Network traffic detection based on part matching and section evolution of immune elements

Caiming Liu\textsuperscript{1,2,3}, Yan Zhang\textsuperscript{1,2,3}, Chunming Xie\textsuperscript{1}, Deming Wang\textsuperscript{4} and Zhonghua Li\textsuperscript{1}

\textsuperscript{1}School of Artificial Intelligence, Leshan Normal University, Leshan 614000, China
\textsuperscript{2}Intelligent Network Security Detection and Evaluation Lab., Leshan Normal University, Leshan 61400, China
\textsuperscript{3}Internet Natural Language Intelligent Processing Key Lab. of Education Department of Sichuan Province, Leshan Normal University, Leshan 61400, China
\textsuperscript{4}Scientific Research Department, Leshan Normal University, Leshan 61400, China

Corresponding author and e-mail: Yan Zhang, zhangyan@lsnu.edu.cn

Abstract. Immune algorithms can improve the self-adaptability for network traffic detection. Classical immune algorithms dynamically evolve and match the whole content of immune elements. However, most features of network traffic are isomeric to each other. The general matching and evolution for the whole immune elements slow down the process of self-adaptability to an extent and reduce the accuracy of immune elements’ evolution. In this paper, available network data features are analysed for the network traffic detection. The section simulation method in the immune system is studied by creating a math method for part matching of immune elements which can evolve in parts. The process of section evolution for immune elements is inferred. A detection method for recognizing anomaly network traffic is proposed. Simulation experiments show that the self-adaptability of the proposed method to the real network environment is superior to that of traditional immune algorithms.

1. Introduction
Artificial immune algorithms have been widely applied to solving network security problems, especially in the areas of the network security detection for network traffics. They have become an important intelligent detection method. Network traffics are exhibited as package sequences. Furthermore, antigens in Biological Immune System (BIS) \cite{1} are exhibited as cell sequences. Both network traffic detection and BIS need to recognize normal and abnormal elements among these sequences. Network traffic detection is inspired by the nature of dynamic evolution in BIS. Therefore, immune algorithms with bionic characteristics have innate advantages in network traffic detection.

Inspired by BIS, researchers proposed Artificial Immune System (AIS) \cite{2,3} and apply it to solving complicated problems in engineering systems \cite{4}. AIS is considered to be a system of great engineering application value \cite{5} and attracts a lot of attentions of many researchers in recent years. Researchers further proposed Computer Immunology (CI) \cite{6,7} and Computer Immune System (CIS) \cite{8} to apply AIS to solving computer engineering problems \cite{9}, computer security problems \cite{10} and network security problems \cite{11}.
In the research of traditional network traffic detection based on immunity, generally, immune detection elements composing of binary strings are entirely created, evolved and matched. However, the heterogeneity between network traffic features and immune system feature information that constitute immune detection elements is not considered. It is manifested in the difference in the meaning and expression of feature information between network packet features and immune system features, between network packet features, and between immune system features. For example, between network packets’ IP addresses and protocol types which characterize network traffic features, the former uses packet numbers which are limited in the range of digital values, while the protocol types are expressed by fixed numbers. Traditional network traffic detection based on immunity mixes IP addresses and protocol types together to create, evolve and match immune elements, which loses the true meanings of the features themselves. To a certain extent, it reduces the convergence speed of the dynamic evolution of immune detectors and the accuracy of the identification of immune elements.

A network traffic detection based on part matching and section evolution for immune elements is created in this paper. It is expected to improve the general matching and overall evolution of immune elements in traditional immune algorithms.

2. Proposed method

According to the difference of network traffic characteristics, the proposed method simulates the immune elements that can relatively fully express the network traffic characteristics, and constructs part matching method and section evolution method for immune elements, so that the immune algorithm can train immune elements according to the network traffic characteristics in the actual network security environment, so as to improve the efficiency and accuracy of anomalous network traffic identifications.

2.1. Definition of network traffic element

According to the definition of the artificial immune principle, the network traffic element of the proposed method aims to simulate antigens in the immune system. Let the network traffic element be $p$, which meets Eq. (1).

$$p = (ip\text{Src}, ip\text{Des}, port\text{Src}, port\text{Des}, protocol\text{Type}, ttl, length, offset, window, first\text{line})$$

The data elements of $p$ come from the key feature information of network packets, where, $ip\text{Src}$ and $ip\text{Des}$ represent the source and destination IP addresses separately, $ip\text{Src} \lor ip\text{Des} = (sec1, sec2, sec3, sec4)$, $sec_i (1 \leq i \leq 4)$ is in decimal, which is one of the 4 sections of an IP address. $port\text{Src}$ and $port\text{Des}$ represent the source and destination port number separately. $protocol\text{Type}$ represents the protocol type. $ttl$ represents the lifecycle of a network packet. $length$ represents the length of the packet. $offset$ represents the offset of the packet’s content. $window$ represents the sliding window size of the packet in TCP style. All data elements above are represented by decimal integers.

