with 500 epochs, .3 learning rate and .2 momentum.

To test detection models, we take the example labelled with U2R from network traffic sample records given before (figure 2). The decision of each model of a specific feature category is confidence vector each value for an output class. The table 1 present decisions of base detection models associated to three feature categories for selected example, the right class for each model is in bold character. Following tables illustrate how implemented combination methods compute probabilities or beliefs on output hypotheses. These methods don't care about base model that allows confidence values of 1.0 or 0 for a given output in their computation steps.

Table 1: Outputs of base models for selected example on three feature categories

Feature categories/ Model		DOS	U2R	R2L	Probe	Normal
Intrinsic Feature Category (IC)	RBM	**0.4789**	0.0133	0.1537	0.0001	0.3539
	PM	3.99E-5	0.0315	1.94E-4	1.39E-5	**0.9681**
	NM	2.7E-4	8.3E-5	0.0045	0.0024	**0.9926**
Content Feature Category (CC)	RBM	**0.9760**	0.0198	0.0001	0.0001	0.0038
	PM	0.0019	**0.998**	5.7E-75	6.E-151	4.11E-7
	NM	9.21E-7	**0.9998**	7.62E-5	4.99E-5	4.66E-7
Traffic Feature Category (TC)	RBM	0.1764	0.2352	**0.4705**	0.0001	0.1176
	PM	1.4E-13	**0.9999**	7.8E-5	3.9E-19	2.8E-10
	NM	2.09E-4	0.0825	**0.9171**	1.2E-4	7.6E-6

Bayesian combination

Bayes product and average rule require both posterior probabilities computed by each detection model. In addition, product rule uses prior probabilities of classes that are computed from training data set. The second variant of product rule incorporates information on feature categories in combination. Therefore, we will consider the number of feature categories in each combination level. The combined decisions of intrusion detection models relying on Bayesian rules and using classes' prior probabilities (table 2) are illustrated by table 3.

Table 2: Output classes prior probabilities

Class	DOS (D)	U2R (U)	R2L (R)	Probe (P)	Normal (N)
Prior	0.37	0.01	0.1	0.3	0.22

Table 3: Combined models decisions using Bayes rules

Bayesian Fusion	Class	First level fusion by category			Second level fusion
		IC	CC	TC	
Average rule	D	0.1597	0.3260	0.0588	0.1815
	U	0.015	**0.6726**	0.4392	**0.3756**
	R	0.0528	2.5E-5	**0.4626**	0.1718
	P	8.1E-4	1.6E-5	4.1E-5	2.9E-4
	N	**0.7715**	0.0013	0.0392	0.2707
Product rule	D	2.5E-8	2.4E-6	7E-15	1.8E-18
	U	6.4E-9	**0.9999**	**0.9888**	**0.9999**
	R	1.6E-7	1E-80	0.0111	2.5E-80
	P	1E-11	3E-156	5E-24	1E-180
	N	**0.9999**	6.1E-13	2E-13	3.1E-16
Product rule modified	D	1.5E-8	8.9E-8	2E-16	4.9E-27
	U	1.0E-7	**0.9999**	**0.9982**	**0.9999**
	R	4.0E-7	2.1E-81	0.0017	3.5E-85
	P	1E-11	1E-157	2E-25	8E-189
	N	**0.9999**	3.8E-14	1.3E-14	1.9E-23

As an example, the probabilities of U2R attacks are computed by the three Bayesian rules as follow; we present an example of decisions fusion within Traffic feature category (TC) and then we perform the second combination step over all categories.

Post-probabilities associated to U2R attacks outputted by base models (table 1):

RBM: $p_{1TC}(U)= 0.2352$
PM: $p_{2TC}(U)= 0.9999$
NM: $p_{3TC}(U)= 0.0825$
Prior probability: $p_r(U)= 0.01$

-**Average rule:**

- First level fusion: $p_{TC}(U)= (p_{1TC}(U)+ p_{2TC}(U)+ p_{3TC}(U)) / 3 = 0.4932$
- Second level fusion: $p(U)= (p_{IC}(U)+ p_{CC}(U)+ p_{TC}(U)) / 3 = 0.3756$

-**Product rule**:

