Integrated Platform of Artificial Immune System for Anomaly Detection

Zejun Wu[1, 2] Yiwen Liang[1]
[1]School of Computer / State Key Lab of Software Engineering, Wuhan University, CHINA
[2]School of Information Science, Zhongnan University of Economics and Law, CHINA

Abstract: - Originated from human immune system, artificial immune system has been widely applied in the computational fields, especially for the task of anomaly detection. Although intensively investigated in the literature, most of the artificial immune systems involve the process of model pre-definition according to the specific scenarios to be coped with. The pre-definition, however, could cause the system to be unreliable and inflexible. Instead, we propose in this paper an integrated artificial immune system platform, which can automatically adapt to various types of immune models. For a particular object to be detected, the platform is able to configure the model set dynamically based on the "pressure" produced during the course of training and testing. In addition, a hybrid evaluation of multi-AIS-models is employed in the integrated platform to improve the self-adaptability of the system.

Keywords: - artificial immune system; anomaly detection; hybrid immune model

1 Introduction

Inspired by the mechanism of human immune system (HIS), researchers in the field of computer science have successfully introduced artificial immune systems (AIS) for solving computational tasks. Amongst them, anomaly detection, due to its wide range of applications such as cancer diagnose, virus detection, mortgage deceit, and fault diagnose, has received increasing attentions [1]. By far, most of the AIS based anomaly detection systems are designed for detecting certain object or its subset using pre-tuned AIS models, causing a large number of variants of the general immune models.

S. Forrest proposed an AIS model based on the binary string expression, r-contiguous matching and negative selection algorithm (NSA) [2]. This model functions well while it is used to analyse sequences of process calls in UNIX system and a small subset of the network traffic data. J. Kim pointed out that NSA could cause serious scaling problem in face of tremendous network traffic data, while r-contiguous matching, as a continuous matching function, is inappropriate for network traffic data with discrete feature intervals [3]. Therefore, she proposed an immune model based on feature interval expression and IF-THEN rule matching and extended the colonel selection algorithm (CSA) to analyse network traffic data with multi-dimensional features. To further enlarge the scope of intrusion detection, F. González brought forward a hyper-rectangle expression and rules based NSA [4]. Nevertheless, the enhancements so far are still limited to cope with a small dynamic range of the traffic data, such that only a small subset of the problem space can be well handled.

Currently, the choice of AIS model, including model expression, matching, training, evaluation and neutralization, is mainly pre-defined by experts based on the hypothesized problem space and their own experiences. The model is then iteratively adjusted based on newly observed experimental results that could be biased from the original model. Two disadvantages exist here: (1) the limited

experiences of experts could be unreliable for detecting objects varying in such a dynamic way; (2) the comparison between different models and algorithms is hard due to different conditions and parameter settings and thus cannot provides much useful information. In this paper we propose an Integrated Platform of Artificial Immune Systems (IPAisys) to address this issue. In this platform, various model prototypes are integrated into a unified framework and optimal model setting can be achieved through a dynamic configuration scheme. In particular, experimental data and their intermediate states generated in the course of training and testing will be utilized to regularize and combine these isolated immune models into a hybrid one. By saying hybrid, we have two implications: the optimal regularization of each single model, and the comprehensive evaluation using multi-AIS-models.

Note that in this paper we do not attempt to manifest all the models used but concentrate on explaining the approaches to the optimal configuration of AIS models.

2 Related work

Applying AIS to anomaly detection has been a hot research topic in the past decade, as evidenced by large amount of work published in the literature. S. Forrest first proposed NSA, in analogy with T cells of HIS, to deal with various anomaly detection problems. This algorithm defines "Self" as the normal behavior patterns of a network monitoring system. Several random patterns are then produced as immature detectors to be tested by “Self”. An immature detector is abandoned in case it can be matched with one individual of "Self". When the network traffic data is complex, however, the scaling problem introduced by NSA cannot be neglected [5]. For this reason, J. Kim developed a semantic expression of detector, i.e. feature intervals, to extract network features for building the rules. Meanwhile, she replaced NSA by CSA to further alleviate the scaling problem. Several real-value based expressions were also discussed by F. González, including hyper-rectangle, fuzzy rules and hyper-sphere, and corresponding detector generation algorithms, resulting an expanded problem space. The idea behind is that AIS models should be redefined or iteratively refined with the changes of problem space, which can be described as a single model evaluation process as shown in Fig.1.