Simulation of Immune Element

The immune elements in artificial immune system include self cells and immune cells. In order to simulate the immune evolution process of biological immune system, the evolution process of immune cells is divided into three stages: immature, mature and memory. Similarly, immune cells in these three stages are called immature immune cells, mature immune cells and memory immune cells.

In the network traffic detection based on the immune principle, the self cell represents the normal network packet, its definition is consistent with the network traffic element. Let the self element be $s$. Immune cells are used to detect whether there are abnormal network packets in network traffic elements. They need not only the characteristic information of abnormal network packets, but also the characteristic information of the dynamic evolution of immune principle. In this paper, the immune cells in network traffic detection based on immune principle are still named as detectors by using the classical immune algorithm. In the proposed method, the detector is used to simulate the immune cell in the artificial immune system. It consists of two parts of information: network traffic element information and immune feature information. Let the element representing a detector be $d$, which
meets \( d = (p, \text{count}, \text{age}) \), where, \( p \) is the network traffic element information, \( \text{count} \) is the number of self cells or antigens matched by the detector, and \( \text{age} \) is the number of generations of the surviving detector.

2.2. Part matching of immune elements

In this paper, the part matching method for immune elements is proposed to solve the problem of the fine identification of network traffic characteristics and avoid the disadvantages of general matching of traditional \( r \)-contiguous matching method. The part matching method proposed in this paper involves matching each feature separately, so a series of part matching functions are established. Let one of the functions be \( f_i \), where, \( i \) is a nature number, \( 1 \leq i \leq 10 \), \( 0 \leq f_i \leq 1 \), it respectively represents the function of part matching for 10 features of network traffic elements, which include \( \text{ipSrc} \), \( \text{portDes} \), \( \text{portSrc} \), \( \text{ipDes} \), \( \text{protocolType} \), \( \text{ttl} \), \( \text{length} \), \( \text{offset} \), \( \text{window} \), \( \text{firstline} \). Let the total matching function be \( f \), as shown in Eq. (2).

\[
f(d_1, d_2) = \sum_{i=1}^{10} \mu_i f_i(d_1, p, d_2, p)
\]  

Where, \( d_1 \) and \( d_2 \) are immune elements whose similar values are to be calculated. \( \mu_i \) is the weight coefficient of each feature, which meets \( 0 < \mu_i < 1 \) and represents the importance of the \( i \)th feature for its corresponding network traffic element.

According to the value type, the characteristic information of network traffic elements is classified into three categories: IP address, integer value and character information. According to these three categories, the part matching functions are constructed in this paper. In the feature information of network traffic elements, \( \text{ipSrc} \) and \( \text{ipDes} \) represent the IP address information. The same part matching function is constructed for the IP addresses, that is, \( f_1 \) and \( f_2 \) are the same. The information of the seven features which include \( \text{portSrc} \), \( \text{portDes} \), \( \text{protocolType} \), \( \text{ttl} \), \( \text{length} \), \( \text{offset} \) and \( \text{window} \), is decimal integer. The part matching functions of these seven features are \( f_3, f_4, f_5, f_6, f_7, f_8 \) and \( f_9 \), respectively. However, the definition domains of the part matching functions corresponding to each feature information are different. Under the condition of defining the definition domain, the same part matching function is also constructed for them. The feature information of \( \text{firstline} \) expresses character information, which is expressed by binary string, and the traditional \( r \)-Contiguous matching method is used to construct the part matching function. However, the definition domain of the part matching function corresponding to each feature information is different. Under the condition of defining the definition domain, the same part matching function is also constructed for them. The feature information of \( \text{firstline} \) expresses characters’ information, which is expressed by binary strings, and the traditional \( r \)-Contiguous matching method is used to construct the part matching function.

\( f_1 \) and \( f_2 \) are the part matching functions of \( \text{ipSrc} \) and \( \text{ipDes} \) respectively. They have exactly the same effect. Let their function be \( f_{IP} \), which is shown in Eq. (3).

\[
f_{IP}(p_1, p_2) = \sum_{i=1}^{4} \frac{|d_1.ip\text{Src}.sec_i - d_2.ip\text{Src}.sec_i|}{d_1.ip\text{Src}.sec_i + d_2.ip\text{Src}.sec_i}
\]  

Where, the matching result of \( f_{IP} \) is the sum of the similarity of the absolute difference of the four sections of the IP addresses of \( d_1.p \) and \( d_2.p \). The smaller it is, the higher the similarity between network traffic elements is.

