# WPMSD: A Malicious Script Detection Method Inspired by the Process of Immunoglobulin Secretion

**Zhao hui，Chen wen, Zeng Jie, Shi Yuanquan, Qin Jian**
*College of computer science, Sichuan University*
*Chengdu, 610065, China*
*E-mail: zhaohui@scu.edu.cn*

## Abstract

Inspired by the process of immunoglobulin secretion in biological body, we present a Web Page Malicious Script Detection Method (WPMSD). In this paper, Firstly, the basic definitions of artificial immune items are given. Secondly, according to the spreading range of malicious script, the immunoglobulin number is changed as the detector clone proliferation is stimulated by malicious scripts. Further more, the nonlinear dynamics of antibody number is discussed. Thirdly, we propose a probability approach to trigger alarms to inform that the detected scripts are harmful. Finally, the WPMSD collects the effective immunoglobulin set based on Hidden Markov Model (HMM) to update the detector gene library. Compared with the traditional immune based detection methods, such as Negative Selection Algorithm (NSA), Dynamic Colonel Selection (DynamiCS), and Variable size Detector (V-detector), the false alarm rate of WPMSD has been reduced by 18.09%, 12.6%, and 7.47% respectively.

*Keywords:* malicious script detection; false alarm rate; immunoglobulin secretion; artificial immune system.

## 1. Introduction

In order to promote the interaction between clients and web pages, many types of scripts have been embedded into the web pages. However, many malicious scripts may hide in the web page, which often lead to serious intrusion to the target computers by some web page connections from attackers. Hence, the detection of the hidden malicious scripts in the web pages is critical for the network security[1,14,15,16].

The problems in Biological Immune System (BIS) are similar to the ones in the web page malicious script detection method. In 1994, Forrest et al. [2] proposed Negative Selection Algorithm (NSA) that simulates the new immune cell tolerance process in BIS to avoid mismatching any normal script patterns. In 2002, Kim[3] presented Dynamic Colonel Selection (DynamiCS) algorithm in which the malicious scripts can be defined as the changed elements to reduce false detection. In 2009, Zhou Ji et al.[4] proposed an algorithm of Variable size Detector (V-detector) that can modify the detection radius of detector according to the changed context of web page scripts.

. In order to exactly catch harmful foreign antigens, the immune cells have to go through two stages of immune response: the primary immune response and the secondary immune response[4, 5,17]. In the primary immune response stage, once the affinity that an immune cell's matching antigen comes up to a certain threshold, the immune cell will be activated. In the secondary immune response, with the continuous expansion of antigen invasion, the immune cell clones itself and secretes much immunoglobulin to catch more antigens, which results in a fast increasing of the

Published by Atlantis Press

immunoglobulin density. After antigens have been eliminated, the immune cell clone proliferation is suppressed and the immunoglobulin density decreased simultaneously. From then on, the biological body restores into a normal health status [6,18,19].

Inspired by the process of immunoglobulin secretion in BIS, we present a Web Page Malicious Script Detection Method(WPMSD) to reduce the high false alarm rate in traditional web page malicious script detection methods based on immune system.

## 2. Method Description

The WPMSD is composed of two important parts, the initial component and the immune response component. The initial component is responsible for the detector initialization, such as the new detector generation and the negative selection of new detectors. The task of immune response component is performing the detector execution procedure, such as the detector memorization and the detector clone proliferation. The architecture of WPMSD is shown in Fig.1.

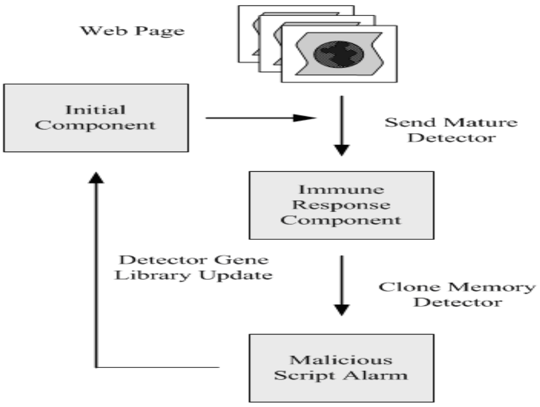

Fig. 1. The architecture of WPMSD

### 2.1. *Basic Definition*

**Definition 1.** *Antigens (Ag) are defined as the web page script patterns.*

The web page script patterns represent the *n*-dimensional vector of script activity features that are gain from web pages. The antigen feature calculation is shown in Fig. 2. Hence, antigen is given by Eq. (1).

$$Ag=\{a \mid a\subset f, f=\{f_1, f_2,\cdots, f_n\}^l, f_n \in[0,1], l>0\}. \qquad (1)$$

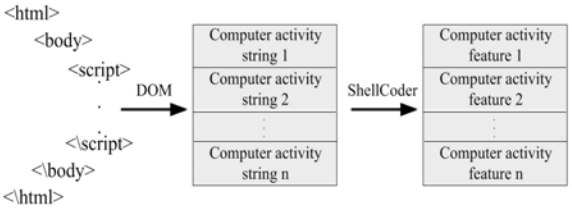
<html>
<body>
<script>
<\script>
<\body>
<\html>
DOM
Computer activity string 1
Computer activity string 2
Computer activity string n
ShellCoder
Computer activity feature 1
Computer activity feature 2
Computer activity feature n

Fig. 2. The antigen feature calculation

Where *f* represents the *n*-dimensional vector of script activity features. The structure of immunoglobulin is the same as the antigen.