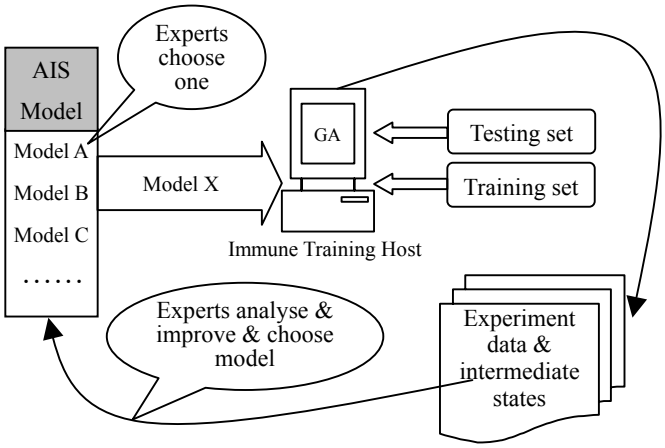

Fig.1. Single model evaluation mechanism

The significant diversities among problem spaces, and thus feature descriptions of objects to be detected, make AIS models greatly differ from each other in terms of expression, matching, training, evaluation and neutralization. To avoid time-consuming model validation, we propose in this paper a dynamic model configuration scheme, which can adjust AIS models by a set of training and testing measurements, such as fitness, false positive rate, false negative rate and so on, acting as a kind of "pressure". Different models with different detectors are then combined to evaluate the object.

3 Construction of IPAisys

In order to obtain a dynamic model configuration scheme and make use of multi-AIS-models to perform evaluation, an integrated platform of AIS need to be set up, namely IPAisys. In general, an IPAisys consists of two main modules, data preprocessing module and model configuration module, each with its own components. All

components are integrated in a uniform framework, as illustrated in Fig.2, to tune the expression and algorithm of each model for compatible cooperation.

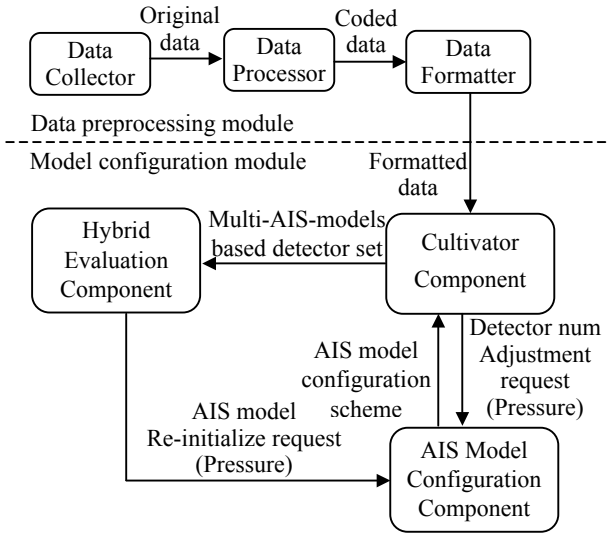

Fig.2. Architecture of the integrated platform of AIS

3.1. Data preprocessing module

In data preprocessing module, antigen data, e.g. TCP Dump data, are first collected by the data collector and transferred to the data processor, wherein the original data are coded for the purpose of compression/decompression or concatenation. Following the data processor comes the data formatter, which provides a data-uniform interface to the model configuration module. It translates coded date into XML format with the help of expert regulation files. In this way, an AIS model can be properly configured regardless the format of the original data.

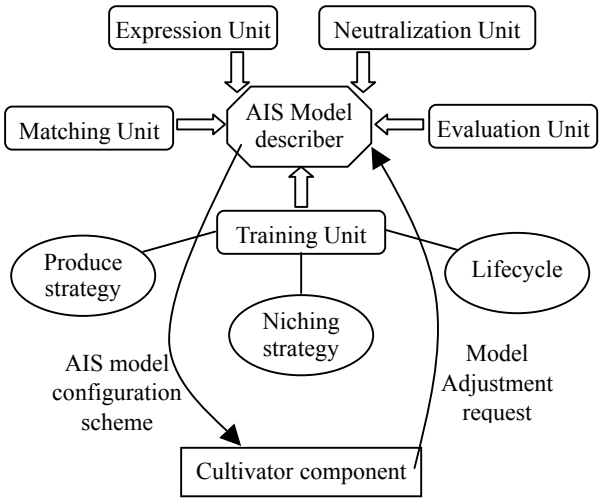

Fig.3. Structure of model configuration component of IPAisys

3.2. Model configuration module

3.2.1. AIS model configuration component

The model configuration component is responsible for providing an optimal configuration scheme to the cultivator component (see Sec. 3.2.2). We can first structure detector generation algorithm based AIS models like Fig.3, due to their common characteristics as described in [6][7][8][9][10].