\( f_3, f_4, f_5, f_6, f_7, f_8 \) and \( f_9 \) are the part matching functions of \( \text{portSrc} \), \( \text{portDes} \), \( \text{protocolType} \), \( \text{ttl} \), \( \text{length} \), \( \text{offset} \) and \( \text{window} \) respectively. They have the same calculation method, but the definition fields of each feature information are different. It therefore needs to be limited. Let their function be \( f_{INT} \), which is shown in Eq. (4).

\[
f_{INT}(p_1, p_2) = \frac{|d_1.p.\text{feature}_i - d_2.p.\text{feature}_i|}{d_1.p.\text{feature}_i + d_2.p.\text{feature}_i}, \quad 3 \leq i \leq 9
\]  

Where, \( \text{feature}_i \) represents the values of the seven features which include \( \text{portSrc}, \text{portDes}, \text{protocolType}, \text{ttl}, \text{length}, \text{offset} \) and \( \text{window} \). The definition domains of the seven features are limited, which is shown in Eq. (5).
2.3. Section evolution of immune elements

In order to adapt the change of the network environment, detectors need to be evolved in network traffic detection. The evolution of detectors is dynamic. The immune system generates new detectors through randomness, cross replication, mutation and other operations. And then through matching with antigens, the detectors continuously accumulate the immune characteristic information. Basically based on the information of the immune characteristic fields including count and age, the detectors are evolved according to the three stages that include immaturity, maturity and memory. In order to increase the effectiveness of new detectors, the network traffic element information of detectors is evolved according to the different characteristic information sections, avoiding the disadvantages of the general evolution method of the overall information.

In the process of section evolution, detectors are evolved according to IP address, integer value and character information of the network traffic elements. Section evolution has the following methods: random generation, section mutation of memory detectors, cross replication of features between different memory detectors.

1) For the feature information of IP address, the features ipSrc and ipDes of network traffic elements are evolved in sections according to the following methods.

   (1) Randomly generate IP address section value. New values of four sections sec_i(1 ≤ i ≤ 4) are randomly generated in the value domain [0,255]. The four new random sections are used to form a new IP address conforming to the IP address rules.

   (2) Mutate the sections of the IP address of memory detectors. The elements in the memory detector data set are selected according to a certain proportion. And the values of some sections (no more than 2 sections) of the IP address field in the selected elements are randomly mutated in the value domain [0,255]. The mutated IP addresses become new IP address fields of immature detectors.

   (3) Crosswise replicate the IP address fields of several memory detectors. From the memory detectors that recognize more abnormal network traffic elements, the sections’ value of depth(2 ≤ depth ≤ 4) memory detectors are selected and crosswise replicated. The crosswise replicated IP address sections are used to form a new IP address. The cross replication method of the source IP address is shown in Eq. (6).

   \[
   ipNew = (ipSrc_{j_1}.sec_{i_1}, ipSrc_{j_2}.sec_{i_2}, ipSrc_{j_3}.sec_{i_3}, ipSrc_{j_4}.sec_{i_4})
   \] (6)

   Where, ipNew is the newly generated IP address after the section evolution. j1 – j4 indicates the serial number of one of the \( \theta \) memory detectors. It means that at least two memory detectors form a new source IP address, i.e. two or three section values of the new immature detector’s source IP address may be the same as the original memory detectors.

   The cross replication method of destination IP addresses has the same principle as Eq. (6). However, the source IP address of memory detectors can only be crosswise replicated to a new source IP address, and the destination IP address of memory detectors can only be crosswise replicated to a new destination IP address.

2) For the feature information of integer value type, the mutation is carried out in the definition domains of portSrc, portDes, protocolType, ttl, length, offset and window. The definition domains are shown in Eq. (5). The mutated feature’s value is assigned to the corresponding feature of the new detector. At the same time, the value of the corresponding feature information in the memory detectors can also be selected as the value of the corresponding feature of the new detector.

3) For the feature information of firstline that is character information type, it is mutated according to the binary string format.

After the IP address, integer value and character information are piecewise evolved, all new segmented information is reconstructed to generate a new network traffic element, and then a new
detector is formed together with the new network traffic element and the new immune feature information \( (\text{count} = 0, \text{age} = 0) \) as the initial value of the dynamic evolution of immune elements.

2.4. Recognition of network traffics
Let the recognition function of network traffics be \( f_r \), which is shown in Eq. (7).

\[
f_r(d, p') = \begin{cases} 
  \text{true}, & \text{if } f(d, p, p') \leq \gamma \\
  \text{false}, & \text{if } f(d, p, p') > \gamma 
\end{cases}
\]

(7)

Where, \( d \) is a detector. \( p' \) is a captured network data element (i.e. antigen). \( \gamma \) is a matching value. The \( if \) operation indicates that the condition judgment "if".

The proposed detection method of this paper is to identify abnormal network traffics, that is, to judge whether the similar value of a memory detector and an antigen has reached the matching threshold.