- First level fusion: $p_{TC}(U)= \alpha\, p_{1TC}(U)\, p_{2TC}(U)\, p_{3TC}(U)\, p_r(U) = 0.998$; (the normalization factor α=5.09E+3)
- Second level fusion: $p(U)= \alpha\, p_{IC}(U)\, p_{CC}(U)\, p_{TC}(U)\, p_r(U) = 0.999$

- **Product rule modified**:

- First level fusion: $p_{TC}(U)= \alpha\, p_{1TC}(U)\, p_{2TC}(U)\, p_{3TC}(U) = 0.999$;
- Second level fusion: $p(U)= \alpha\, p_{IC}(U)\, p_{CC}(U)\, p_{TC}(U)\, p_r(U)^{-2} = 0.999$

DS combination

DS combination methods adopted in this work use Dempster's orthogonal rule and suppose a normalized BPA. Methods of first class require confusion or contingency matrices of detection models for computing recognition and substitution rates and predictive rate respectively for RSR and PRM methods, respectively. The second class's method needs an extended training data set for each class to compute meaningful reference vectors. When sufficient data is unavailable, this is true for U2R attacks, confidence values that range in [0, 1] interval returned by each detection model can serve as belief masses [4].

The methods of first class require a belief masse of each hypothesis in Ω. Belief masses must be computed for each base model and combined models within the feature category (fist level combination). They are computed using confusion matrices outputted by models when tested on validation data set. Validation data set used for generating confusion matrices are composed by 100 examples for each output class. For instance, confusion matrix for rule-based model on traffic features is in following table (table 4). It states that 59 over 100 instances of DOS attack in validation data were correctly classified using rule-based model generated for traffic feature category.

Table 4: Rule-based model on traffic features category confusion matrix

	DOS	U2R	R2L	Probe	Normal
DOS	59	2	6	4	29
U2R	9	32	6	0	53
R2L	7	1	39	0	53
Probe	0	1	0	68	31
Normal	21	2	0	0	77

Belief masses for RSR and PRM methods are computed using confusion matrices for all base models and first level combined models. Computed belief masses for DOS attacks using confusion matrix of table 4 are as follow.

-Xu et al. method belief masses computation for DOS attacks and all other classes uses (11):

$m_{1TC}(DOS) = 59/100 = 0.59$ (Recognition rate)

$m_{1TC}(\neg DOS) = (2 + 6 + 4 + 29)/100 = 0.41$ (Substitution rate)

$m_{1TC}(\Omega) = 1-(59+61)/100 = 0$ (Rejection rate)

- PRM method belief masses computation for DOS attacks:

$m_{1TC}(DOS) = 59/(59+9+7+21) = 0.6146$ (Predictive rate)

$m_{1TC}(\neg DOS) = 1-0.614 = 0.3854$

Combined detection models decisions using the three evidential fusion methods are illustrated by following table (table 5).

Base models (RBM, PM and NM) outputs for selected example on traffic feature category were respectively classes R2L, U2R and R2L. To combine their decisions within the same feature category combined masses of same output

Table 5: Combined decisions using DS methods

DS Fusion	Class	First level fusion by category			Second level fusion
		IC	CC	TC	
RSR method	D	**0.826**	5.0E-4	0	**0.9998**
	U	0	**0.9995**	**0.713**	2.0E-4
	R	0	0	0.287	0
	P	0	0	0	0
	N	0.1754	0	0	0
PRM method	D	**0.6914**	1.0E-4	0	0.0569
	U	0	**0.9999**	0.0836	**0.9165**
	R	0	0	**0.9164**	0.0266
	P	0	0	0	0
	N	0.3086	0	0	0
Rogova's method	D	0.1997	0.4874	0.0507	0.2305
	U	0.0137	**0.5115**	**0.4763**	**0.3034**
	R	0.0519	2.0E-5	0.4400	0.1652
	P	7.0E-4	1.0E-5	3.0E-5	2.0E-4
	N	**0.734**	0.0010	0.0327	0.3007

hypothesis over three base models are computed in first step using formula (10).Then they are combined in second step with others with different hypotheses. The same fusion process is performed for second level combination. A complete illustration of this process for three methods is given bellow; belief masses computed for PM and NM on traffic category and used in this numerical example can computed form models confusion matrices.

