2000 년 한국정보처리학회 추계 학술발표논문집 제 7 권 제 2 호

네트워크 침입 탐지를 위한 인공 면역 시스템에서의 부정적 선택(Negative Selection) 알고리즘

*김 정원, *Peter Bentley, **최 종욱

* 런던대학교 전산학과
** 상명대학교 정보통신학부

Negative Selection within an Artificial Immune System for Network Intrusion Detection

*Jungwon Kim, *Peter J. Bentley and **Jong-Uk Choi
*Department of Computer Science, University College London
** School of Information Communication , Sangmyung University
e-mail: J.Kim@cs.ucl.ac.uk

요 약

This paper describes on-going research, applying an artificial immune system to the problem of network intrusion detection. The paper starts by introducing the motivation and rationale of this research. After describing the overall architecture of the proposed artificial immune system for network intrusion detection, the real network traffic data and its profile features used in this research are explained. As the first step of this effort, the negative selection algorithm, which is one of three significant evolutionary stages comprising an overall artificial immune system, is investigated and initial results are briefly discussed. Finally, the direction of future work is discussed based on this initial result and the contribution of this research is addressed.

1. Introduction

The biological immune system has been successful at protecting the human body against a vast variety of foreign pathogens [10]. A growing number of computer scientists have carefully studied the success of this competent natural mechanism and proposed computer immune models for solving various problems including fault diagnosis, virus detection, and mortgage fraud detection [2]. Among these various areas, intrusion detection is a vigorous research area where the employment of an artificial immune system has been examined [2], [9]. The main goal of intrusion detection is to detect unauthorised use, misuse and abuse of computer systems by both system insiders and external intruders. Among automated intrusion detection systems, a particular system for network intrusion detection, known as a network-based intrusion detection system (IDS), monitors any number of hosts on a network by scrutinising the audit trails of multiple hosts and network traffic. This research proposes a novel approach to building a network-based IDS, which is inspired by a human immune system.

Currently many network-based IDS's have been developed using diverse approaches [7]. Nevertheless, there still remain unresolved problems for building an effective network-based IDS [5]. As one approach of providing the solutions to these problems, previous work [6] identified a set of general requirements for a successful network-based IDS and three design goals to satisfy these requirements: being *distributed, self-organising* and *lightweight*. In addition, Kim and Bentley [5] introduced a number of remarkable features of human immune systems that satisfy these three design goals. It is anticipated that the adoption of these features should help the construction of an effective network-based IDS.

However, it is not clear yet how to implement these beneficial features for a real network-based IDS. For instance, Hofmeyr's work [4] showed the unique features of artificial immune systems that are advantageous for network intrusion detection, but this system's applicability to detect various real network intrusions was not validated. This is because his system used the small set of selected profile features and thus detected only a limited number of network intrusions [4]. As a consequence, this research focuses on building an artificial immune system that is more applicable to the detection of various real network intrusions.

2. System Overview

The main idea of this model is distinguishing self, which is normal, from non-self, which is abnormal [9]. In this research, with respect to network intrusion detection, we view the normal activities of monitored networks as self and their abnormal activities as non-self. Many sophisticated network intrusions such as sweeps, co-ordinated attacks and Internet worms are detected by monitoring the anomalies of network traffic patterns. Thus, the artificial immune model is designed for distinguishing normal network activities from abnormal network activities and is expected to detect various network intrusions.

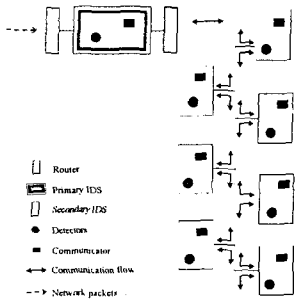

Figure 1: The Physical Architecture of an Artificial Immune System

More details about this model are explained in [5], [6] and a brief summary follows. The artificial immune model for network intrusion detection consists of a primary IDS and secondary IDS's [6]. For a human body, at the bone marrow and the thymus, various detector cells, called antibodies, are continuously generated and distributed to secondary lymph nodes, where antibodies reside to monitor living cells. The distributed antibodies monitor all living cells and detect non-self cells, called antigens, invading into the human body. For the artificial immune model, the primary IDS, which we view as the bone marrow and thymus, generates numerous detector sets. They describe abnormal patterns of network traffic packets. They are unique and transferred to each local host. We view local hosts as secondary lymph nodes, detectors as antibodies and network intrusions as antigens. At the secondary IDS's, which are local hosts, detectors are used by background processes which monitor whether non-self network traffic patterns are observed from network traffic patterns profiled at the monitored local host. The primary IDS and each secondary IDS have communicators to allow the transfer of information between each other. Figure 1 shows the physical architecture of artificial immune system proposed in this research.