3. Simulation experiment
In order to verify the effectiveness of the proposed method, the algorithm proposed in this paper is programmed, and the prototype software is formed, and the simulation experiment is carried out in the real network environment.

3.1. Experimental scheme
The simulation experiments are carried out on Dell Precision 7540 mobile workstation. The software environment parameters of the simulation experiment are as follows. 1) Operating system: Windows 10. 2) Programming language: Java.

3.2. Performance comparison
In order to compare the performance of the proposed method with the traditional immune method, the traditional immune algorithm is also implemented by programming, and the prototype software of traditional immune algorithm is formed. Both methods adopt the same immune implementation process, capture network traffic in real time, and adopt the first six key features of network traffic elements defined in Section 2.1. However, the traditional immune algorithm uses binary strings to represent the network data to be detected, and uses \( r \)-Contiguous bit matching algorithm to conduct the overall evolution of immune elements.

The performance comparison experiment was performed for three times, and the average value of the number of mature detectors generated by the two methods was taken. The comparison data of the experiments’ results of the two methods are shown in Table 1 which only lists the number of mature detectors and the time consumed in the key iterations.

| Number of Iterations | Proposed Method | Traditional Method |
|----------------------|-----------------|--------------------|
|                      | Number of Mature Detectors | Time-consuming (ms) | Number of Mature Detectors | Time-consuming (ms) |
| 1                    | 24              | 2161               | 220              | 3962              |
| 50                   | 49              | 12673              | 305              | 15753             |
| 100                  | 75              | 13212              | 390              | 17313             |
| 500                  | 289             | 53621              | 1053             | 56180             |
| 1000                 | 561             | 149179             | 1878             | 151987            |

It can be seen from Table 1 that in the same iteration, the mature detectors generated by the proposed method are relatively less and time-consuming is relatively less, but the performance of mature detectors generated by the proposed method is relatively excellent, which is mainly reflected in the generation of memory detectors. For the proposed method, in the three experiments, the number of
memory detectors is 4, 3 and 4 respectively. Anomaly network traffic was detected by the proposed method in each experiment. The number of abnormal packets detected is 5452, 668 and 6080 respectively. For the traditional method, no memory detector was generated and no anomaly network traffic was detected in three experiments.

4. Conclusions
According to the type and meaning of the value of each feature, the part matching function is constructed separately. The similarity value calculated by the part matching function can more accurately calculate the similarity value between immune elements and network data elements or between network data elements. The section evolution method can generate each feature’s information of the network data elements independently, so that the new detector can better learn the good detection ability from the existing memory detector. The results of the simulation experiments show that immune elements can effectively evolve by the proposed section evolution method and the part matching method to effectively identify the abnormal network traffics in the network data flow.

Acknowledgement
This work is supported by Sichuan Science and Technology Program (No. 2018JY0523), the Scientific Research Project of Leshan Normal University (No. LZDP005), the Scientific Plan Project of Leshan (No. 19GZD027 and 19GZD042), the Project of Internet Natural Language Intelligent Processing Key Laboratory of Education Department of Sichuan Province (No. INLP201907).

References
[1] Cao Xue-Tao. Medical Immunology(7th Edition). Beijing: People's Medical Publishing House, 2018. (In Chinese)
[2] Timmis J, Neal M, Hunt J. Data analysis with artificial immune systems and cluster analysis and kohonen networks: some comparisons[C]. In: Proceedings of IEEE SMC '99 Conference Proceedings, Piscataway, NJ, USA, 1999.
[3] Timmis J, Neal M. A resource limited artificial immune system for data analysis[J]. Knowledge Based Systems, 2001, 14(3-4).
[4] Farmer J. D., Packard N. H., Perelson A.S. The Immune System, Adaptation, and Machine Learning. Physica 22D, 1986. pp. 187-204.
[5] Mo Hong-Wei, Zuo Xing-Quan. Artificial Immune System[M]. Beijing: Science Publishing House, 2009.
[6] Li Tao. Computer Immunology. Beijing: Publishing House of Electronics Industry, 2004. (In Chinese)
[7] Forrest S., Beauchemin C. Computer Immunology[J]. Immunological Reviews, 2007, 216(1):176–197.
[8] Somayaji A., Hofmeyr S., Forrest S. Principles of a Computer Immune System[C]. Proc New Security Paradigms Workshop. 1997.
[9] Timmis J, Neal M, Hunt J. An artificial immune system for data analysis[J]. Biosystems, 2000, 55(1-3):143-150.
[10] Harmer P K, Williams P D, Gunsch G H, et al. An artificial immune system architecture for computer security applications[J]. IEEE Transactions on Evolutionary Computation, 2002, 6(3):252-280.
[11] Kim J, Bentley P J. Towards an artificial immune system for network intrusion detection: an investigation of clonal selection with a negative selection operator[C]// Congress on Evolutionary Computation. 2001.