**Definition 2.** *Self Antigens (Self) are defined as the normal web page script patterns.* It is given by Eq. (2).

$$Self=\{<s,rd>\mid s\subset f, rs\in R\}, \qquad (2)$$

Where *f* represents the *n*-dimensional vector of normal script activity features, *rs* is the self radius, *R* is the set of real numbers.

**Definition 3.** *Nonself Antigens (Nonself) are defined as the malicious web page script patterns.* It is given by Eq. (3).

$$Nonself=\{ns \mid ns\subset f\}, \qquad (3)$$

Where *f* represents the *n*-dimensional vector of malicious script activity features. And we have:

$$Self \bigcup Nonself = Ag. \qquad (4)$$

$$Self \bigcap Nonself = \varnothing. \qquad (5)$$

**Definition 4.** *Detectors (De) simulate immune cells in BIS for detecting the web page malicious scripts.* It is given by Eq. (6).

$$De=\{ab, rd, aff, den, cos, count, age>\mid ab\in f,$$
$$(rd, aff, den, cos)\in R, (count, age)\in N\}, \qquad (6)$$

where *ab* represents the *n*-dimensional vector of scripts activity features, *rd* is the detection radius of detector, *aff* is the affinity of detector, *den* is the density of detector, *cos* is the costimulation of detector, *count* is the matched number of a type of scripts activity, *age* is the age of detector, *R* is the set of real number, *N* is the set of natural number. The calculation of detection radius of detector is according to Eq. (7).

$$x.rd=\min\{\| x.ab-y.s \|: y.rd\}. \qquad (7)$$

Where $x \in De$ , $y \in Self$ and $\|\cdot\|$ represents the Euclidean Distance. The calculation of affinity of detector is according to Eq. (8).

$$x.aff = 1/[1+\exp(1-x.rd)]. \tag{8}$$

The affinity of detector reflects the performance of individual in the detector set. The greater detection radius a detector has, the higher affinity of this detector would have. The density of detector is based on the theory of immune networks[7], which is built on the principle that antibodies can match others as well as antigens. The calculation of density of detector is according to Eq. (9).

$$x.den = \sum_{y \in De} f_{similar}(x,y)/|De|. \tag{9}$$

Where $x \in De$ , the $\sum_{y \in De} f_{similar}(x,y)$ represents the similarity between detector *x* and detector *y*, the calculation of this similarity is according to Eq. (10).

$$f_{similar}(x,y) = \begin{cases} 1, \|x.ab - y.ab\| \le \alpha \\ 0, otherwise \end{cases}. \tag{10}$$

Where $\alpha (0 \le \alpha \le 1)$ is the similarity threshold. The calculation of costimulation of detector is according to Eq. (11).

$$x.cos = x.aff / \exp(x.den). \tag{11}$$

A detector with higher affinity and lower density will gain more co-stimulation from the system.

## 2.2. *Initial Component*

- New Detector Generation

A new detector is generated from the detector gene library. The library contains a lot of useful detector gene sequences for detecting the malicious web page scripts and these detector gene sequences can be used to generate new detectors.

- Negative Selection

In order to avoid matching any normal web page script, the new generated detector must go through the process of negative selection, as is shown in Eq. (12).

$$f_{mature}(x) = \begin{cases} 1, x.age > \omega \wedge \|x.ab - y.s\| > x.rd \\ 0, otherwise \end{cases}. \tag{12}$$

Where $x \in De$ , $y \in Self$ and $\omega$ is the period of self tolerance. If the result of Eq. (12) equals to 1, then the negative selection of the detector is successful.

## 2.3. *Immune Response Component*

The task of immune response component is to perform the detector execution procedure, including detector activation, the detector memorization, and the detector clone proliferation.

- Detector Activation

The detector activation simulates the primary immune response in BIS. When the matched number of script activity is greater than match threshold, which means the detector has found an anomaly and it will be activated. The process of detector activation is shown in Eq. (13).

$$f_{active}(x) = \begin{cases} 1, x.age \le \xi \wedge x.count \ge \phi \\ 0, otherwise \end{cases}. \tag{13}$$

Where $x \in De$ , $\phi$ is the threshold of detector activation, $\xi$ is the period of detector activation. If the result of Eq. (13) equals to 1, then the activation of the detector is successful. After that, this mature detector get activated.