For the proposed artificial immune system, the several sophisticated mechanisms of the human immune system which allow it to satisfy three design goals of a competent network-based IDS are embedded in three evolutionary stages: gene library evolution, negative selection and clonal selection. (Figure 2).

Gene library evolution simulates the first stage of evolution, which learns knowledge of currently existing antigens. This process allows the model to be lightweight and self-organising. *Gene expression* and *negative selection* form the second stage of evolution, generating diverse pre-detectors and selecting mature detector sets by eliminating false pre-detectors in a self-organising way. The transfer of unique detector sets to the secondary IDS's also occurs at this stage, making the model distributed. *Clonal selection* is the third stage of evolution, detecting various intrusions with a limited number of detector sets using approximate binding, and generating memory detectors. The generality and efficiency of these mechanisms results in the model being lightweight. In addition, this process drives the gene library evolution in the primary IDS. These three stages are co-ordinated across a network to satisfy the three goals for designing effective IDS's: being *distributed, self-organising* and *lightweight*. Analysis of the characteristics of this unified evolutionary approach show that, unlike existing approaches, the proposed artificial immune model does satisfy the requirements of network-based IDS's.

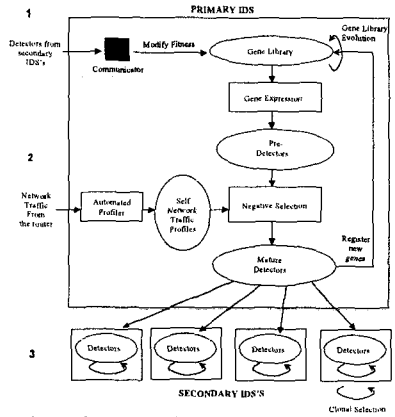

Figure 2. Conceptual Architecture of the Artificial Immune Model

3. Network Traffic Data VS Network Intrusion Signature

The data chosen for this work was collected for a part of an Information Exploration Shootout, which is a project providing several data sets publicly available for exploration and discovery and collecting the results of participants. It is available at http://iris.cs.uml.edu:8080/network.html. This set was created by capturing TCP packet headers that passed between the intra-LAN and external networks as well as within the intra-LAN. This set consists of five different data sets. The TCP packet headers of the first set were collected when no intrusion occurred and the other four sets were collected when four different intrusions were simulated. These intrusions are: IP spoofing attack, guessing rlogin or ftp passwords, scanning attack and network hopping attack. The details of attack signatures and attack points of the four different attacks are not available.

The data originally had the fields of network packet capturing tool's format such as time stamp, source IP address, source port, destination IP address, destination port and etc. However, the primitive fields of captured network packets are not enough to build a meaningful profile. Consequently, it is essential to build a data-profiling program to extract more meaningful fields, which can distinguish "normal" and "abnormal". Many researchers have identified the security holes of TCP protocols [8] and so the fields used by our profiles are selected based on the extensive study of this research. They are usually defined to describe the activities of each single connection.

The automated profile program was developed to extract the connection level information from TCP raw packets and it was used to elicit the meaningful fields of the first data set.

For each TCP connection, the following fields are extracted :

- Connection identifier: each connection is defined by four fields, initiator address, initiator port, receiver address and receiver port. Thus, these four fields are included in the profile first in order to identify each connection.
- Known port vulnerabilities: many network intrusions attack using various types of port vulnerabilities. There are fields to indicate whether an initiator port or a

receiver port potentially holds these known vulnerabilities.

- 3-way handshaking: TCP protocol uses 3-way handshaking for a reliable communication. When some network intrusions attack, they often violate the 3-way handshaking rule. Thus, there are fields to check the occurrences of 3-way handshaking errors.
- Traffic intensity: network activities can be observed by measuring the intensity over one connection. For example, number of packets and number of kilobytes for one specific connection can describe the normal network activity of that connection.

Thus, in total, self profile fields have 35 different fields for the first data set and 41 different fields for the second data set.

4. First Stage – Negative Selection Algorithm

Out of three evolutionary stages of the artificial immune system, the first stage, negative selection, is investigated and its experiment results are briefly reported in this section.

As the first step of the human immune system's detection mechanism, when a new antibody is generated, the gene segments of different gene libraries are randomly selected and concatenated in a random order. The main idea of this gene expression mechanism is that a vast number of new antibodies can be generated from new combinations of gene segments in the gene libraries. However, this mechanism introduces a critical problem. The new antibody can bind not only to harmful antigens but also to essential self cells. To prevent such serious damage, the human immune system employs negative selection. This process eliminates immature antibodies, which bind to self cells passing by the thymus and the bone marrow. Therefore, the negative selection stage of the human immune system is important to assure that the generated antibodies do not attack self cells.

