Method Based on GSCPN for Network Vulnerability Analysis

Xiang Gao
State Key Laboratory of Mathematical Engineering and Advanced Computing
Zhengzhou, 450002, P.R. China.
E-mail: feiyu4321@163.com

Yue-fei Zhu, Jin-long Fei, Tao Han
State Key Laboratory of Mathematical Engineering and Advanced Computing
Zhengzhou, 450002, P.R. China.
E-mail: {zyf, fjlong,hantao}@yahoo.com.cn

***Abstract*— With the development of network security research, network attack modeling and analysis techniques have been paid more and more attention. A generalized stochastic colored Petri Net (GSCPN) Model is proposed. To each attack, a GSCPN model is constructed to describe the relation of components graphically. Algorithm to construct a composite attack and method for network vulnerability analysis are delivered also. The exploitation cost of vulnerabilities is estimated. The method is relatively simple, which is different from traditional method. The network example further validates the proposed method for network vulnerability analysis.**

***Index Terms*—security assessment, GSCPN, combined attack, vulnerability analysis**

I. INTRODUCTION

With the rapid development of computer network, the network dependence of the people is strengthening gradually, and the information security problem has become particularly prominent. The computer viruses and hacker attacks cause immeasurable loss to users and businesses, so we must take effective measures to ensure the safe operation of the computer network. Traditional passive security defense technology such as intrusion detection and firewalls already could not satisfy the demand of people. Many scholars at home and abroad have been interested in studying active security analysis as well as assessment methods, and the network attack modeling and analysis technology are the foundation of network security assessment.

Currently, in terms of network attack modeling, people have achieved some results. The familiar models include attack tree model [1], attack graph [2, 3], vulnerability state diagram [4], threats propagation model [5], game model [6,7,8], vulnerabilities exploiting graph [9] and Hidden Markov Model [10]. They reflect the state change of the attacker and the network system from different angles, but these models lack of ability to describe concurrent and collaborative attack process for the combined network attacks. By contrast, Petri net is graphics-based mathematical modeling tool, and has more advantages, such as semantic normalization, strong expression ability. It is more conducive to describe the process of network attacks. In addition, recently most security assessments based on the model use the method by analyzing success probability of attack sequence [11, 12]. The drawback is that calculating the maximum success probability of invasion easily generates extreme analysis results. If there is a situation with unreasonable setting of probability, it would make the results large deviate. So the researchers try to analyze network security from the angle of attack and defense cost.

According to the above problems, this paper gives a generalized stochastic and colored Petri net model (GSCPN) which is the combination of generalized stochastic Petri net [13] and colored Petri net [14]. The model can clearly describe the behaviors of combined attacks and represent related attributes of attacks with color sets. It is particularly suitable for concurrent and collaborative attacks. Furthermore, we conduct the quantitative evaluation and analysis by evaluating performance of system, avoiding the problem of analyzing success probability of attack sequence. Here, generalized stochastic Petri net [15] is introduced. It is an extension of stochastic Petri net that the transitions are divided into instant transitions and timed transitions, and it is more suitable for network attacks modeling.

Section 2 describes in detail the related definition of GSCPN model, combination operations of attacker behaviors and the basic thought of model building. Section 3 proposes one kind of the best attack path algorithm based on GSCPN model. Section 4 validates the proposed method through a case. At last, there is the conclusion.

This work is supported by National Natural Science Foundation of China (60902102), Zhengzhou Science and Technology Innovation Team Project (10CXTD150).

Corresponding author: Xiang Gao

doi:10.4304/jsw.8.8.2032-2038

II. PROPOSED MODEL

A. Related Definition

Definition 1: A Petri net is a triple $N=(P,T,F)$ where:

① P is a set of states, called places.

② T is a set of transitions.

③ F where $F \subset (P\times T)\cup(T\times P)$ is a set of flow relations called "arcs" between places and transitions (and between transitions and places). A Petri net is a bipartite graph, where P is one partition and T is the other. Moreover, for every t in T there exist p and q in P so that (p,t) and (t,p) are in F and for every p and q in P, if (p,t) and (t,p) are in F then $p \neq q$.

The set $P \cup T$ are the net elements. The set of places define the local states of a net, however, the global state of a net can be defined by place subsets.

