International Journal of Security and Its Applications
Vol. 10, No. 3 (2016), pp.151-160
http://dx.doi.org/10.14257/ijsia.2016.10.3.14

Transfer Model Based on State of Finite Semi-Markov Automata Intrusion Tolerance

Wang Guangze, Wang Peng, Luo Zhiyong and Zhu Suxia

Harbin University of Science and Technology, Harbin 150080, Heilongjiang Province, China
wangguangze_hust@sina.com

Abstract

Existing network security technology may not against most of intrusion so that we need to study intrusion tolerance technology. On the basis of existing model of intrusion tolerance, we putted forward an optimization of finite automata state transition model in intrusion tolerance system by adding strategy and updating status. Since the conversion process between the states of the model meets Semi-Markov theory, therefore, the model can be used to quantify this theory; it gives calculation process of the steady probability by each state. By stabilizing the probabilistic analysis of each state, provide theoretical guidance and basis for network management personnel to better maintain network security .Proven, the Semi-Markov applied to finite automata intrusion tolerant system is feasible, effective, and has simple features.

***Keywords**: Intrusion Tolerance; State Transition; Finite Automata; Semi-Markov Process*

1. Introduction

Traditional security work can be attributed to two aspects: preventing the occurrence of invasion and solving the system security vulnerabilities[1]. Since it is impossible to predict all the unknown intrusions form and eliminate the presence of the new security vulnerabilities completely, this will cause some intrusion[2]. Therefore, it is necessary to research a system which is hacked that can still function, which is intrusion tolerant system.

Intrusion tolerance technology as the core of the third generation network security technology, it acknowledges the existence of loopholes in the system, and assuming some of these vulnerabilities could be exploited by intruder over time[3]. Its design goal is to make the system in case of error or being invaded, able to ensure critical functions continue running and implement key systems continue providing service (possibly based downgrade mode). Since this method is not only makes consideration of protection to the system availability, but also considered in both system data and service's confidentiality , integrity and other security attributes, it is possible to achieve the purpose of preventive measures, and it has been called System Security protection of the last line of defense[4].

Based on SITAR[5] intrusion tolerant system architecture, this paper increased policy update status, and presenting an optimized finite automata state transition model Intrusion Tolerance System. Due to the finite automaton in the transfer system between the states meet Semi-Markov characteristics, so it used Semi-Markov theory to quantify, calculating the probability of each steady state, in order to determine the critical nodes of the system. By increasing the runtime of system key nodes, increase the difficulty of system intrusion to provide network management personnel to effectively maintain the network with work direction and theoretical basis.

ISSN: 1738-9976 IJSIA

2. Optimization of Intrusion Tolerance System State Transition Model

Since the object intrusion tolerance system protect is multifarious[6], so the system framework, intrusion tolerance policy, security algorithms that each intrusion tolerance system used are different, in order to facilitate the abstract description of the dynamic behavior of intrusion tolerant systems, this paper based on SITAR intrusion tolerant system architecture, increased policy update status, presented an optimized finite automaton state transition model, the model shown in Figure 1.