-**RSR method computation**:

-First level fusion: Base models belief masses:

RBM: $m_{1TC}(R)= 0.39$, $m_{1TC}(\neg R)= 0.61$

PM: $m_{2TC}(U)= 0.45$, $m_{2TC}(\neg U)= 0.55$

NM: $m_{3TC}(R)= 0.34$, $m_{3TC}(\neg R)= 0.66$

– Combine masses with same hypothesis

$K=1/ (1- m_{1TC}(R)\, m_{3TC}(\neg R)- m_{3TC}(R)\, m_{1TC}(\neg R)) = 1.864$

$m_{13TC}(R)= m_{1TC}(R)\, m_{3TC}(R)\, K = 0.2478$

$m_{13TC}(\neg R)= m_{1TC}(\neg R)\, m_{3TC}(\neg R)\, K = 0.7522$

– Combine masses with different hypotheses after normalization

$K=1/(1- m_{2TC}(U)\, m_{13TC}(R))=1.1255$

$m_{TC}(U)= m_{2TC}(U)\, m_{13TC}(\neg R)\ 1.87K=$ **0.713**

$m_{TC}(R)= m_{13TC}(R)\, m_{2TC}(\neg U)\ 1.87K= 0.287$

Combined models within traffic feature category output class U2R as the right class for the given example according to fused masses.

-Second level fusion: combined models (CM) of first level have selected respectively DOS, U2R and U2R classes for taken example. Their belief masses on outputted hypotheses are (belief masses for the CM used by RSR method can be computed using their confusion matrices as for base models):

CM intrinsic category: $m_{IC}(D)= 0.9999$, $m_{IC}(\neg D)= 0.0001$

CM content category: $m_{CC}(U)= 0.23$, $m_{CC}(\neg U)= 0.77$
CM traffic category: $m_{TC}(U)= 0.06$, $m_{TC}(\neg U)= 0.94$

- Combine masses with the same hypothesis
  $K=1/ (1- m_{CC}(U)\, m_{TC}(\neg U)- m_{TC}(U)\, m_{CC}(\neg U)) =1.3557$
  $m_{CCTC}(U)= m_{TC}(U)\ m_{CC}(U)\ K = 0.0187$
  $m_{CCTC}(\neg U)= m_{TC}(\neg U)\ m_{3TC}(\neg U)\ K = 0.9813$
- Combine masses with different hypotheses after normalization
  $K=1/(1- m_{CCTC}(U)\, m_{IC}(D))=1.0001$
  $m(U)= m_{CCTC}(U)\, m_{IC}(\neg D)\ 1.87K= 1.9E\text{-}4$
  $m(D)= m_{CCTC}(\neg U)^* m_{IC}(D)\ 1.87K= 0.9998$

Combined models within traffic feature category output class DOS as the right class for the given example according to fused masses.

-PRM method

-First level fusion: Base models belief masses:

Base models belief masses:
RBM: $m_{1TC}(R)= 0.7647$, $m_{1TC}(\neg R)= 0.2353$
PM: $m_{2TC}(U)= 0.4369$, $m_{2TC}(\neg U)= 0.5631$
NM: $m_{3TC}(R)= 0.7234$, $m_{3TC}(\neg R)= 0.2766$

- Combine masses with the same hypothesis
  $K=1.6174$
  $m_{13TC}(R)= 0.8947$
  $m_{13TC}(\neg R)= 0.1053$
- Combine masses with different hypotheses after normalization
  $K=1.642$
  $m_{TC}(U)= 0.0836$
  $m_{TC}(R)= 0.9164$

-Second level fusion: CM belief masses on selected output classes (complete belief masses for CM used by PRM method are computed as for base models)

CM intrinsic category: $m_{IC}(D)= 0.588$, $m_{IC}(\neg D)= 0.412$
CM content category: $m_{CC}(U)= 0.9583$, $m_{CC}(\neg U)= 0.0417$
CM traffic category: $m_{TC}(R)=0.4$, $m_{TC}(\neg R)=0.6$

- Combine masses with different hypotheses after normalization
  $K= 3.722$
  $m(D)= 0.0569$
  $m(U)=$ **0.9165**
  $m(R)= 0.0266$

Combined models decision over all feature categories is the U2R class, the correct class of processed example.