In the aspects of graphical indication, P is represented by a circle, T is represented by a square or a rectangular, the flow relationship between the elements is represented by arrowed arc. The method is as follows:

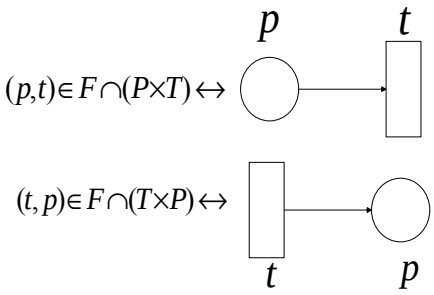

Figure 1. Flow relationship between the elements

Definition 2: Generalized Stochastic Colored Petri Nets is a nine-tuple:

$$GSCPN=(\sum,P,T,F,C,G,E,\lambda,M_0,I)$$

where:

(1) $\sum$ is a finite set of non-empty types, also called color sets.

(2) P is a finite set of places.

(3) T is a finite set of transitions, $T=T_t\cup T_i$, $T_t\cap T_i=\phi$, T_t denotes timed transitions set, T_i denotes instant transitions set.

(4) F is a finite set of arcs such that: $F\subseteq P\times T\cup T\times P$, and the arc only exists between P and T.

(5) C is a color function, $C:P\to\sum$.

(6) G is a guard function, $G:T\to BoolExpression$. It is defined from T into expressions satisfying $\forall t\in T:[Type(G(t))=Boolean\wedge Type(Var(G(t)))\subseteq\Sigma]$.

(7) E is an arc expression function, $E:F\to FE$,satisfying $\forall f\in F,[Type(E(f))=C(p)_{MS}\wedge Type(Var(E(f)))\subseteq\Sigma]$, $C(p)_{MS}$ denotes the multi-sets of $C(p)$.

(8) λ is average implementation rate of timed transition, or priority set between instant transitions.

(9) M is marking set, M_0 usually denotes the initial marking, represents starting position of the stack.

(10) I is an initialisation function, $I:P\to\sum$, assigning to the initial color for each place.

In the definition above, $Type(x)$ denotes the type of x value, $Boolean$ denotes boolean variable with True or False, $Var(x)$ denotes that x is one variable.

Definition 3: $\sum$ (**Color Sets**) is defined as

```
Color Host = string;
Color Vul = string;
Color AttackCons=SrcHost*DstHost*Perms;
Color SrcHost = Host;
Color DstHost = Host;
Color Perms = {anonymous, guest, root/admin};
Color AttackRes = {root access, crash, confident,
                   compromised…};
Color Condition = BoolExpression;
Color Boolean = {true, false}.
```

Among them, AttackCons is composed of SrcHost (source host), DstHost(destination host), Vul(attack exploit vulnerabilities) and Perms (user rights when attack is launched). In which, Perms is composed of anonymous, guest and root/admin. AttackRes(results of attack) is composed of root access, compromised and crash. Condition is represented by boolean expression, and it's used to indicate the needed condition of attack. Boolean denotes logical constant.

Definition 4: Attack behavior is a tuple:

$Attack=(\sum,P,i,o,T,F,C,G,E,\lambda,I)$, in which, the meaning of $\sum,P,T,F,C,G,E,\lambda,I$ is the same as definition 1, $i\in P$ is input place and its pre-set is empty; $o\in P$ is output place and its post set is empty. When $Attack$ denotes atomic attack behavior, the place set is $p=\{i,o\}$. i denotes the equipments that attacker located in as well as the status of attacker when attack is launched, o denotes the equipments that attacker located in as well as the status of attacker after completing the attack. $T=T_t\cup T_i$, in which, T_t is timed transitions set, it denotes transitions set of attack behavior. This paper assumes that attacker behaviors obey exponential distribution.

For convenience of description, attack behavior is divided into atomic attack and combined attack. Figure 2 shows an atomic attack behavior model. Transition t denotes attack behavior.

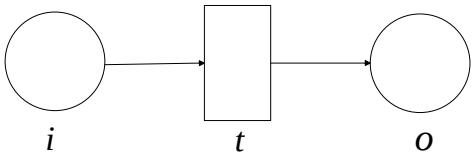

Figure 2. Atomic attack behavior model

Definition 5: Average Time of Attack (ATA). It represents the expected cost that the attacker successfully exploits the system vulnerability to achieve its target. The larger the expected value, the higher the cost of attacker to complete the target. We can evaluate the cost of successful attack by calculating the average time $\frac{1}{\lambda}$. Here, the delay time of instant transitions can be negligible.