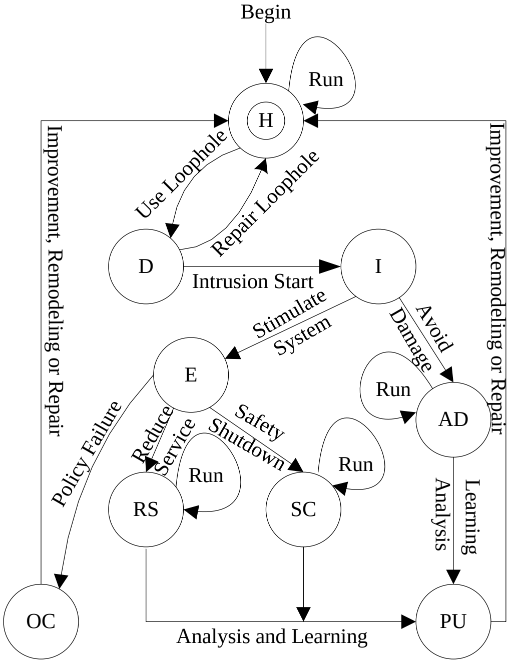


Figure 1. State Transition Model of Intrusion Tolerant Systems

The state transition model by the impact of various attacks on system services generally describes the events, state and treatment that a generalized intrusion tolerance system may occur when resisting the invasion. State model shift in the basic state include: *H* (Health State), *D* (Dangerous State), *I* (Invasion State), *AD* (Avoid Damage State), *E* (Excited State), *RS* (Reduce Service Status), *SC* (Safety Closed State), *OC* (Out of Control State). When the system is in a state that avoids damage, reduces service and shuts down safely, you can enter the intrusion analysis study to update the status of learning strategies, identify the type of the attack and stored, so that it can make quick response for the next attack. When the system is in a non-healthy state, and sometimes can be automatically restored to the *H* state, sometimes you need to manually recover.

Effect of some unknown attacks as long as the service provided to the system caused by similarity to known state, it can be treated with the state transition model. Therefore, the model can handle unknown forms of attack. System state are divided into several levels, each state can take the appropriate security policies to ensure the healthy operation of the system, so the model has some flexibility and security too.

3. Finite Automata Analysis Intrusion Tolerance System

Finite state automaton is a control limited and finite symbol automaton, which is divided into deterministic finite automata (DFSA)[7] and non-deterministic finite automata (NDFSA).

As shown in Figure 1, with intrusion tolerant system running, the system transitions are from one state to another state, these states may be healthy states, may be carrier states, different system states represent different meanings. At some point, have some kind of presence state and system corresponding to the determined, the system's running will be ultimately terminated state in any case, therefore the state of the system is limited, so finite automata intrusion tolerance system could be described by finite state automaton. Because the intrusion tolerance system has a non-deterministic finite automata characteristics, namely: in the case of a given state and symbol, not uniquely determine the next state. So the paper adopted non-deterministic finite automata theory[8] to study the formal description method of intrusion tolerance system.

3.1. NDFSA Intrusion Tolerance System

A non-deterministic finite automaton NDFSA is a quintuple NDFSA = (Φ, Σ, *dom*, s_0, *ED*), where:

Φ is a non-empty finite set of states, in which each element becomes a state;

Σ is a non-empty finite input alphabet, in which each element becomes an input character;

Mapping *dom* of $\Phi \times \sum \rightarrow \Phi$ subset, namely *dom* is a multi-valued mapping;

$\mathrm{s}_0 \subseteq \Phi$ is a non-empty set;

$\mathrm{ED} \subseteq \Phi$ is an end state set of Φ, the desirability of a null value.

If the automaton is in state *s*, and enters the character δ, the system switches to the state s', then note $dom(s,\delta)=s'$.

Non-deterministic finite automata working status is available to convert the state transition table and graph. Suppose there are *M* System Status nodes and n conversion conditions, then the state transition diagrams contain the *M* state switching nodes, the maximum value of each node is *n*, each arc condition marked by an input, each state transition figure includes a unique system initial node and several systems terminating node.

According to Figure 1, it can be non-abstract model of intrusion tolerance system deterministic finite automata NDFSA = (Φ, Σ, *dom*, s_0, *ED*), where $\Phi = \{H, D, I, AD, E, RS, SC, PU, OC\}$; $\Sigma = \{0,1, \varepsilon\}$, 1 and 0, respectively, intrusion tolerance system security policy successes and failures, ε represents the empty shift; $s_0 = \{H\}$; $ED = \{H\}$.

