Network Security Risk Assessment System Based on Attack Graph and Markov Chain

Fuxiong Sun, Juntao Pi, Jin Lv and Tian Cao
School of Information and Safety Engineering, Zhongnan University of Economics and law, Wuhan, China
Email stxsfx@znufe.edu.cn, 1521030213@qq.com, 1530018223@qq.com, tian1163722295@qq.com

Abstract. Network security risk assessment technology can be found in advance of the network problems and related vulnerabilities, it has become an important means to solve the problem of network security. Based on attack graph and Markov chain, this paper provides a Network Security Risk Assessment Model (NSRAM). Based on the network infiltration tests, NSRAM generates the attack graph by the breadth traversal algorithm. Combines with the international standard CVSS, the attack probability of atomic nodes are counted, and then the attack transition probabilities of ones are calculated by Markov chain. NSRAM selects the optimal attack path after comprehensive measurement to assessment network security risk. The simulation results show that NSRAM can reflect the actual situation of network security objectively.

1. Introduction
While network applications have been fully into the social community of all aspects of life, network security issues has been becoming more and more prominent. As of December 2016, the number of PC infected Trojan horses monitored by the 360 security center was 247 million units in the report of China Internet Network Information Center (CNNIC) [1]. It is well known that network vulnerabilities lead to network security problems, so network security risk (NSR) assessment as a means of active defense came into being, which can detect potential network vulnerabilities and threats to reduce the enterprise and individual network security risk.

This paper provides a NSR mode to unearth network vulnerabilities and evaluate risk levels. The paper is organized as follows: section 2 introduces associated successes achieved in research; in section 3, the NSRAM is proposed. In Section 4, some simulation experiments are performed to verify the performance of the model. Finally, this paper concludes from this work briefly.

2. Related work
In the field of NSR assessment, many great works are accomplished by scholars. According to complex network environment, many intelligent algorithms and theories are introduced into this field. Jinsoo Shin et al. proposed a network security assessment model based on Bayesian network and event tree [2]. Fuzzy, AHP and D-S evidence theories were introduced to evaluate operations of the network services and the overall security situation of the network [3] [4] [5]. An improved immune algorithm was provided into a network security assessment method with risk theory [6]. An algorithm with polynomial complexity was analyzed based on network flow for network security risk assessment [7].

Network attack graph and its probability analysis are frequently used in network risk assessment. Yu Yajun et al. proposed an automatic analysis method of network attack graph based on graph kernel to
analyze the attack mode of intruder effectively, and take the corresponding preventive measures according to the possible attack way [8]. For real-time analysis of network risk, Wang Xiao et al presented a real-time risk prediction method based on Markov time-varying model [9]. Qi Yong et al tion probability, which improved the association between vulnerabilities in the network. [10].

CVSS is a universal vulnerability scoring system, as an industry open standard, which measures the severity of vulnerabilities and to help determine the urgency and importance of the required response [11]. Wang Zuoguang et al introduced CVSS into quantitative risk assessment of industrial control systems based on attack-tree [12].

Based on current research findings, this paper proposes a new model NSRAM based on the penetration test, attack graph and CVSS, in which the scalable Markov chain is used to qualitatively calculate the risk value. According to the magnitude of the probability, the risk level of the enterprise's network security can be determined.

3. Model Structure
The structure of NSRAM is shown in Figure 1.

NSRAM is divided into four main modules: information acquisition; penetration test; automatic generation of attack graph; attack graph evaluation.

3.1 Information Acquisition
This module is mainly the source of related information, which includes target information, network
environment, test steps and so on. The target information consists of a five-tuple pattern \((\text{Host}_\text{id}, \text{Host}_\text{vulSet}, \text{Host}_\text{con}, \text{Host}_\text{port}, \text{Host}_\text{data})\):

- \(\text{Host}_\text{id}\) represents the only host.
- \(\text{Host}_\text{vulSet}\) indicates vulnerabilities exist in host, of which value is defined according to CVE (Common Vulnerabilities and Exposures).
- \(\text{Host}_\text{con}\) represents network connection between the host and other hosts, which is used to analyze network topology.
- \(\text{Host}_\text{port}\) shows what ports are open on the host.
- \(\text{Host}_\text{data}\) means host weight which indicates its location and importance in the network.

### 3.2 Penetration Implementation

The module is an automated test platform which is developed by Ruby script to automatically call security tools to complete network scanning and penetration. Security tools include burp suite, nikto, and metasploit and so on. According to \(\text{Host}_\text{vulSet}\), the module generates the final penetration report.

### 3.3 Automatic Generation of Attack Graph

In the module, a breadth-first traversal algorithm is designed to generate the attack graph. The attribute of attacker is represented by a five-tuple pattern \(\text{Att} (\text{Att}_\text{id}, \text{Att}_\text{start}, \text{Att}_\text{target}, \text{Att}_\text{getAuth}, \text{Att}_\text{other})\).