- Detector Memorization

After detector activation, the activated detectors should be checked by the detector control center. When the costimulation signal from control center is greater than memory threshold, this activated mature detector will evolve into a memory detector. The process of detector memorization is shown in Eq. (14).

$$f_{memory}(x) = \begin{cases} 1, x.age \le \theta \wedge x.cos \ge \psi \\ 0, otherwise \end{cases}. \tag{14}$$

Where $x \in De$ , $\psi$ is the threshold of detector memorization, $\theta$ is the period of detector memorization. If the result of Eq. (14) equals to 1, then the memorization of the detector is successful.

- Detector Clone Proliferation

The detector clone proliferation simulates the secondary immune response in BIS: when the similar or same antigens attack the biological body again, the memory detector can be activated in short time, meanwhile, this activated memory detector starts cloning itself and secretes much immunoglobulin to catch more antigens[8].

### 2.3.1. *Immunoglobulin Number Accumulation*

In this paper, we propose the nonlinear dynamic system Eq. (15), to model the accumulation process of antibody:

$$\begin{cases} \dfrac{dDe_i}{dt} = S_{clone}R_{mem}De_i + S_{new}, \\ \dfrac{dS_i}{dt} = -S_{mem}S_i + S_{max\_secrete}S_{mem}R_{mem}. \\ \dfrac{dnum_i}{dt} = -S_{abr}den_i num_i + De_i S_i. \end{cases} \tag{15}$$

Where $De_i$ is the *ith* detector subset of a system, $S_{new}$ is the rate of generation new $De_i$ detector in system, $S_{clone}$ is the clone rate of memory detectors, $R_{mem}$ is the memorization ratio of $De_i$, $S_i$ is the rate of antibody secretion, $S_{max\_secrete}$ is the maximum rate of antibody secretion, $S_{mem}$ determines the rate of memorization, $num_i$ is the accumulation of antibody number in detector subset $De_i$, $S_{abr}$ is the rate of antibody bound by other antibodies, and $den_i$ is the detector density of detector subset $De_i$.

The Equilibrium Value of nonlinear dynamic system (15) is shown in Eq. (16).

$$\left(De_i^*, S_i^*, num_i^*\right) = \left(\frac{-S_{new}}{S_{clone}R_{mem}}, S_{max\_secrete}R_{mem}, -\frac{S_{new}S_{max\_secrete}}{S_{clone}S_{abr}den_i}\right). \tag{16}$$

From the result of Equilibrium Value of this nonlinear dynamic system, we can know that the antibody is keeping the emancipation status. However, both of the scale of detector and the number of antibody are apparently representing the tendency of decrement at the same time.

Therefore, we get the Jacobian Matrix [9] of nonlinear dynamic system. Eq. (15) is shown as following:

$$\begin{bmatrix} S_{clone}R_{mem} & 0 & 0 \\ 0 & -S_{mem} & 0 \\ 0 & 0 & -S_{abr}den_i \end{bmatrix}. \tag{17}$$

Based on the stability analysis of Lyapunov [10], we have Eq. (18).

$$L(De_i, S_i, num_i) = De_i^2 + S_i^2 + num_i^2. \tag{18}$$

Hence,

$$L = \frac{\partial L(De_i, S_i, num_i)}{\partial De_i}\frac{dDe_i}{dt} + \frac{\partial L(De_i, S_i, num_i)}{\partial S_i}\frac{dS_i}{dt} + \frac{\partial L(De_i, S_i, num_i)}{\partial num_i}\frac{dnum_i}{dt}$$

$$\begin{aligned} &= 2De_i\left(S_{clone}R_{mem}De_i + S_{new}\right) + 2S_i\left(-S_{mem}S_i + S_{max\_secrete}S_{mem}R_{mem}\right) \\ &\quad + 2num_i\left(-S_{abr}den_i num_i + De_i S_i\right) \\ &= 2S_{clone}R_{mem}De_i^2 + 2S_{new}De_i + 2S_{max\_secrete}S_{mem}R_{mem}S_i \\ &\quad + 2De_i S_i num_i - 2S_{mem}S_i^2 - 2S_{abr}den_i num_i^2. \end{aligned} \tag{19}$$

So, when

$$S_{clone}R_{mem}De_i^2 + S_{new}De_i + S_{max\_secrete}S_{mem}R_{mem}S_i + De_i S_i num_i > S_{mem}S_i^2 + S_{abr}den_i num_i^2. \tag{20}$$

The number of antibody will gradually accumulate under the stimulation of continuous network attacks.