Dom mapping: $\Phi\times\Sigma\rightarrow\Phi$ is:

$dom(H,0)=D$, $dom(H,1)=H$; $dom(D,0)=\mathrm{I}$, $dom(D,1)=H$;

$dom(I,0)=AD$, $dom(I, \varepsilon)=E$; $dom(AD,0)=PU$, $dom(AD,1)=[AD,PU]$;

$dom(E,0)=OC$, $dom(E,1)=[RS,SC]$; $dom(RS,0)=PU$, $dom(RS,1)=[RS,PU]$;

$dom(SC,0)=PU$, $dom(SC,1)=[SC,PU]$; $dom(OC,1)=H$; $dom(PU,1)=H.$

The intrusion tolerance model non-deterministic finite automata state transition table is shown in Table 1.

Table 1. Intrusion Tolerance Non-deterministic Finite Automata System State Transition

Status	Enter Alphabet: Σ		
	0	1	ε
H	*D*	*H*	
D	*I*	*H*	
I	*AD*		*E*
AD	*PU*	[*AD*,*PU*]	
E	*OC*	[*RS*,*SC*]	
RS	*PU*	[*RS*,*PU*]	
SC	*PU*	[*SC*,*PU*]	
OC		*H*	
PU		*H*	

The non-intrusion tolerance system model deterministic finite automata state transition diagram is shown in Figure 2.

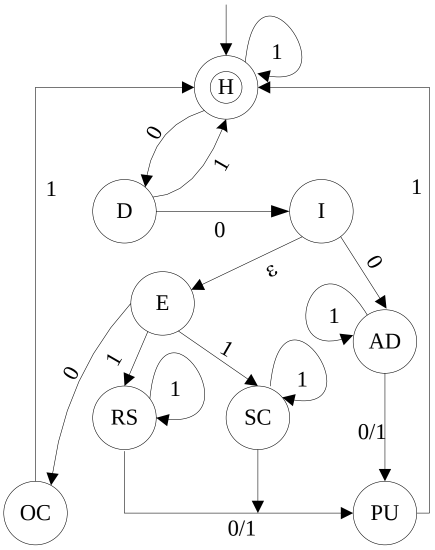


Figure 2. Intrusion Tolerance System Non-deterministic Finite Automata State Transition Diagram

Figure 2 depicts a reflection of the dynamic behavior of intrusion tolerant systems framework. System uses a variety of strategies to support and maintain the different levels of security requirements; state transition model represents the corresponding measures actual intrusion and system security requirements[9~11].

3.2. Intrusion Tolerance System Finite Automata Working Process

System initial work in health state *H*; because of its inherent vulnerability, the system is very easy to enter the dangerous state *D*. At this point, the intruder has not caused injury to applications and services, if you can fix vulnerabilities in a timely manner, the system will revert to the state *H*; if this intrusion bypasses protective measures, the system will enter the invaded state *I*. In case *I*, some parts or features of the application server has been compromised, the damage may be static, one-time, there may be dynamic, persistent. When the intrusion is not detected, because we have been prepared some fault tolerant measures in the design of the system, which can be controlled and the elimination of such damage, the system will enter the state avoid damage *AD*; after the intrusion is detected, intrusion tolerance mechanism is activated, the system will enter the excited state *E*, in this state, the system can enter the state needed to reduce service *RS* or safety off *SC*; if the intrusion tolerance mechanisms fail, the system will enter into the runaway state *OC*, it should alarm immediately restore manually by an administrator. When the system is returned by the *AD*, *RS* and *SC* status to *H* state, *PU* state intrusion can be analyzed to suffer learning and updating, thereby enhancing the anti-intrusion system's ability to prepare for the defense of the future invasion.