Definition 6: If the attacker can continue to implement attack behavior B with new attack resources after implementing attack behavior A, then we consider that there is the relationship between A and B. The attack behavior A and B may be aimed at the same host, or may be different hosts.

Definition 7: The best attack path is the path with the least average time of attack, when starting from initial state to the attack target in different paths.

B. Combination Operations of Attacker Behaviors

The combination operations of attacker behaviors are that multiple attacks are combined into a composite attack according to the relationship between attack behaviors, with sequential operation, concurrent operation, and selection operation. The combination of attack behaviors can be defined in formalization as follows:

$$A ::= (A \cdot A) \mid (A \| A) \mid (A \oplus A)$$

Where A denotes attacker behavior, $\cdot$ denotes sequential operation, $\|$ denotes concurrent operation, $\oplus$ denotes selection operation.

Suppose that there are two attacks,

$A_1 = (\sum_1, P_1, p_1^i, p_1^o, T_1, F_1, C_1, G_1, E_1, \lambda_1, I_1)$ and

$A_2 = (\sum_2, P_2, p_2^i, p_2^o, o_2, T_2, F_2, C_2, G_2, E_2, \lambda_2, I_2)$

.

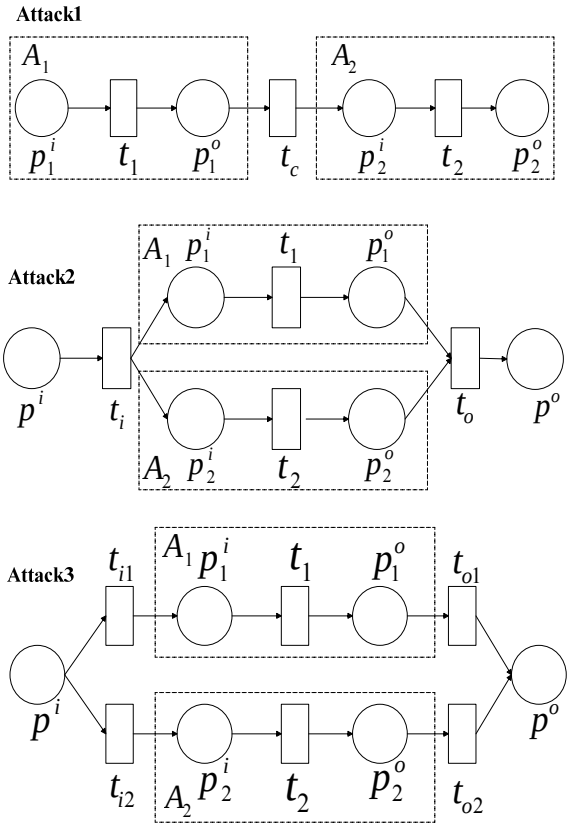

Figure 3. Combination operations of attack behavior

- **Sequential Operation:**

The $Attack1$ in figure 3 is composed of attacker behaviors A_1 and A_2 by sequential operation. The role of instant transition t_c is to connect the two attacker behaviors. The average implementation rates of timed transitions t_1 and t_2 are respectively as λ_1 and λ_2.

The average time of attack is $ATA = \frac{1}{\lambda_1} + \frac{1}{\lambda_2}$ [16].

If the attack is composed of attacker behaviors $A_1, A_2, ..., A_n$ by sequential operation, and the average implementation rates of timed transitions $t_1, t_2, ...t_n$ are respectively as $\lambda_1, \lambda_2, ...\lambda_n$, then the average time of attack is $ATA = \sum_{i=1}^{n} \frac{1}{\lambda_i}$ [16].

- **Concurrent Operation**

The $Attack2$ in figure 3 is composed of attacker behaviors A_1 and A_2 by concurrent operation. The places p^i and p^o are input place and output place of

combined attack. The role of instant transition t_i is to generate the initial conditions of attacker behaviors A_1 and A_2 according to input data, and the role of instant transition t_o is to generate total output result. The average implementation rates of timed transitions t_1 and t_2 are respectively as λ_1 and λ_2.