Therefore, the result of nonlinear dynamic system (15) is given as follow:

$$De_i = \frac{e^{S_{clone}R_{mem}(t-C_1)} - S_{new}}{S_{clone}R_{mem}} \approx \frac{e^{S_{clone}R_{mem}t} - S_{new}}{S_{clone}R_{mem}}, C_1 = 0.(21)$$

$$S_i = \frac{S_{mem}S_{max\_secrete}R_{mem} - e^{S_{mem}(C_2-t)}}{S_{mem}} \approx \frac{S_{mem}S_{max\_secrete}R_{mem} - e^{-S_{mem}t}}{S_{mem}}, \qquad C_2 = 0. \tag{22}$$

$$\begin{aligned} num_i = &\frac{e^{S_{clone}R_{mem}t}S_{max\_secrete}}{e^{C_1}\left(S_{clone}^2R_{mem} + S_{clone}S_{abr}den_i\right)} \\ &- \frac{e^{C_2}e^{\left(S_{clone}R_{mem} - S_{mem}\right)t}}{e^{C_1}\left(S_{clone}R_{mem}(S_{clone}R_{mem} - S_{mem}) + S_{abr}den_i S_{clone}R_{mem}\right)} \\ &+ \frac{S_{new}e^{S_{mem}C_2}e^{-S_{mem}t}}{S_{abr}den_i S_{clone}R_{mem}S_{mem}} - \frac{S_{max\_secrete}S_{new}}{S_{clone}S_{abr}den_i} + C_3. \end{aligned} \tag{23}$$

When $C_1, C_2, C_3 = 0$, we have

$$\begin{aligned} num_i = &\frac{S_{max\_secrete}e^{S_{clone}R_{mem}t}}{S_{clone}^2R_{mem} + S_{clone}S_{abr}den_i} + \frac{S_{new}e^{-S_{mem}t}}{S_{abr}den_i S_{clone}R_{mem}S_{mem}} \\ &- \frac{e^{\left(S_{clone}R_{mem} - S_{mem}\right)t}}{S_{clone}R_{mem}(S_{clone}R_{mem} - S_{mem}) + S_{abr}den_i S_{clone}R_{mem}} \\ &- \frac{S_{max\_secrete}S_{new}}{S_{clone}S_{abr}den_i}. \end{aligned} \tag{24}$$

According to Ref. 6, the arguments in Eq. (24), including $S_{clone}$, $S_{mem}$, $S_{abr}$, and $den_i$ are shown in Eq. (25)- (28), where $S_{ai}$ is the intensity of script attacks and $|Ma_i|$ is the number of mature detectors:

$$S_{clone} \approx e^{-\frac{S_{ai}(S_{ai}+1)|Me_{isup}|}{2|Me_i|}} S_{ai}. \qquad (25)$$

$$S_{mem} \approx \left[1 - e^{-P_m \frac{|Ma_i|}{\gamma\varepsilon}(1-P_n)}\right] S_{ai}. \qquad (26)$$

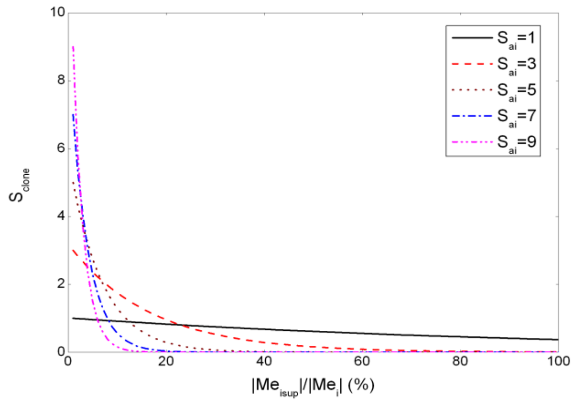

Fig. 3. The effect of $S_{ai}$ on $S_{clone}$.

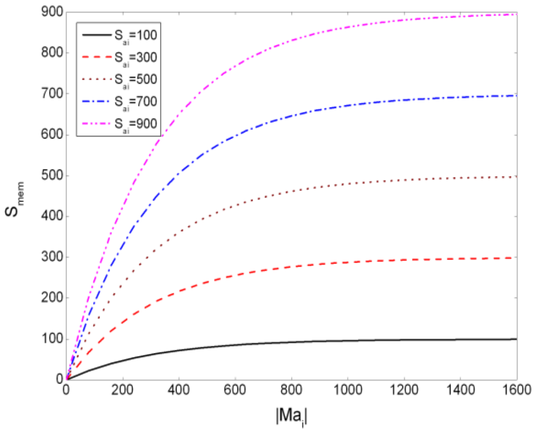

Fig. 4. The effect of $S_{ai}$ on $S_{mem}$

$$S_{abr} \approx \frac{\ln^{\left(1+\frac{n}{n-1}\right)}}{n}\left[1 - e^{-P_m \frac{|Ma_i|}{\gamma\varepsilon}(1-P_n)} + e^{-\frac{S_{ai}(S_{ai}+1)|Me_{isup}|}{2|Me_i|}}\right] S_{ai}. \quad (27)$$

$$x.den_i = \frac{1}{|De_i|}\sum_{y\in De_i} e^{\frac{-\sqrt{\sum_{i=1}^{n}(x.ab\_gene.f_i - y.ab\_gene.f_i)^2}}{x.radius - y.radius}}. \qquad (28)$$

The effect of $S_{ai}$ on $S_{clone}$, $S_{mem}$ are shown in Fig 3 and Fig 4, respectively. From Fig 3 we can see that as the increase of $|Me_{isup}|/|Me_i|$, the clone rate of memory detectors $S_{clone}$ is decreasing, mean while the higher the intensity of network attack is, the higher the value of $S_{clone}$ is, which means the clone rate of memory detectors can reflects the attacking situation.