4. Semi-Markov Process of Finite Automata Quantization

4.1. State Transition Model DTMC

Because Figure 2 shows a non-deterministic finite automata state transition Markov meet half theoretical relationship, s_o it can be quantified, convenient for further analysis. The finite state automaton further defined as Φ={*H*, *D*, *I*, *AD*, *E*, *RS*, *SC*, *PU*, *OC*}, s_0={*H*}, *ED*={*H*}, represented by P_i corresponding state transition probability wherein $i \in \Phi$, the finite automata by the input table space $\Sigma\{0,1, \varepsilon\}$ further quantified as {P_h, P_{hd}, P_{dh}, P_{di}, P_{ie}, P_{iad}, P_{eoc}, $P_{esc,}$ P_{ers}, P_{ad}, P_{rs}, P_{sc}, P_{rspu}, P_{scpu}, P_{adpu}, P_{puh}, P_{och}}, finite state automaton quantized conversion process is shown in Figure 3.

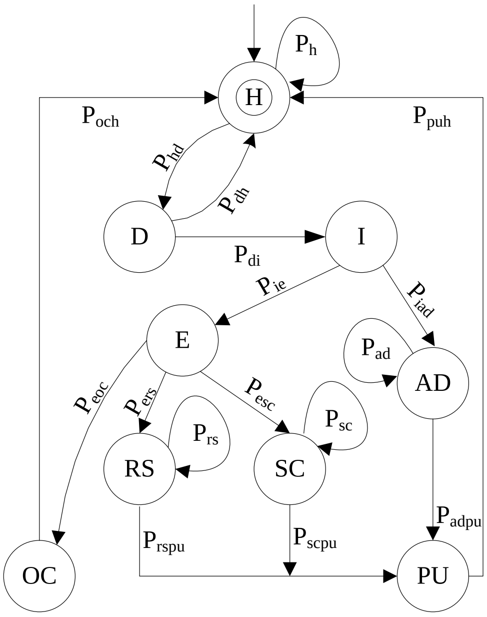


Figure 3. Semi-Markov Process of Finite Automata Quantization

According to Semi-Markov theory [12], known finite automaton state transition probability matrix P as follows:

$$P=\begin{array}{c|ccccccccc|}
 & H & D & I & E & AD & RS & SC & OC & PU \\
\hline
H & P_h & P_{hd} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
D & P_{dh} & 0 & P_{di} & 0 & 0 & 0 & 0 & 0 & 0 \\
I & 0 & 0 & 0 & P_{ie} & P_{iad} & 0 & 0 & 0 & 0 \\
E & 0 & 0 & 0 & 0 & 0 & P_{ers} & P_{esc} & P_{eoc} & 0 \\
AD & 0 & 0 & 0 & 0 & P_{ad} & 0 & 0 & 0 & P_{adpu} \\
RS & 0 & 0 & 0 & 0 & 0 & P_{rs} & 0 & 0 & P_{adpu} \\
SC & 0 & 0 & 0 & 0 & 0 & 0 & P_{sc} & 0 & P_{scpu} \\
OC & P_{och} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
PU & P_{puh} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
\end{array}$$

Matrix P describes the possibility of the system among state transition, in which the probability value can be determined empirically or by way of injection invasion assay. According to the actual situation and Semi-Markov theory, in matrix P, there are the relationships: P_{dh}=1-P_{di}, P_{ie}=1-P_{iad}, P_{eoc}=1-P_{ers}-P_{esc}.

After the introduction of the state transition probability matrix P, finite automata model satisfies the relationship: $dom(s, P_i) = \Phi \bullet P$, where: $s \in \Phi$, $P_i \in \Sigma$.