Note that the five units in the figure, i.e. expression unit (Ex), matching unit (Ma), training unit (Tr), evaluation unit (Ev) and neutralization unit (Ne), can sufficiently define an AIS model. For simplicity, we fix Tr unit and ignore the Ne unit at this stage and mainly focus on the effects of the other three units. Therefore, an AIS model is represented by a triple of units, i.e.

$$AisModel = (Ex, Ma, Ev).$$

Each unit is assembled by several optional items, shown in Table 1.

Unit name	Optional item one	Optional item two	Optional item three
Expression Unit (Ex)	Binary String (BS)	Digital Sequence (DS)	Hyper-rectangle (HR)
Matching Unit (Ma)	R-contiguous (RC)	R-chunk (RK)	Satisfy Rule (SR)
Evaluation Unit (Ev)	Autarchy Evaluation (AE)	Marker Evaluation (ME)	

Table 1. Partial optional items in units of AIS model

Also note that the configuration of an AIS model should be constrained by the Ex-Ma relationship, preventing invalid AIS models from being assembled. Table 2 lists several typical relationships of dependency and restriction between Ex and Ma.

Unit name	Unit name	Type of relationship	Relationship list
Ex	Ma	Dependence	BS+RC; DS+RC; BS+RK; DS+RK; HR+SR
Ex	Ma	Restriction	BS-SR; DS-SR; HR-RC; HR-RK;

Table 2. Ex-Ma relationship of dependency and restriction

While an AIS model set can be assembled according to Table 1 and Table 2, its parameters, such as the length of digital sequence, the value of r in r-contiguous matching, and the threshold of Hamming distance matching, should be set by experts according to the models chosen. Table 3 shows an example of a randomly assembled AIS model set, where the numbers are the corresponding parameters advised by experts.

AIS model set	Ex	Ma	Ev
M_1	DS (6)	RC (3)	AE
M_2	BS (49)	HM (27)	ME
M_3	BS (49)	RC (25)	ME
…	…	…	…

Table 3. Example of AIS models randomly assembled

In the integrated platform, AIS models iteratively transform according to a probability matrix, P_{matrix}, defined as

$$P_{matrix} = \begin{pmatrix} p_{11} & p_{12} & ..... & p_{1k} \\ p_{21} & p_{22} & ... & p_{2k} \\ ... & ... & ... & ... \\ p_{k1} & p_{k2} & ... & p_{kk} \end{pmatrix}$$

where $\sum_{j=1}^{k} p_{ij} = 1 \ (i = 1...k)$.

P_{matrix} can be viewed as the weights of different detector sets generated from AIS model set. The model transition probability $p_{ij}(i = 1...k, j = 1...k)$ is the chance for M_i to transform into M_j. We set $p_{ii} = 1.0 \ (i = 1...k) \ and \ p_{ij} = 0 \ (i \neq j)$ in initialization for considering that each model should be equally weighted without other prior knowledge. For illustration, an example is shown in Fig. 4 where the AIS model set M = {M_1, M_2, M_3} contains 100 detectors for each model at the beginning. With the change of P_{matrix}, the number of detectors of M_1, M_2 and M_3 alters accordingly, to 100, 90, 80 and then 100, 80, 60 as shown on the diagonal.

$$P_{matrix} = \begin{pmatrix} 1.0 & 0 & ..... & 0 \\ 0 & 1.0 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & 1.0 \end{pmatrix}$$ First change

$$P'_{matrix} = \begin{pmatrix} 1.0 & 0 & ..... & 0 \\ 0.1 & 0.9 & ... & 0 \\ ... & ... & ... & ... \\ 0.1 & 0.1 & ... & 0.8 \end{pmatrix}$$ Second change

$$P''_{matrix} = \begin{pmatrix} 1.0 & 0 & ..... & 0 \\ 0.2 & 0.8 & ... & 0 \\ ... & ... & ... & ... \\ 0.2 & 0.2 & ... & 0.6 \end{pmatrix} .....$$