From Fig 4 we knows that the attacking intensity $S_{ai}$ can linearly increase with the memorization rate of detectors $S_{mem}$ when the number of mature detectors $|Ma_i|$ is less than 300. $S_{mem}$ could also reflects the attacking situation of malicious script. The analyzed result is in consistent with Ref.6.

The arguments of Eq. (24) are replaced by Eq. (25), (26), (27), and Ref.6 shows that when $t$=1, $S_{clone}$: $S_{ai}$ = 1:3, therefore, the number of antibody in every time point is calculated as Eq. (29).

$$num_i = \frac{S_{max\_secrete}e^{\frac{1}{3}S_{ai}R_{mem}}}{\frac{1}{9}S_{ai}{}^2R_{mem} + \frac{\ln^{\left(1+\frac{n}{n-1}\right)}}{3n}\left[\frac{4}{3} - e^{-P_m\frac{|Ma_i|}{\gamma\varepsilon}(1-P_n)}\right]S_{ai}{}^2den_i} +$$

$$\frac{S_{new}e^{-\left[1-e^{-P_m\frac{|Ma_i|}{\gamma\varepsilon}(1-P_n)}\right]S_{ai}}}{\frac{\ln^{\left(1+\frac{n}{n-1}\right)}}{3n}\left[\frac{4}{3} - e^{-P_m\frac{|Ma_i|}{\gamma\varepsilon}(1-P_n)}\right]\left[1 - e^{-P_m\frac{|Ma_i|}{\gamma\varepsilon}(1-P_n)}\right]S_{ai}{}^2den_iR_{mem}} -$$

$$\frac{e^{\left(\frac{1}{3}S_{ai}R_{mem} - \left[1-e^{-P_m\frac{|Ma_i|}{\gamma\varepsilon}(1-P_n)}\right]S_{ai}\right)}}{A + B\cdot R_{mem}} - \frac{S_{max\_secrete}S_{new}}{B} \qquad (29)$$

Where

$$A = S_{ai}R_{mem}\left(\frac{1}{3}S_{ai}R_{mem} - \left[1 - e^{-P_m\frac{|Ma_i|}{\gamma\varepsilon}(1-P_n)}\right]S_{ai}\right) \text{ and}$$

$$B = \frac{\ln^{\left(1+\frac{n}{n-1}\right)}}{3n}\left[\frac{4}{3} - e^{-P_m\frac{|Ma_i|}{\gamma\varepsilon}(1-P_n)}\right]S_{ai}{}^2den_i$$

Hence, we simplify Eq. (29) and get the calculation of the number of antibody as follow:

$$num_i \to \omega \times \frac{e^{S_{ai}}}{{S_{ai}}^2}, \text{ where } \omega \in (0.0, 1.0). \tag{30}$$

From Eq. (30), we know that the number of antibody can keep the 0.0 to 1.0 times relationship with the intensity of network attacks.

### 2.3.2. *Immunoglobulin Number Decrement*

The immunoglobulin number synchronously reduces with the suppressed process of detector clone proliferation. The process of immunoglobulin number decrement is shown in Eq. (31).

$$num(age) = \left(1 - \frac{\lambda}{age}\right) \times num(age-1). \tag{31}$$

Where $\lambda$ is the immunoglobulin number maintain period, *age* is the age of detector.

**Theorem 1.** *Suppose immunoglobulin number is num (i) in period i, and since then, the detector fails to match any anomaly, num(i) will decrease to zero gradually.*

**Proof.** As the detectors failed to match any anomaly, there will be no detector activation and proliferation. According to Eq. (16), the immunoglobulin number is decreased as Eq. (31) shows, when detector age equals to $\lambda$ the immunoglobulin number will be zero. □

From Fig. 5 we can see that as the increase of detector age, the immunoglobulin number decreases simultaneously. Further more, we find that greater $\lambda$ could result in longer period for the depression of immunoglobulin number. So we can set large value of $\lambda$ to maintain enough number of immunoglobulin when there are high-frequency script attacking.

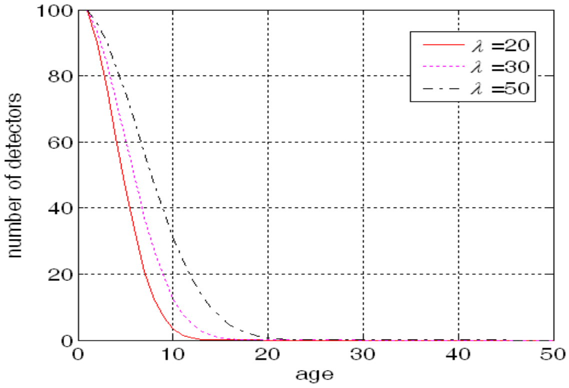

Fig. 5. The effect of $\lambda$ on num, num(0)=100.