In addition to this role of negative selection in a human immune system, which is to eliminate harmful antibodies, Forrest et al [3], [9] claimed that it shows some other important features, which can help us to devise a more effective anomaly detection algorithm. They proposed and used a negative selection algorithm for various anomaly detection problems. This algorithm consisted of three phases: defining self, generating detectors and monitoring the occurrence of anomalies. In the first phase, it defines 'self' in the same way that other anomaly detection approaches establish the normal behaviour patterns of a monitored system. In other words, it regards the profiled normal patterns as 'self' patterns. In the second phase, it generates a number of random patterns that are compared to each self pattern defined in the first phase. If any randomly generated pattern matches a self pattern, this pattern fails to become a detector and thus it is removed. Otherwise, it becomes a 'detector' pattern and monitors subsequent profiled patterns of the monitored system. During the monitoring stage, if a 'detector' pattern matches any newly profiled pattern, it is then considered that new anomaly must have occurred in the monitored system.

D'haeseleer, Forrest and Helman [1] showed that this algorithm has several advantages of negative selection as a novel distributed anomaly detection approach. One of the formidable features is that this novel approach does not define specific anomalies to be detected and thus it does not require the prior knowledge of anomalies. This feature allows it to be able to detect previously unseen anomalies.

However, the current negative selection algorithms show several drawbacks. The most significant problem is the excessive computational time caused by the random-generation approach to building valid detectors. This results in the exponential growth of computational effort with the size of self patterns [1]. Moreover, it is very difficult to know whether the number of generated detectors is large enough that can satisfy the acceptable detection failure probability. D'haeseleer derived a formula presenting an appropriate number of detectors when an acceptable failure probability is given and claimed that the derived formula allows the negative selection algorithm to tune its detection accuracy against the cost of generating and storing detectors. This work has been accomplished under some unrealistic assumptions: it does not take into account false positive error and dependence between self patterns. Furthermore, he only considered binary patterns and a simple r-contiguous bit matching rule. Nevertheless, it is not easy to estimate the appropriate number of detectors when the negative selection algorithm employs numerical patterns and a more sophisticated matching rule. This difficulty may force the negative selection algorithm to adopt an arbitrary number of detectors and this may cause an unexpectedly low detection accuracy or the inefficient computation by generating more than is needed.

4.1 Experiment Design

These problems are exemplified through a series of experiments that apply a negative selection algorithm on the first data set. The negative selection algorithm used in these experiments mainly followed the implementation details which are used in [3]. However, there are several things that are different from Forrest's implementation details. In the encoding of detectors, each gene of detector has an alphabet of cardinality 10 with values from '0' to '9' and the allele of this gene indicates the 'cluster number' of corresponding field of profiles. As described in the previous section, the profile built from the first data set has 35 fields and this number determines the total number of corresponding genes in the detectors. From these 35 fields, the values of 28 fields are continuous and the values of the other 7 fields are discrete. Specifically, the continuous values of 28 fields show a wide range of values. In order to handle this various and broad range of values, an overall range of real values for each field is sorted. Then, this range is discretised into a predefined number of clusters. The lower bound and higher bound of each cluster are determined by ensuring that each cluster contains the same number of records. This modification is necessary in order to save the length of encoded detector. Furthermore, our implementation of measuring the similarity between a generated detector and a self profile is operated at the phenotype level while Forrest's is performed at the genotype level.

Other implementation details have been kept the same as Forrest's [3]. For example, the same matching function, the r-continuous matching function for measuring the similarity is used. Its matching threshold is defined as 9. In order to define this number, the formula to approximate the appropriate number of detectors when a false negative error is fixed [1], [3] is used. It was shown that the longer matching threshold drives the creation of more general detectors, but it also causes a larger number of random detector generation trials, which need to avoid the matching a

self profile [1], [3]. Thus, we can derive an approximate appropriate matching threshold number by varying the expected false negative error and random detector generation trial number. Even though this formula is clearly useful to predict the appropriate number of detectors and its generation number, its predicted number showed how infeasible this approach is for applying it on a more complicated search space. For instance, when the expected false negative error rate is fixed as 20%, its predicted the detector generation trial number is 51 and the appropriate number of generated detectors is 21935 for the matching threshold is 3. Similarly, when we define the matching threshold is 4, it predicted 535 for the former and 955 for the latter. None of these cases seem to provide any feasible test case in terms of computing time. In addition, it was observed that when we fixed the matching threshold number as four and ran the system, the system could not manage to generate any single valid detector after one day. Thus, we generated valid detectors by setting the matching threshold number that allowed a system to generate a valid detector in a reasonable time.