The average time of attack is $ATA=\frac{1}{\lambda_1}+\frac{1}{\lambda_2}-\frac{1}{\lambda_1+\lambda_2}$ [16]. If the attack is composed of attacker behaviors $A_1, A_2, ..., A_n$ by concurrent operation, and the average implementation rates of timed transitions $t_1, t_2, ...t_n$ are respectively as $\lambda_1, \lambda_2, ...\lambda_n$, then the average time of attack is $ATA=\sum_{i=1}^{n}\frac{1}{\lambda_i}-\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\frac{1}{\lambda_i+\lambda_j}+\sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}\sum_{k=j+1}^{n}\frac{1}{\lambda_i+\lambda_j+\lambda_k}+\cdots+(-1)^{n-1}\frac{1}{\sum_{i}^{n}\lambda_i}$ [16].

- **Selection Operation**

The $Attack3$ in figure 3 is composed of attacker behaviors A_1 and A_2 by selection operation. The places p^i and p^o are input place and output place of combined attack. The role of instant transitions $t_{i1}, t_{i2}, t_{o1}, t_{o2}$ is to transmit the token from input place to output place, and the initiation probabilities of t_{i1}, t_{i2} are respectively as $\alpha, 1-\alpha$. The average implementation rates of timed transitions t_1 and t_2 are respectively as λ_1 and λ_2.

The average time of attack is $ATA=\frac{\alpha}{\lambda_1}+\frac{1-\alpha}{\lambda_2}$ [16]. If the attack is composed of attacker behaviors $A_1, A_2, ..., A_n$ by selection operation, and the average implementation rates of timed transitions $t_1, t_2, ...t_n$ are respectively as $\lambda_1, \lambda_2, ...\lambda_n$, then the average time of attack is $ATA=\sum_{i=1}^{n}\frac{\alpha_i}{\lambda_i}$ [16].

C. GSCPN Model Building

The basic thought of GSCPN model building is as follows:

①Collect the vulnerability information of equipments in network, including the vulnerability information of host and service information, also collect the connective relations of equipments.

②Generate atomic attack behavior model for each atomic attack, and define strictly the conditions that transitions occur.

③Define the initial state of the network system, and starting from the initial state, to describe the process of combined attacks with sequential operation, concurrent operation, selection operation according to the relationship between the behaviors of atomic attack.

④Simplify the model for reducing the complexity. The atomic attacks or combined attacks are represented by compound transitions for reducing the complexity of model.

⑤ Verify the validity of the model, such as reachability tree [17], if not correct, to modify the graphical model.

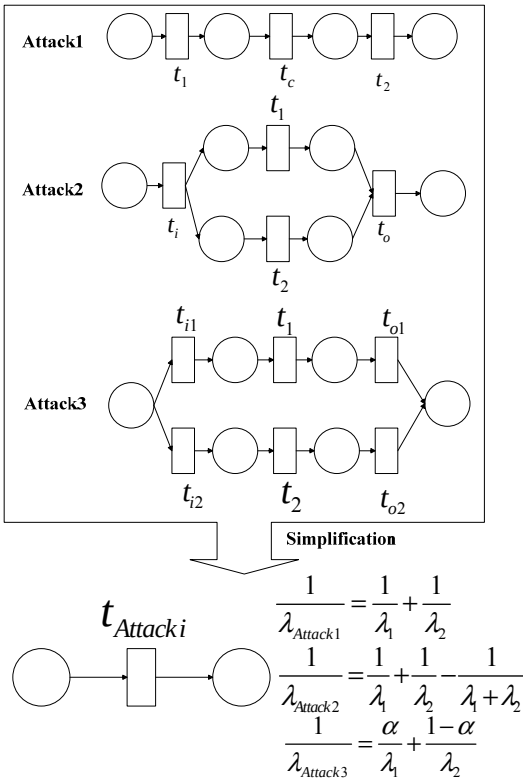

Figure 4. Simplified models

D. Model Reduction

If there are too many states in the generated model, we can simplify the model with the method in literature [16], the combined attack behaviors can be simply represented in figure 4. The average implementation rate of timed transitions $t_{Attacki}$ is as $\lambda_{Attacki}$.

III. THE BEST ATTACK PATH ALGORITHM

In this paper, we take the node that the attacker launches DDOS attacks on the target network as the final state. It means the attack can be successfully implemented at this time. There may be multiple paths during the attack process, and the main purpose of this paper is to obtain the best attack path.