-Rogova's method

Rogova's method is based on mass computation process different than the used by RSR and PRM methods. The proposed process for this method uses confidence values returned by base models to compute belief mass of each hypothesis. Using confidence values of table 1, belief mass for U2R attack is computed by this method using (14) as follow:

$p_{uTC}(U)=1-(1-0.2352)\ (1-0.9999)\ (1-0.0825)\ 0.44=0.4443$; (0.44, the normalization factor)

In first and second combination levels formulas (14, 15) are used to compute beliefs on hypotheses

- First level fusion:
  $m_{TC}(D)= 0.0507$
  $m_{TC}(U)= 0.4763$
  $m_{TC}(R)= 0.4400$
  $m_{TC}(P)= 3.0E\text{-}5$
  $m_{TC}(N)= 0.0327$

- Second level fusion: combined beliefs masses within the feature category in first level are fused with others of different categories in the second step:
  $m_{TC}(D)= 0.2305$
  $m_{TC}(U)=$ **0.3034**
  $m_{TC}(R)= 0.1652$
  $m_{TC}(P)= 2.0E\text{-}4$
  $m_{TC}(N)= 0.3007$

6. Conclusion

Multiple empirical studies and specific machine learning and pattern recognition applications have confirmed that even if a given model outperforms others in specific problem it is incapable to reach the best results on the overall problem domain. It is the case in intrusion detection field because often single algorithm can't deal with all attack classes at the desired accuracy level. Thus, combination of multiple models tries to take advantage of the local different behaviour of the base model to improve overall performance of IDS system. Moreover, it enforces the system error recovery mechanism when single model fails in predicating the right class of attack and increases the opportunities of IDS to detect difficult attacks such as those of U2R and R2L classes. This was approved empirically by our combined model for intrusion detection that has increased detection rates of rare attacks and the overall system respectively by nearly 6% and 15%. Therefore, we will explore in our future works the capabilities of such model in detecting different attacks stages.

References

[1] Al-Ani, A. and M. Deriche. A New Technique for Combining Multiple Classifiers using The Dempster-Shafer Theory of Evidence. Journal of Artificial Intelligence Research 17, p. 333-361, 2002.

[2] Aslandogan, Y. A. and G. A. Mahajani and S. Taylor. Evidence Combination in Medical Data Mining. IEEE International Conference on Information Technology. Coding and Computing. LasVegas, NV, April 2004.

[3] Canndy J., Applying CMAC-Based On-line Learning to Intrusion Detection. Proceedings of the 2000 IEEE International Joint Conference on Neural Networks, July 2000.

[4] Catalin, I. T. and Sargur N. Srihari, Combination of Type III Digit Recognizers using the Dempster-Shafer Theory of Evidence. Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003) 0-7695-1960-1/03, 2003.

[5] Chen, B. and P. K. Varshney, A Bayesian Sampling Approach to Decision Fusion Using Hierarchical Models. IEEE Transactions on Signal Processing, Vol. 50 No. 8. August 2002.

[6] Chen, K., L. Wang and H. Chi, Methods of Combining Multiple Classifiers with Different Features and Their Applications to Text-Independent Speaker Identification. International Journal of Pattern Recognition and Artificial Intelligence, 11(3), pp. 417-445, 1997.

[7] Debar, H., M. Dacier, and A.Wespi. A Revised Taxonomy for Intrusion Detection Systems. Annals of Telecommunications, 55(7–6):361–378, July–August 2000.

[8] Diplaris, S., G. Tsoumakas, P. A. Mitkas, and I. Vlahavas, Protein Classification with Multiple Algorithms. 10th Panhelllenic Conference on Informatics (PCI 2005), P. Bozanis and E.N. Houstis (Eds.), Spring-Verlag, LNCS 3746, pp. 448-456, Volos, Greece, 11-13/11, 2005.

[9] Elouedi, Z., K. Mellouli, P. Smets, Decision Trees Using the Belief Function Theory. Proceedings of the International Conference on Information, Processing and Management of Uncertainty, IPMU'2000, 2000.

[10] Flach, P. A. and N. Lachiche, Naive Bayesian Classification of Structured Data. Kluwer Academic Publishers, Boston. Manufactured in The Netherlands, Machine Learning, 1–37, 2004.