4.2. Stable Probability Finite Automata

Steady state probability means the probability distribution of each system in a stable state. The probability of the system in a state of steady DNFS is described by Ψ_i , ∂_i represent the steady-state probability of DNFS state, $\bar{\partial}=\left[\partial_H,\partial_D,\partial_I,\partial_E,\partial_{AD},\partial_{RS},\partial_{SC},\partial_{OC},\partial_{PU}\right]$; then use hi to represent a state with an average hold time; P is DNFS state transition probability matrix. Ψ_i calculated as formula (1), ∂_i satisfies the equation (2).

$$\Psi_i=\frac{\partial_i h_i}{\sum_j \partial_j h_j}, i, j \in \Phi$$

(1)

$$\begin{cases} \bar{\partial}=\bar{\partial}P \\ \sum_i \partial_i = 1, i \in \Phi \end{cases}$$

(2)

State average hold time h_i at all random time in the state is determined by the model, which time and other technical means in turn the attacker's technical capabilities and systems used in the decision, so the use of the average state hold time to simplify the analysis of the model. Order {h_H, h_D, h_I, h_E, h_{AD}, h_{RS}, h_{SC}, h_{OC}, h_{PU}} respectively for the state average hold time, and the introduction of the parameters H, according to formula (1) and (2) to calculate each state finite automaton stable probability as follows:

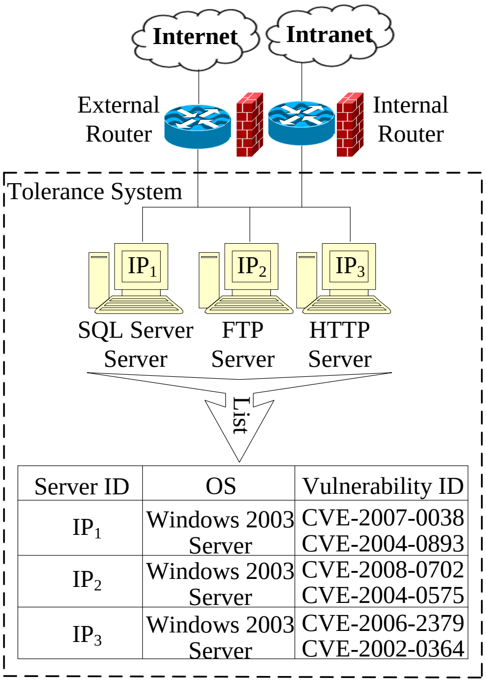
$$\Psi_H=(P_{och}+P_{puh}+P_{hd}P_{dh})h_H/H;\Psi_D=P_{hd}h_D/H;\ \Psi_I=P_{hd}P_{di}h_I/H;$$
$$\Psi_E=P_{hd}P_{di}P_{ie}h_E/H;\Psi_{AD}=P_{hd}P_{di}P_{iad}h_{AD}/H;\ \Psi_{OC}=P_{hd}P_{di}P_{ie}P_{eoc}h_{OC}/H;$$
$$\Psi_{RS}=P_{hd}P_{di}P_{ie}P_{ers}h_{RS}/H;\ \Psi_{SC}=P_{hd}P_{di}P_{ie}P_{esc}h_{SC}/H;$$
$$\Psi_{PU}=(P_{hd}P_{di}P_{iad}P_{adpu}+P_{hd}P_{di}P_{ie}P_{ers}P_{rspu}+P_{hd}P_{di}P_{ie}P_{esc}P_{scpu})h_{PU}/H;$$
$$H=(P_{och}+P_{puh})h_H+P_{hd}(h_D+P_{dh}h_H+P_{di}(h_I+P_{iad}(h_{AD}+P_{adpu}h_{PU})$$
$$+P_{ie}(h_E+P_{eoc}h_{OC}+P_{esc}(h_{SC}+P_{scpu}h_{PU})+P_{ers}(h_{RS}+P_{rspu}h_{PU}))))$$

Stable probability of DNFS model is an important indicator to measure the resistance tolerance system invaders capabilities in each state. Larger stable probability in model, means the longer run, the greater the price paid, higher reliability of the system, when the intruder makes the system shut by the state.

5. Numerical Analysis Results

5.1. Test Network Environment

The DNFS model designed in this paper, its network topology and server software configurations and vulnerabilities shown in Figure 4. IP_1 to IP_3 figure server form an intrusion tolerant system under the control of the firewall policy that provides the appropriate network services to users inside and outside the network hosts. Organizing students to simulate an intruder attack the tolerant systems, thereby obtaining test data.