Let P be the threshold value of attack cost. Maxtime is the biggest cost. S_i is the factor of network security. It

refers to the network properties involved in the process of network attacks, and they are the pre-conditions and results of attack behaviors. p^i is the place that attack is launched. p^0 is the place of attack target.

Assumption 1: The attacker is well aware of the vulnerabilities that exist in the system, and has the ability of exploiting the vulnerabilities of system and applications to intrude the system.

Assumption 2: The attackers are as intelligent agents, they would not launch attack in order to obtain the security factors that already exist in the current network. Any transition t is allowed to implement only once in attack path. It is the monotonicity assumption for attack behaviors.

Algorithm: the best attack path algorithm

Step 1 Determine p^i and p^0, and set ATA value zero. The average implementation rates $\lambda_1, \lambda_2, ...\lambda_n$ are assigned;

Step 2 Starting from the initial place, traverse all attack paths by depth first search algorithm. If $AttackCons \in S_i$ and $AttackRes \notin S_i$, then turn to Step 3;

Step 3 Calculate the value of each attack combination with the above-mentioned formulas of sequential operation, concurrent operation and selection operation.

Step 4 Accumulate the ATA value of each attack combination. If the accumulated ATA value of attack path is bigger than Maxtime, then we discard the path. Finally, the ATA values of each attack path are obtained.

Step 5 Compare the ATA value of each path, the path with the the minimum ATA value is the best attack path.

Step 6 Mark the place p^0 and use backtracking method to search the attack path.

Step 7 End.

In order to reduce the complexity of the algorithm, this paper sets the various limiting conditions and makes a judge. For example, in the second step, the monotonicity of attack path is judged; in the fourth step, it is judged whether the ATA value is larger than the threshold value. These restrictions will greatly reduce the complexity of the algorithm, and enhance the practicality of the algorithm.

The key of this algorithm is based on a depthfirst traversal of each attack path. Suppose that the generated model contains V vertices and E edges, in which, $|V|=m$, $|E|=n$. Therefore, the time complexity needed to traverse all the places and transitions is $O(m+n)$. So it would meet the needs of network security assessment.

IV. EXPERIMENTS

A. Experimental Environment and System Modeling

This paper constructs an experimental network, as shown in Figure 5, IP1 host provides telnet service, IP2 host provides FTP service, IP3 host provides database service, and IP4 host provides HTTP service. In this experiment, the goal of attacker is to control three hosts to launch a denial of service attack on the host IP4.

The vulnerability information of experimental network is shown in Table 1.

TABLE 1.
HOST VULNERABILITY INFORMATION

host	Vulnerability	Service	Result
IP1	Linux7.0 telnet	telnet	Root
IP2	ServU5.0	ftp	Root
IP3	Sql no password	Mysql	Root
IP4	SYN Flood	http	Crash

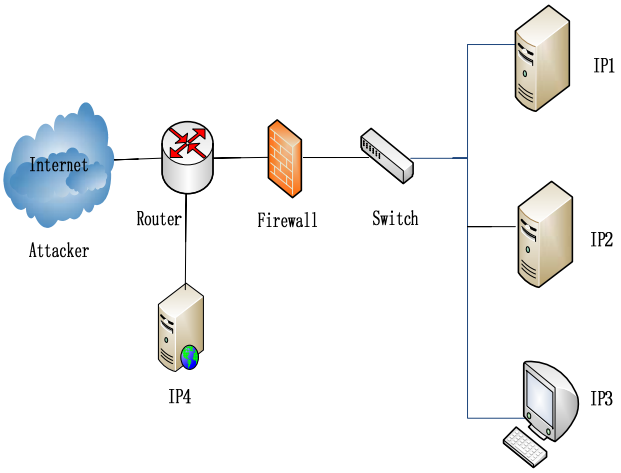

Figure 5. Network topology

According to the model building algorithm, the GSCPN model generated is shown in Figure 6. The validity and effectiveness of the model can be proved by the method of reachability tree. The atomic attack behavior A_m is composed of input place p_m^i, output place p_m^o, timed transition t_m, and $m \in [1,12]$. In the figure, we can find that there are three attack paths: $A^1 = A_1 \cdot (A_4 \| A_5) \cdot A_{10}$, $A^2 = A_2 \cdot (A_6 \| A_7) \cdot A_{11}$, $A^3 = A_3 \cdot (A_8 \| A_9) \cdot A_{12}$.