4.2 Experiment Result

It was observed that the average time of successful detector generation took about 70sec CPU time and the average number of trails to generate a valid detector was 2.6 when a matching threshold was nine. Even though this number gave reasonable computing time to generate a valid detector set, very poor detection accuracy by generated detectors was shown. The maximum 1000 valid detectors were generated and the detection accuracy was measured per every 100 detectors. The observed detection accuracy was less than 20% for four different intrusion data sets and one artificially generated random test set. This result was gained as the average of five runs.

In contrast to the promising results shown in Hofmeyr's negative selection algorithm for network intrusion detection [4], the experiment result of this research raises doubt whether this algorithm should be used for network intrusion detection. These contradictory findings can be explained by the fact that Hofmeyr's encouraging result originated from the adoption of limited profile features which a negative selection algorithm can handle, while the experiment of this research used the more complicated but more realistic profile features that a negative selection algorithm struggles to solve. More importantly, Forrest [3], [9] and Hofmeyer [4] view that the network intrusion detection of artificial immune system is achieved mainly by the sole function of negative selection stage than the co-ordination of three different evolutionary stages. This is somewhat different from our view.

Consequently, the initial results of our experiments motivated us to re-define the real role of negative selection stage within an overall network-based IDS and design a more applicable negative selection algorithm, which following a newly defined role. As much of the other immunology literature addresses [10], the antigen detection powers of human antibodies rise from the evolution of antibodies via a clonal selection stage. While Forrest et al's negative selection algorithm allows it to be an invaluable anomaly detector, its infeasibility is also caused from allocating a rather overambitious task to it. To be more precise, the job of a negative selection stage should be restricted to tackle a more modest task which is closer to the role of negative selection of human immune system. That is simply filtering the harmful antibodies rather than generating competent ones.

5. Conclusion and Future Work

The novel artificial immune system presented in this paper is designed to overcome the weaknesses of conventional network-based IDS's. This system combines the three evolutionary stages: gene library evolution, negative selection and clonal selection into a single methodology. These three processes are co-ordinated across a network to satisfy the three goals for designing effective IDS's: being distributed, self-organising and lightweight. Analysis of the characteristics of this unified evolutionary approach show that, unlike existing approaches, the proposed artificial immune model does satisfy the requirements of network-based IDS's. Consequently, algorithms based on this model show considerable promise for future IDS's.

As the first attempt of this effort, the negative selection stage was implemented and experiments showed its infeasibility for its application to the essential profiling fields of real network data. This result directs this research to re-define the role of negative selection algorithm within the overall artificial immune system framework. For the future work, the intrusion detection mechanism of clonal selection stage will be investigated and the clear understanding of task of clonal selection stage will help us to comprehend the distinct job of negative selection stage.

The contributions of this work will provide an applicable methodology for designing an artificial immune system to be able to perform network intrusion in a truly distributed, self-organising and lightweight way.

6. Acknowledgement

This work was supported by the Korea International Collaboration Research Funds (I-03-002), the Ministry of Science and Technology, Korea.

References

[1] D'haeseleer, P, 1997, A Distributed Approach to Anomaly Detection, *ACM Transactions on Information System Security.*

[2] Dasgupta, D., "An Overview of Artificial Immune Systems and Their Applications", In Dasgupta, D. (editor). *Artificial Immune Systems and Their Applications*, Springer-Verlag. pp.3-21, 1998.

[3] Forrest, S. et al, "Self-Nonself Discrimination in a Computer", *Proceeding of 1994 IEEE Symposium on Research in Security and Privacy*, Los Alamos, CA: IEEE Computer Society Press, 1994.

[4] Hofmeyr, S., *An Immunological Model of Distributed Detection and Its Application to Computer Security*, Phd Thesis, Dept of Computer Science, University of New Mexico, 1999.

[5] Kim, J. and Bentley, P.,"The Human Immune System and Network Intrusion Detection", *EUFIT'99, Aachen, Germany*, 1999.

[6] Kim, J. and Bentley, P., 1999, "The Artificial Immune Model for Network Intrusion Detection", *EUFIT'99, Aachen, Germany*, 1999.

[7] Mykerjee, B., et al, "Network Intrusion Detection", *IEEE Network*, Vol.8, No.3, pp.26-41, 1994.

[8] Porras, P. A. and Valdes, A., "Live Traffic Analysis of TCP/IP Gateways", *Proceeding of ISOC Symposium of Network and Distributed System Security*, 1998.

[9] Somayaji; A., Hofmeyr, S. and Forrest, S.,"Principles of a Computer Immune System", *Proceeding of New Security Paradigms Workshop, Langdale, Cumbria*, pp.75-82. 1997.

[10] Tizard, I. R., 1995, *Immunology: Introduction*, 4th Ed, Saunders College Publishing