Fig.4. P_{matrix} changes from initialization to other states

The changes of $P_{matrix,}$, as the core of the model's adaptability, are incurred by the "pressure" during training, that is, the average fitness of each detector set from AIS models after N generations of evolvement in the cultivator component. Let D = {D_1, D_2, D_3…D_k} be the detector set produced by the AIS model set M = {M_1, M_2, M_3…M_k}. After N generations of evolvement, the average fitness of D_i (i=1, 2, 3…k), denoted by F_{Di}, is calculated and P_{matrix} changes based on the following rules:

(1) $P_{ij}' = P_{ij} + p_{change} \ iff \ F_{Di} \leq F_{Dj} \ (i \neq j)$;

(2) $P_{ij}' = P_{ij} - p_{change} \ iff \ F_{Di} > F_{Dj} \ (i \neq j)$;

(3) $P_{ii}' = 1 - \sum_{j=1}^{k} P_{ij}' \ (i \neq j \ and \ i = 1...k)$;

where P_{change} is a predefined offset constant (0.1 in Fig.4).

After T rounds of transformation, the average false positive rate (*FP*) of D = {D_1, D_2, D_3…D_k}, FP_{avg}, as well as the average false negative rate (*FN*)

FN_{avg}, will be measured in the hybrid evaluation component. If FP_{avg} is higher than a threshold β and FN_{avg} is higher than another threshold γ, M = {M_1, M_2, M_3…M_k} will be re-initialized. Otherwise D = {D_1, D_2, D_3…D_k} will be matured and put into real environment with hybrid evaluation function.

3.2.2 Cultivator component

The cultivator component is devised to train multi-AIS-models based detector set D = {D_1, D_2, D_3…D_k} using Genetic Algorithm (GA) and return fitness values as the "pressure". We use the standard GA algorithm whose parameters are given in Sec. 4.

3.2.3 Hybrid Evaluation Component

The hybrid evaluation component then evaluates D = {D_1, D_2, D_3…D_k} in terms of FP_{avg} and FN_{avg} using the hybrid evaluation function, the values of which decide whether an AIS model need to be re-initialized as described in Sec. 3.2.1. The hybrid evaluation function $f_{hybrid}(x)$ can be defined in two forms below:

(1) $f_{hybrid}(x) = f_{D1}(x) \otimes f_{D2}(x) \otimes ..... f_{Dk}(x)$;

(2) $f_{hybrid}(x) = \sum_{i=1}^{k} f_{Di}(x)$;

where $f_{Di}(x)$ (i=1,2…k) is the evaluation function of Detector set D_i, depending on Evaluation Unit selected in the model configuration component. $f_{Di}(x)$=1 (positive response) means that D_i detects *x* as anomaly and $f_{Di}(x)$=0 (negative response) on the contrary.

Clearly, the two functions above represent two kinds of hybrid evaluation philosophies. For function (1), *x* is detected as anomaly if at least one detector set D_j (j ∈ [1…k]) gives positive response; and detected as normality only if all detector set D = {D_1, D_2, D_3…D_k} are negative. This is the Autarchy Evaluation used in a hybrid fashion based on multi-AIS-models. On the other hand, *x* is detected as anomaly in function (2) if $f_{hybrid}(x) > \eta$ ($0 \le \eta \le k$), and vice versa, which is in fact a hybrid version of the Marker Evaluation, i.e. the decision is made by cooperative marking of D. In our case, η is set to $\lfloor k/2 \rfloor$ according to the major voting rule. Then, FP_{avg} and FN_{avg} are counted after 10 trails.

4 Experiment analysis

The data used in our experiment is a version of the 1999 DARPA intrusion detection evaluation data set maintained by MIT Lincoln Lab [11]. The statistics of the original data, such as the number of bytes per second, the number of packets per second and the number of ICMP packets per second, are obtained by data preprocessing module for later use. As there are two weeks of data at hand, we use the data of the first week to train three isolated AIS models shown in Table 4, i.e. M_1, M_2 and M_3, in the cultivator component. The population size of the whole detector is set to 300, with 100 for each AIS model. The training process finishes when no change occurs in P_{matrix}. D_i will then be sent to the evaluation component where *FP* and *FN* are evaluated using the data of the second week by the procedures described in Sec. 3.2.3.

The other parameters are set as follows. The evolvement generation in GA is equal to 100. The reproduction, crossover and mutation rate are 0.7, 0.25 and 0.05, respectively. The number of change of P_{matrix}, denoted as T, is defined to be 10. η is set to 50, and β and γ equally to 0.3. Table 5 compares the means and standard deviations of detection performances of five different models after 10 trials in terms of *FP* and *FN*.