As shown in Figure 6, p^i denotes that the attack is launched; p^o denotes that the attack target is achieved; $p_1^i, p_1^o, p_2^i, p_2^o, p_3^i, p_3^o$ denote respectively the state of attacker before and after attacking IP1, IP2, IP3; p_4^i, p_4^o denote the state of attacker before and after attacking IP2, when the attacker is in IP1; p_5^i, p_5^o denote the state of attacker before and after attacking IP3, when the attacker is in IP1; p_6^i, p_6^o denote the state of attacker before and after attacking IP1, when the attacker is in IP2; p_7^i, p_7^o denote the state of attacker before and after attacking IP3, when the attacker is in IP2; p_8^i, p_8^o denote the state of attacker before and after attacking IP1, when the attacker is in IP3; p_9^i, p_9^o denote the state of attacker

before and after attacking IP2, when the attacker is in IP3; $p_{10}^i, p_{10}^o, p_{11}^i, p_{11}^o, p_{12}^i, p_{12}^o$ denote respectively the state of attacker before and after implementing DDoS attack.

The role of instant transitions t_a,t_b,t_c,t_d,t_e,t_f,t_g,t_h,t_i,t_j,t_k,t_l is to connect the two attack behaviors, instant transition t_m denotes returning to the initial state, and t_a, t_b, t_c have the same transition probability. The average implementation rates of timed transitions $t_1, t_2, ...t_{12}$ are respectively as $\lambda_1, \lambda_2, ...\lambda_{12}$, in which, t_1, t_6, t_8 denote overflow attacking by Linux7.0-telnet and installing Trojan software, t_2, t_4, t_9 denote overflow attack by ServU5.0 and installing Trojan software, t_3, t_5, t_7 denote attack by Sql-no-password and installing Trojan software, t_{10}, t_{11}, t_{12} denote DDoS attack.

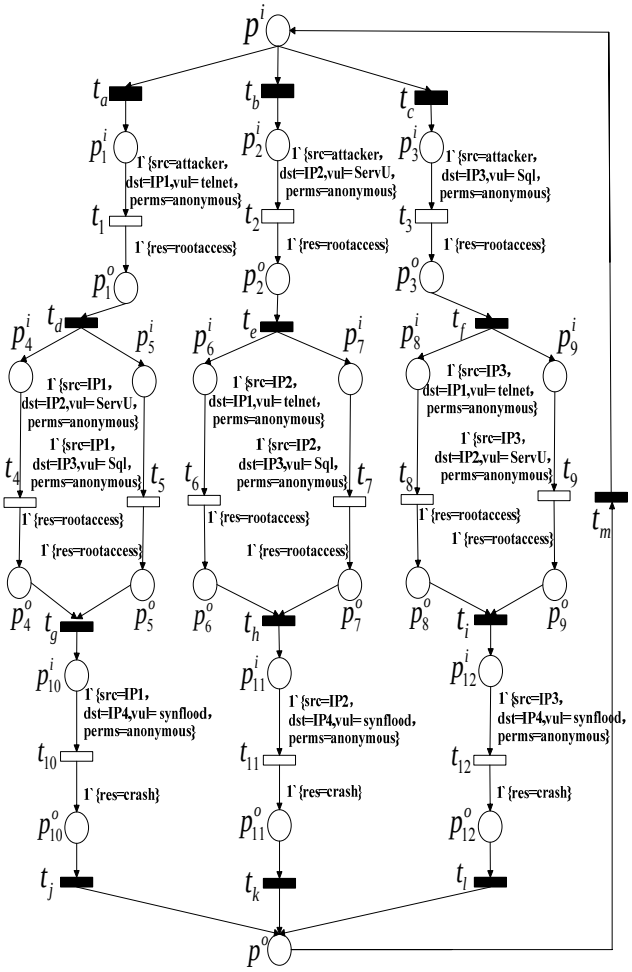

Figure 6. The GSCPN model of experimental network

B. *Experimental Results and Analysis*

By using the domain expert knowledge, the average time of attacks is as follows:

$$\frac{1}{\lambda_1}=\frac{1}{\lambda_6}=\frac{1}{\lambda_8}=5\text{(Unit Time)},$$