## 2.4. *Web Page Malicious Script Alarm Production*

At time *t*, the probability of malicious script alarm production is shown in Eq. (32).

$$P_{alarm}(t) = \min\{1 : \eta \times con(t) \times allcount(t)\}. \tag{32}$$

Where $\eta$ is the adjusted argument for malicious script alarm production, *con*(*t*) is the immunoglobulin concentration at time *t*, *allcount*(*t*) is the spreading range of all web page malicious scripts at time *t*. The function of *con*(*t*) is given by Eq. (33).

$$con(t) = 1 - \frac{1}{1 + \ln^{\mu \sum_{j=1}^{m} u_j [\sum_{i=1}^{n} o_i \times num_i(t)]}}. \tag{33}$$

where $\mu$ is the adjusted argument, $u_j$ is the weight of the *jth* web page importance, $o_i$ is the weight of the *ith* type of malicious script destruction. The function of *con*(*t*) is given by Eq. (34).

$$allcount(t) = \sum_{j=1}^{m} \sum_{i=1}^{n} \sum_{x_i \in De_i} x_i.count(t). \tag{34}$$

## 2.5. *Detector Gene Library Update*

According to Hidden Markov Model (HMM) [1], we calculate the effective immunoglobulin set and update this immunoglobulin set into the detector gene library. The calculation of the effective immunoglobulin set is shown in Eq. (35).

$$P(G \mid A) = \frac{P(A \mid G) P(G)}{P(A)}. \tag{35}$$

Where $A = A_1, A_2, \cdots, A_n$ is a malicious script alarm sequence that is composed of the *n* time points. At time *t*, $A_t = A_{t1}, A_{t2}, \cdots, A_{tm}$ . $G^* = G_1, G_2, \cdots, G_n$ present a set of effective detector gene sequence which calculated the maximum result by the Eq. (36).

$$\begin{aligned} G^* &= \arg\max_G \left[P(A \mid G) P(G)\right]. \\ &= \arg\max_G \left[P(A_1, A_2, \cdots, A_n \mid G_1, G_2, \cdots, G_n) P(G_1, G_2, \cdots, G_n)\right] \\ &= \arg\max_G \left[\prod_{t=1}^{n} P(A_{t1}, A_{t2}, \cdots, A_{tm} \mid G_t) P(G_t \mid G_{t-1})\right]. \end{aligned} \tag{36}$$

Considering the entire detector gene sequences are independent from each other. Hence:

$$G^* \approx \arg\max_{G}\left[\prod_{t=1}^{n} P(A_{t1}, A_{t2}, \cdots, A_{tm} \mid G_t)\right]. \qquad (37)$$

## 3. Experiment

In order to test the performance of WPMSD, we use web page malicious scripts to test our model and compare it with the traditional immune based detection methods, including NSA[2], DynamiCS[3], V-detector[4]. There are three web pages for the performance test, which could be down loaded from Ref. 13.

Firstly, we set weights of the web pages: 0.6, 0.1, and 0.3 for Web Page1, Web Page2 and Web Page3 respectively. Meanwhile, we set weights of the malicious script destructions: 0.6, 0.2 and 0.1 for Applet, malicious JavaScript, and malicious VBScript respectively. Then we utilize the malicious Applet, malicious JavaScript, and malicious VBScript to attack the web pages. The experiment results are shown in Fig. 6 – Fig.8.

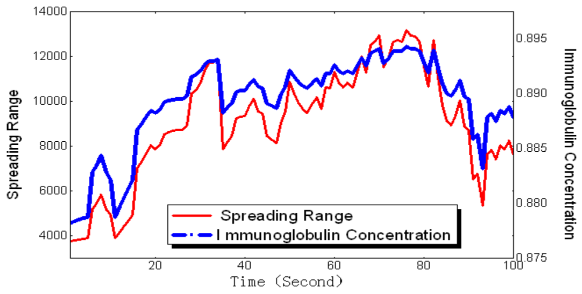

Fig.6. Malicious Applet spreading range and corresponding immunoglobulin concentration

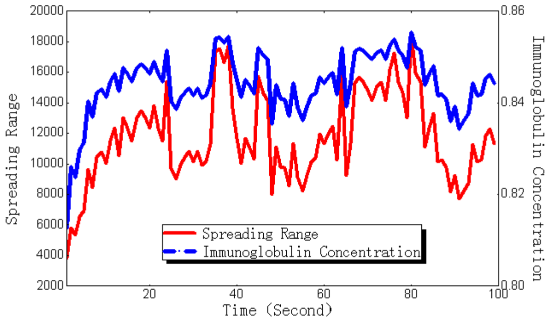

Fig. 7. Malicious JavaScript spreading range and corresponding immunoglobulin concentration.