Server ID	OS	Vulnerability ID
IP_1	Windows 2003 Server	CVE-2007-0038 CVE-2004-0893
IP_2	Windows 2003 Server	CVE-2008-0702 CVE-2004-0575
IP_3	Windows 2003 Server	CVE-2006-2379 CVE-2002-0364

Figure 4. Network Topology of DNFS Model

5.2. Finite Automata System Data Calculation and Analysis

Through the analysis of test data and recommends that combine CVSS using statistical estimates manner, obtain the parameter values as follows:

We describe DNFS state transition probability with matrix P. Because the system is always considered to be in a normal state operation, and the system is in states H, RS, SC, AD will be able to run, namely: $P_h=P_{ad}=P_{rs}=P_{sc}=1$, but for the convenience of study, let ignored, and even if: $P_h=P_{ad}=P_{rs}=P_{sc}=0$. In addition, when the system maintenance and management personnel will be re-run, so $P_{puh}=P_{och}=P_{adpu}=P_{rspu}=P_{scpu}=1$. Because the system is bound to be some loopholes and found by the intruder, therefore $P_{hd}=1$. According to a lot of loopholes in each server shown in Fig.4, the probability of these vulnerabilities being successfully exploited by intruders is $P_{di}=0.5$; the probability of the system detects vulnerabilities and timely repair of $P_{dh}=1\text{-}P_{di}=0.5$. The probability of the system be invaded and successfully evade intrusion is $P_{iad}=0.4$; the probability of the system detects the presence of the invasion and successfully triggers intrusion tolerant systems is $P_{ie}=1\text{-}P_{iad}=0.6$; the probability of the system detects the presence of the invasion and continues to operate but provides downgrade services is $P_{ers}=0.5$; the probability of the system detects the presence of the invasion and stops running is

P_{esc}=0.4; the probability of invasion make the system in troubles and the system stops running is P_{eoc}=1-P_{ers}-P_{esc}=0.1.

The experience shows that the system works in state *H* relatively for long time. Here are located h_H=1.0, h_D=1.8; when the system finds were transferred to the state after the invasion of *AD* and state *E*, the state of *AD* will enter state *H* through learning, so h_I =0.4, h_{AD}=0.5, h_{PU}=0.5; state *E* in accordance with intrusion tolerance policy decides the direction of system transferring, so make h_E=0.2, h_{RS}=4.0, h_{SC}=1.5, h_{OC}=2.5. It should be noted that all h_i ($i \in \Phi$) variables are time units.

Depending on the state system of transition probabilities and the average holding time, you can get DNFS model parameters for each state, as shown in Table 2:

Table 2. DNFS Model Parameters of Each State

State Node *i*	Duration h_i	Steady State Probability Ψ_i	Probability Value
H	1.0	Ψ_H	0.4348
D	1.8	Ψ_D	0.3130
I	0.4	Ψ_I	0.0348
E	0.2	Ψ_E	0.0104
AD	0.5	Ψ_{AD}	0.0174
RS	4.0	Ψ_{RS}	0.1043
SC	1.5	Ψ_{SC}	0.0313
PU	0.5	Ψ_{PU}	0.0409
OC	2.5	Ψ_{OC}	0.0130

As can be seen from Table 2, each of the intermediate steady state probability of the duration of the overall impact on the system in descending order as: {*H*, *D*, *RS*, *PU*, *I*, *SC*, *AD*, *OC*, *E*}. Since the state *PU* and *OC* need to manually adjust the system to return to the state *H*, and therefore the state of *OC* and *PU* stable regardless of the probability of network security maintenance, negligible. If the increase in intermediate state {*H*, *D*, *RS*}, it can effectively increase the cost of the invasion, and enhance the reliability of the system.