$$\frac{1}{\lambda_2}=\frac{1}{\lambda_4}=\frac{1}{\lambda_9}=7\text{(Unit Time)},$$

$$\frac{1}{\lambda_3}=\frac{1}{\lambda_5}=\frac{1}{\lambda_7}=3\text{(Unit Time)},$$

$$\frac{1}{\lambda_{10}}=\frac{1}{\lambda_{11}}=\frac{1}{\lambda_{12}}=10\text{(Unit Time)},$$

$$Maxtime = 25 \text{ (Unit Time)}.$$

According to the best attack path algorithm in this paper, we can get

Attack path A^1:

$$ATA=\frac{1}{\lambda_1}+(\frac{1}{\lambda_4}+\frac{1}{\lambda_5}-\frac{1}{\lambda_4+\lambda_5})+\frac{1}{\lambda_{10}}=22.9$$

Attack path A^2:

$$ATA=\frac{1}{\lambda_2}+(\frac{1}{\lambda_6}+\frac{1}{\lambda_7}-\frac{1}{\lambda_6+\lambda_7})+\frac{1}{\lambda_{11}}=23.1$$

Attack path A^3:

$$ATA=\frac{1}{\lambda_3}+(\frac{1}{\lambda_8}+\frac{1}{\lambda_9}-\frac{1}{\lambda_8+\lambda_9})+\frac{1}{\lambda_{12}}=22.1$$

By contrast, we can find that the attack path A^3 costs the least average time of attack, so it is the best attack path. We should give priority to strengthen the security measures in this path. Also we can get the simplified model of GSCPN model in figure 7. In which, $\frac{1}{\lambda_{13}}=22.9$, $\frac{1}{\lambda_{14}}=23.1$, $\frac{1}{\lambda_{15}}=22.1$.

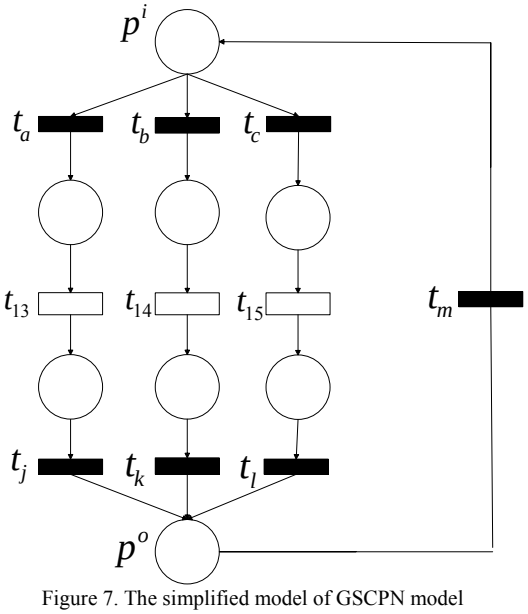

Figure 7. The simplified model of GSCPN model

We use stochastic Petri net simulation software (PIPE2.5 [18]) for solving the problem. Then the results are as follows: ATA of A^1 is 22.89, ATA of A^2 is 23.14 and ATA of A^3 is 22.07. The difference between the two is very small. Traditional performance analysis method

based on Markov chain [19] has exponential time complexity, and the analysis method in this paper has linear time complexity. So the calculation method is simple and more practical.

V. CONCLUSION

In order to get more accurate and comprehensive network vulnerability analysis results, this paper presents a calculation method of the best attack path based on GSCPN model. The validity of the method is verified through an example. The proposed calculation method is simple, and it is easy to perform manipulations automatically. So it can help managers effectively eliminate the security drawbacks and hazards hidden in the network.

ACKNOWLEDGMENT

This paper is supported by National Natural Science Foundation of China (60902102), Zhengzhou Science and Technology Innovation Team Project (10CXTD150).

REFERENCES

[1] B Schneier. Attack Trees. *Dr. Dobb' Journal*, vol. 24, pp. 21-29, 1999.

[2] Jajodia S, Noel S. *Topological Analysis of Network Attack Vulnerability*. Dordrecht, Netherlands: Kluwer Academic Publisher, 2003.

[3] Ammann P, Wijesekera D, Kaushik S. Scalable graph-based network vulnerability analysis. *Proc of the 9th ACM Conference on Computer and Communications Security*. pp. 217-224, 2002.