From the above experimental results, we know that as the web page malicious script spreading range increases, the corresponding immunoglobulin concentration increases synchronously. Meanwhile, when the web page malicious script spreading range decreases, the corresponding immunoglobulin concentration also decreases synchronously. According to Eq. (32), we set the alarm threshold to be 0.5 and the probability of malicious script alarm production is shown in Fig. 9 to Fig. 11. From these figures we can see that higher threshold could greatly reduce the false alarm rate, while increase the rate of fail report. So the alarm threshold is an important adjustment for the detection performance.

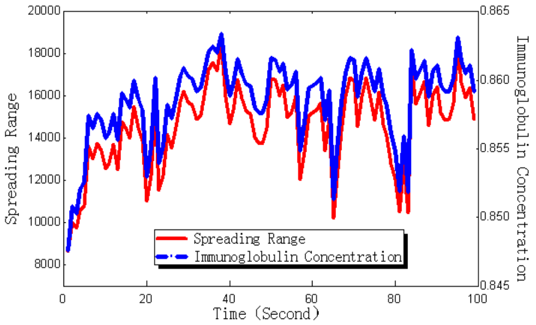

Fig. 8. Malicious VBScript spreading range and corresponding immunoglobulin concentration.

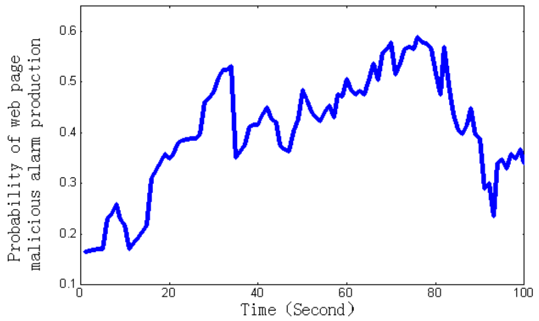

Fig. 9. The probability of Malicious Applet script alarm.

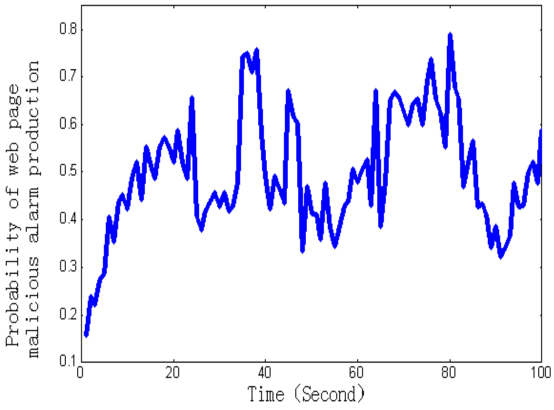

Fig. 10. The probability of Malicious JavaScript script alarm.

Published by Atlantis Press

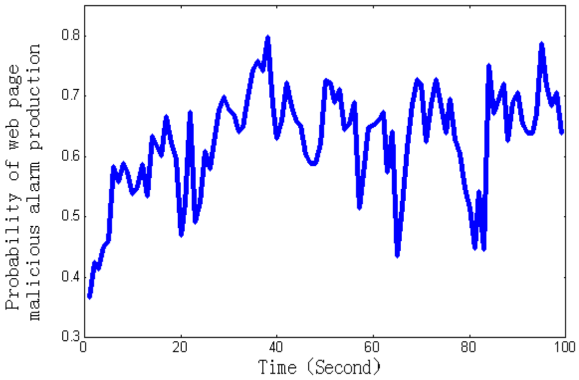

Fig. 11. The probability of Malicious VBScript script alarm production.

To compare the performance of WPMSD with NSA[2], DynamiCS[3], and V-Detector[4], the comparative experiment was also conducted on the same data set. According to Ref. 4, 11, 12 the parameters setting are shown in table 1.

The detection result is shown in table 2. As table 2， we can see that the detection performance of V-detector is better than NSA and DynamiCS, because V-detector sets the detection radius according to the changed context of web page script, which avoids the Boundary dilemma problems[4] and improves the detection rate. However, the false alarm rate of WPMSD is lower than V-detector while the detection rate of WPMSD is comparable to V-detector, further more the standard deviation of WPMSD is far less than other methods, which means its performance is more stable.