[11] KDD cup99, preprocessed DARPA 1998 evaluation data sets available online at : http://kdd.ics.uci.edu/databases/kddcup99

[12] Kittler, J., A Framework for Classifier Fusion: Is it still needed?. Proceedings of SSPR'00, p.45-56, 2000.

[13] Kittler, J., Combining Classifiers: A Theoretical Framework. Pattern Analysis and Applications, 18-27/1, 1998.

[14] Kruegel, C. and T. Toth, Using Decision Tree to Improve Signature Based Intrusion Detection. 6th Symposium on Recent Advances in Intrusion Detection (REID), Lecture Notes in Computer Science, Spring Verlag, USA, September 2003.

[15] Kumar, G.P. and P.Venkaterm, Security Management Architecture for Access Control to Network Resources. IEEE Proceedings on Computer and Digital Techniques Vol.144-6, p.362-370, Nov 1997.

[16] Kuncheva, L., On Combining Multiple Classifiers. Proceeding 7th International Conference of Information Processing and Management of Uncertainty (IPMU'98), Paris, France 1998.

[17] Novokhodko, A., A Survey of the Applications of Neural Networks to Intrusion Detection", Lab. Of Applied Computational Intelligence, Dep. Of Elect. And Computer Engineering Univercity of Missouri-Rolla, 2001.

[18] Parikh, C.R., M.J. Pont and N.B. Jones, Application of Dempster-Shafer Theory in Condition Monitoring Systems: A case study. Pattern Recognition Letters, 22 (6-7): 777-785, 2001

[19] Rogova, G., Learning in Distributed Systems for Decision Making. Center for Multi-sources Information Fusion, State University of New York at Buffalo 421 Bell Hall Buffalo, NY 14260, Final Technical Report, 1998.

[20] Ruta, D. and B. Gabrys, An Overview of Classifier Fusion Methods. Computing and Information Systems, 7, p.1-10, 2000.

[21] Ruthven, I. and M. Lalmas, Using Dempster-Shafer's Theory of Evidence to Combine Aspects of Information Use. Journal of Intelligent Information System, v.19, n.3, p. 267-301, 2002.

[22] Sabhnani, M. and G. Serpen, Application of Machine Learning Algorithms to KDD Intrusion Detection Dataset within Misuse Detection Context. Proceeding of MLMTA'03, p.209-215, 2003.

[23] Siaterlis, C. and B. Maglaris, Towards Multisensor Data Fusion for DoS Detection. SAC '04, March 14-17, Nicosia, Cyprus, 2004.

[24] Sinclair, L. P. and S. Matzner, An Application of Machine Learning to Network Intrusion Detection. ACSAC'99, p.371-377, 1999.

[25] Stein, G., B. Chen and A. S. Wu, K. A. Hua, Decision Tree Classifier For Network Intrusion Detection With GA-based Feature Selection. Proceedings of the 43rd ACM Southeast Conference, Kennesaw, GA, March 18-20, 2005.

[26] Twa, M. D., S. Parthasarathy, M. A. Bullimore, C. Roberts, A. M. Mahmoud, T. Raasch, and D. J. Schanzlin, Automated decision tree classification of keratoconus from Videokeratography. Investigative Ophthalmology and Vision Science, E-Abstract 1082 46, ARVO, 2005.

[27] Witten, I. H. and E. Frank, WEKA: Machine Learning Algorithms in Java. This tutorial is Chapter 8 of the book Data Mining: Practical Machine Learning, Tools and Techniques with Java Implementations. Morgan Kaufmann, 2000.

[28] Xu, L., A. Krzyzak and C.Y. Suen, Methods of Combining Multiple Classifiers and their Applications to Handwriting Recognition. IEEE Trans. SMC 22 418-435, 1992.

Chaker Katar, is a PH D candidate and a Teaching Assistant of computer science in Ecole Supérieure Des Sciences et Technologies de Tunis. He has received an MS degree in computer science from Institut Supérieur de Gestion de Tunis in 2002. His research interest includes intrusion detection and responses systems, data mining, neural computing, genetic computing, fuzzy set theory, and Dempster and Shafer theory.