[4] Feng Ping-Hui, Lian Yi-Feng, Dai Ying-Xia, Li Wen, Zhang Ying-Jun. An Evaluation Model of Vulnerability Exploitation Cost for Network System. *Chinese Journal of Computers*, vol. 29, pp. 1375- 1381, 2006.

[5] Chen Feng, Liu De-hui, Zhang Yi, Su Ji-shu. A Hierarchical Evaluation Approach for Network Security Based on Threat Spread Model. *Journal of Computer Research and Development*, vol. 48, pp. 945-954, 2011.

[6] Yuanzhuo Wang, Chuang Lin, Kun Meng, Junjie Lv. Analysis of Attack Actions for E-Commerce Based on Stochastic Game Nets Model. *Journal of Computers* vol. 4, pp. 461-468, 2009.

[7] Jiang Wei, Fang Bin-xing, Tian Zhi-hong, Zhang Hong-li. Evaluating Network Security and Optimal Active Defense Based on Attack-Defense Game Model. *Chinese Journal of Computers*, vol. 32, pp. 817-827, 2009.

[8] Karin Sallhammar, Bjarne E. Helvik, Svein J. Knapskog. On Stochastic Modeling for Integrated Security and Dependability Evaluation. *Journal of Networks*, vol. 1, pp. 31-42, 2006.

[9] Wu Di, Feng Deng-Guo, Lian Yi-feng, Chen Kai. Efficiency Evaluation Model of System Security Measures in the Given Vulnerabilities Set. *Journal of Software*, vol. 23, pp. 1880-1898, 2012.

[10] Alireza Shameli Sendi, Michel Dagenais, Masoume Jabbarifar, Mario Couture. Real Time Intrusion Prediction based on Optimized Alerts with Hidden Markov Model. *Journal of Networks*, vol. 7, pp. 311-321, 2012.

[11] Wang Yuan-zhuo, Lin Chuang, Cheng Xue-Qi, Fang Bin-xing. Analysis for Network Attack-Defense Based on Stochastic Game Model. *Chinese Journal of Computers*, vol. 33, pp. 1748-1762, 2010.

[12] Wang Yong-jie, Xian Ming, Liu Jin, Wang Guo-yu. Study of network security evaluation based on attack graph model. J*ournal on Communications*, vol. 28, pp. 29-34, 2007.

[13] Chiola G, Marsan M A, Balbo G, et al. Generalized stochastic Petri nets: A definition at the net level and its implications. *IEEE Transactions on Software Engineering*, vol. 19, pp. 89-107, 1993.

[14] Jensen K. *Coloured Petri nets: Basic concepts analysis methods and practical use*. *Volume 1, basic concepts.* Berlin: Springer-Verlag, 1997.

[15] Lin Chuang. *Stochastic Petri nets and system performance evaluation*. Beijing: Tsinghua University Press, 2005.

[16] Lin Chuang, Qu Yang, Zheng Bo, Tian Li-qin. An Approach to Performance Equivalent Simplification and Analysis of Stochastic Petri Nets. *ACTA ELECTRONICA SINICA*, vol. 30, pp. 1620-1623, 2002.

[17] Zhang Pei-yun, Huang Bo, Sun Ya-min. Petri-Net-Based Description and Verification of Web Services Composition Model. *Journal of System Simulation*, vol. 19, pp. 2872-2876, 2007.

[18] Nicholas J. Dingle, William J. Knottenbelt, Tamas Suto. PIPE2: A Tool for the Performance Evaluation of Generalized Stochastic Petri Nets. *ACM SIGMETRICS Performance Evaluation Review*, vol. 36, pp. 34-39, 2009.

[19] Ferscha A. Business workflow analysis using generalized stochastic petri nets. *In Proc. 9th Austrian-Hungarian Informatics Conf*,. pp. 222–234, 1994.

Xiang Gao, born in 1984, is a doctoral student of State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou. His main research interests include network and information security, artificial intelligence.

Yue-fei Zhu is professor of State Key Laboratory of Mathematical Engineering and Advanced Computing. His main research interests include applied mathematics and information security. He has published numerous papers and gotten some of important scientific awards in this area.

Jin-long Fei, is a doctoral student of State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou. His main research interests include information security and data mining.

Tao Han, is a doctoral student of State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou. His main research interests include applied mathematics and artificial intelligence.