Table 1. The parameters set.

| Parmaeter | Value | Description |
|---|---|---|
| **Alarm threshold** | 0.7 | The alarm theshold of WPMSD |
| ***l*** | 49 | The legnth of antigen and detectors in NSA |
| ***R*** | 25 | The match thershod of r-continus mthoed in NSA |
| **significant level** | 95% | The maximum probability to accept Type I Error of V-detector |
| **Self radius** | 0.05 | The variation range of self antigen |

Table 2. The result of comparative experiment

| Method | Detection Rate | False Alarm Rate |
|---|---|---|
| **WPMSD** | 95.27%± 1.16% | 3.06%± 2.5% |
| **NSA** | 72.63%± 6.52% | 21.15%± 7.6% |
| **DynamiCS** | 89.29%± 3.91% | 15.66%± 5. 49% |
| **V-detector** | 97.51%± 1.77% | 10.53%± 2.63% |

## 4. Conclusion

This paper proposed a Web Page Malicious Script Detection Method inspired by the process of immunoglobulin secretion (WPMSD) to reduce the high false alarm rate in immune-based web page malicious script detection methods. We proved the relationships between antibody number and the script attacking intensity. The nonlinear dynamics of antibody number is discussed and the calculation method of antibody number is given. Compared with traditional immune-based web page malicious script detection methods, such as Negative Selection Algorithm (NSA), Dynamic Colonel Selection (DynamiCS), and Variable size Detector (V-detector), the false alarm rate of WPMSD has been reduced dramatically, while the high detection rate is maintained.

## Acknowledgements

This work is sponsored by National Natural Science Foundation of China (No.60873246) and the National Research Foundation for Doctoral Program of Higher Education of China (No. 20070610032)

## References

1. J. L. Chen, P. Zhong, T. Cook, “Detecting web content function using generalized hidden markov model,” Proceeding of International Conference on Machine Learning and Applications, pp. 279–284, 2006.
2. S. Forrest, A. S. Perelson, “Self-nonself discrimination in a computer,” Proceedings of IEEE Symposium on Security and Privacy, 1994, pp. 202–212.
3. J. Kim, P. J. Bentley, “Towards an artificial immune system for network intrusion detection: An investigation of dynamic clonal selection,” Proceeding of IEEE on Evolutionary Computation, pp. 1015–1020, 2002.
4. Zhou Ji, D. Dasgupta. V-detector: An efficient negative selection algorithm with “probably adequate” detector coverage. Information sciences, 2009, pp.1390-1406
5. Perelson AS, Weisbuch G, Immunology for physicists. Rev Mod Phys 69(4)1997, pp. 1219-1267.
6. Jie Zeng, Tao Li, Guiyang Li, Haibo Li, and Jinquan Zeng. “A novel intrusion detection approach learned from the change of antibody concentration in biological immune response”. Applied Intelligence, Springer. 2010.
7. N. K. Jerne, “Towards a network theory of the immune system,” Annales D’immunologie, 1974, pp. 373–389.
8. M. F. Bachmann, U. Kalinke, A Althage, “The role of immunoglobulin concentration and avidity in antiviral protection,” Science, 1997, pp. 2024–2027.
9. M. X. Kong, Y. Zhang, Z. J. Du, L. N. Sun, “A novel approach to deriving the unit-homogeneous jacobian

matrices of mechanisms," Proceeding of International Conference on Mechatronics and Automation, 2007 , pp. 3051–3055.

10. H. C. Zou, J. W. Lei, H. Y. Yu, "Extended lyapunov stability theorem and its applications in control system with constrained input," Proceeding of International Symposium on Computer Network and Multimedia Technology, 2009, pp. 1–4.
11. Chien-Cheng Chang, Hwai-En Tseng, Ling-Peng Meng,Artificial immune systems for assembly sequence planning exploration,Engineering Applications of Artificial Intelligence, Volume 22, Issue 8, December 2009, pp.1218-1232
12. Gonzalez,F., D.Dasgupta, L.F.Nino, A randomized real-valued negative selection algorithm. in Proceedings of the 2nd International Conference on Artificial Immune Systems (ICARIS), 2003, pp.261-272.
13. http://homepage.tudelft.nl/n9d04/occ/index.html
14. fault detection: A Negative Selection Approach, Expert Systems with Applications, Volume 37, Issue 7, July 2010, pp.5507-5513
15. Aydin I, Karakose M, Akin E. A multi-objective artificial immune algorithm for parameter optimization in support vector machine[J]. Applied Soft Computing, 2011,11(1):120-129.
16. Yuanquan Shi, Xiaojie Liu, Tao Li, Xiaoning Peng, Wen Chen, Ruirui Zhang, Yanming Fu, Chaotic Time Series Prediction Using Immune Optimization Theory, International Journal of Computational Intelligence Systems, Vol.3(S1), p43-60, 2010
17. Peng Lingxi, Li Tao, Liu Xiaojie, et al. An immune system-inspired paradigm for anomaly detection[J]. Journal of Computational and Theoretical Nanoscience, 2007, 4(7-8): 1394-1398.
18. Lingxi Peng, Tao Li, Xiaojie Liu, Caiming Liu, Jinquan Zeng, Jian Zhang. An Artificial Immune Network Based Algorithm for Diabetes Diagnosis, Protein and Peptide Letters
19. Secker A, Freitas AA, Timmis J. AISIID: An artificial immune system for interesting information discovery on the web[J]. Appl Soft Comput, 2008, 8(2): 885-905.