	Single Model1 (M_1)	Single Model2 (M_2)	Single Model3 (M_3)	Multi-AIS Models (Autarchy)	Multi-AIS Models (Marker)
FP(%)	5.78 (2.31)	2.57 (1.45)	2.35 (1.53)	1.39 (0.74)	2.06 (1.12)
FN(%)	33.67 (6.11)	42.53 (6.02)	47.12 (8.56)	23.81 (5.78)	17.94 (6.65)

Table 5. An example of FP and FN comparison between single model and multi-AIS-models

From Table 5, it can be seen that the *FP* and *FN*

yielded by multi-AIS-models are much lower than those by each single model. *FP* becomes higher when multi-AIS-models adopt the autarchy way of hybrid evaluation scheme while *FN* is greater when the marker evaluation is used.

To this end, two conclusions can be reached: (1) the "pressure" produced during training and testing effectively invokes the dynamic configuration of AIS models such that the hybrid model set achieves better self-adaptability and detection performance. (2) A hybrid evaluation scheme based on multi-AIS-models improves the detection performance while different hybrid ways can lead to different *FP* and *FN* trends, providing a possibility of choice for different application needs.

Although multi-AIS-models outperforms each single model in detection, it is computationally more expensive, especially when $FP_{avg} > \beta$ or $FN_{avg} > \gamma$ during training. According to the average statistics from experiment, the multi-AIS-models were reinitialized for 5 to 6 times out of 10.

5 Conclusion

In this paper, we propose a novel IPAisys to improve the adaptability of AIS by optimally integrating and configuring various immune models in a unified framework. In particular, the IPAisys is superior to the traditional application-specific model in the following aspects: (1) it offers a platform for easy comparison between different AIS models and algorithms; (2) it is provided with the ability to find optimal model set by self-learning; (3) A hybrid evaluation scheme is naturally embedded, leading to an improved detection performance.

In the future work, unit library and model library can be introduced to make the assembly process of AIS model set more intelligent. Apart from that, the change rules of P_{matrix} and the hybrid evaluation mechanism can be further exploited to improve the performance of the system.

Acknowledgements

This work is supported by Key Research Grant No. 90204011 from National Natural Science Foundation of China.

Reference:

[1] Zejun Wu, Hongbin Dong, Yiwen Liang, R. I. McKay, A Chromosome-based Evaluation Model for Computer Defense Immune Systems, proceedings of 2003 IEEE Congress on Evolutionary Computation, vol.3, pp.1363-1369, 2003

[2] S. Forrest, A.S. Perelson, L. Allen & R. Cherukuri. Self-nonself discrimination in a computer. In: Proceedings of the 1994 IEEE Symposium on Research in Security and Privacy. Los Alamos, CA: IEEE Computer Society Press, 1994

[3] J. Kim, Integrating Artificial Immune Algorithms for Intrusion Detection. Ph.d thesis, University College London, 2002

[4] F. Gonzalez, A Study of Artificial Immune Systems Applied to Anomaly Detection. Ph.d thesis, The University of Memphis, 2003

[5] Zejun Wu, Lijin Qian, Yiwen Liang, An Immunity-based Clonal Selection Algorithm for Intrusion Detection Systems, Computer Engineering, Vol.30 (6), pp.50-52, 2004

[6] Uwe Aickelin, Julie Greensmith, and Jamie Twycross, Immune System Approaches to Intrusion Detection – A Review. In Proc. of the Third Internation Conference on Artificial Immune Systems (ICARIS-04), pages 316–329, 2004

[7] P. Bentley and J. Timmis, A Fractal Immune Network. In Proc. of the Third Internation Conference on Artificial Immune Systems (ICARIS-04), pages 133–145, 2004

[8] F Gonzalez and D Dasgupta, Anomaly detection using real-valued negative selection. Journal of Genetic Programming and Evolvable Machines, 4:383–403, 2003

[9] L de Castro and J Timmis, Artificial Immune Systems: A New Computational Intelligence Approach. Springer, 2002.

[10] S. Hofmeyr and S. Forrest, Architecture for an artificial immune system, Evolutionary Compuiation, vol. 8, no. 4, pp. 443-473, 2000

[11] 1999 Darpa intrusion detection evaluation. MIT Lincoln Labs, 1999.