6. Conclusion

Intrusion tolerance technology is an important network security management technology, which is a technology of ensure network keep operating after the invasion, therefore intrusion tolerance technology is a heated topic in recent research today. This is the basis of the SITAR intrusion tolerant system structure, increasing the attack to learn the status of proposed optimization of finite automaton state transition model. Since converting the model each state meet Semi-Markov theory, so the model was quantified to calculate the probability of a stable finite automaton each state.

Finally, through the analysis of the test data, we obtained increased intermediate model state {*H*, *D*, *RS*} of duration may increase the difficulty of intrusion. Further research will focus on further improving the system, increasing the online repairing tolerant system, reducing the system stop state, improves system availability.

Acknowledgements

Supported by National Natural Science Foundation of China (NO: 61502123) and Heilongjiang Provincial Youth Science Foundation (NO: QC2015084).

References

[1] R. K. C. Chang, "Defending Against Flooding-based Distributed Denial of Service Attacks: A Tutorial", IEEE Communications Magazine, vol. 40, no. 10, **(2002)**, pp.42-51.

[2] Z.-Y. Luo, Y. Bo, X. Jia-zhong, Y. Gui-xin and L. Ya-hui, "Attack Graph Algorithm in the Application of Intrusion Detection System", International Journal of Security and Its Applications, vol. 5, no. 7, **(2013)**, pp. 249-256.

[3] Y. Li-hua, F. Bin-xing, "Security Attributes Analysis for Intrusion Tolerant Systems", Chinese Journal of Computers, vol. 29, no. 8, **(2006)**, pp. 1505-1512.

[4] F. Gong, "Characterizing Intrusion Tolerant Systems Using a State Transition Model", Proceedings of t he DARPA Information Survivability Conference and Exposition (DISCEX II), **(2001)**.

[5] K. S. Trivedi, "Probability and Statistics With Reliability, Queuing, and Computer Science Applications, 2nd Edition", New York: John Wiley and Sons, **(2002)**.

[6] L. H. Yin and H. Song, "Research and Implement on the Intrusion Tolerant System", Journal on Communications, vol. 27, no. 2, **(2006)**, pp. 131-136.

[7] B. Madan, K. G. Popstojanova, K. Vaidyanathan and K. S. Trivedi, "A Method for Modeling and Quantifying the Security Attributes of Intrusion Tolerant Systems", Performance Evaluation, vol. 56, no. 1-4, **(2004)**, pp. 167-186.

[8] W. L. Peng, L. N. Wang and H. G. Zhang, "Research of Intrusion Tolerant System Based on Finite State Automaton Machine", Mini-micro Systems, vol. 26, no. 8, **(2005)**, pp. 1296-1300.

[9] L. B. Chen, J. H. Jiang, D. Q. Zhang and C. Y. Shuai, "Intrusion Tolerant System Based on Multi-version Redundant Processes", Journal of Tsinghua University (Science and Technology), vol. 51, no. 1, **(2011)**, pp. 1519-1526.

[10] T. Wang, H. Yan, S. Zhong and Y. Zhang, "Research of Fire Alarm System Based on Extension Neural Network", International Information and Engineering Technology Association, vol. 1, no. 2, **(2015)**, pp. 9-16.

[11] Y. Yue, J.-H. Xiao and S.-Y. Luo, "A Practice Guide of Predicting Resource Consumption in A Web Server", International Information and Engineering Technology Association, vol. 4, no. 2, **(2015)**, pp. 1-6.

[12] Z. Feng, J. Hai, J. Li and P.-P.Yuan, "VFRS: A Novel Approach for Intrusion Tolerance in Virtual Computing Environment", Journal of Computer Research and Development, vol. 47, no. 3, **(2010)**, pp. 493-499.

Authors

Wang Guang-ze, He received a bachelor's degree in Mechanical Engineering (1987) from the University. Now he is an associate researcher at the university library. His current research interests include different aspects of computer network technology and network security.