- \(\text{Att}_\text{id}\) represents an attacker.
- \(\text{Att}_\text{start}\) is the \(\text{Host}_\text{id}\) which starts to attack.
- \(\text{Att}_\text{target}\) is the \(\text{Host}_\text{id}\) of target host.
- \(\text{Att}_\text{getAuth}\) is access rights obtained after a successful attack.
- \(\text{Att}_\text{other}\) refers to the attack path, attack time, attack ability and other information.

### 3.4 Attack Graph Evaluation

In the module, the CVSS standard is used to score vulnerabilities on an atomic node. The vulnerability score is related to attack complexity that is the risk probability of an atomic node. Based on extended Markov chain, transition probabilities between nodes can be calculated. According to \(\text{Host}_\text{con}\), all attack paths are drawn, and their attack risk are quantified. Finally, NSRAM takes the largest probability of attack path as the whole risk probability of the system.

### 4. System Modeling

#### 4.1 Attack Graph Modeling

The attack graph is defined as \(A = (A_s \cup A_d, T, R, E)\). Where \(A_s\) denotes a set of starting nodes; \(A_d\) denotes a set of target reachable nodes; \(T\) denotes an attack of an atomic node; \(R \subset A_s \times A_d\) denotes a transition relation between states; \(E\) denotes a set of directed edges. The target result in \(A\) is defined as:

\[
A(\text{Att}_\text{getAuth} == \text{root} | \text{Att}_\text{getAuth} == \text{user})
\]  

Equation (1) denotes the root or user permissions obtained on the target host after a successful attack.

The breadth-first algorithm is to build attack graph. From a host \(\text{Att}_\text{start}\) of which initial value is \(\text{Att}_\text{id}\), algorithm traverses all hosts that can be connected to it, and tries to attack these connected hosts based on \(R\). If an attack succeeds, a five-tuple pattern \(\text{Att}\) is formed. After a loop, the value of \(\text{Att}_\text{start}\) is updated by \(\text{Att}_\text{target}\) from an \(\text{Att}\). Above steps are repeated until all nodes have traversed. In order to limit the length of attack paths, the maximum length \(L\) is set, which means the length of all traversal paths must not exceed \(L\). Attack graph generation algorithm is shown in Figure 2.

In Figure 2, \(S\) is the stack of nodes, and \(\text{Pre} (H)\) and \(\text{Post} (H)\) are the pre and post node set, which can be obtained by network topology. Function \(\text{Exploit}(H_i)\) is to determine whether the node is an atomic attack node, and form the atomic ones set \(\text{Exploit}(H_i)\). Will run until the traversal length is up
to \(L\), which can prevent the reverse circuit. In addition, repeated edge judgments (in steps 12, 13) are used to avoid duplication of paths and edges.

4.2 Markov Chain Risk Assessment

Based on the extended Markov chain, the automatic generation of attack graph is used to evaluate the network security status and realize the quantitative analysis of security risk of the whole system. The Markov Chain is a stochastic process in which the state changes randomly with time. A two-tuple \(MC=(I, P)\) is defined to represent the homogeneous Markov Chain, where \(I=\{b_1, b_2, \ldots, b_m\}\) is the state space and \(P\) is the transition probability matrix.

![Figure 2 Attack Graph Generation Algorithm](image)

In the Markov process, the time \(T\) and state \(X\) are discrete and random. The Markovian property is usually expressed by the conditional distribution rate, and for any positive integer \(m, q\) and \(0 \leq t_1 < t_2 < \cdots < t_q < n\), exists:

\[
P\{X_{n+m} = b_j | X_t = b_i, X_{t_2} = b_i, \ldots, X_{t_q} = b_i, X_n = b_i\} = P\{X_{n+m} = b_j | X_n = b_i\}, b_i \in I\quad (2)
\]

In (2), \(X_{n+m}\) is the state at time \(n+m\) and \(n, t_1, m+n \in T\). The conditional probability \(P_{ij}\) of state \(b_i\) to \(b_j\) is defined as follows:

\[
P_{ij}(m, m+n) = P\{X_{n+m} = b_j | X_n = b_i\}\quad (3)
\]

\(P_{ij}\) is also called the transition probability of the moment. Based on the extensible Markov chain, after analyzing the attack graph, it is found that the simple Markov chain only indicates the state of the attack, and there is no reason for the migration of the attack state. For this reason, its concept is represented by the triplet \(EMC=(I, P, A)\), where the set of all available methods is denoted as \(A=\{G_1, G_2, G_3, \ldots, G_k\}\). The \(A\) is called the set of attack actions.

The nodes in the chain represent the state of the network, and the directed edge represents the weight of the state transition which corresponds to the transition probability. For each node, the sum of the weights is homogeneous. The transition probability matrix is defined as:
\[ P = \begin{bmatrix} p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ p_{n1} & \cdots & p_{nn} \end{bmatrix} \] (4)

In (4), \( \forall i = 1, 2, 3, \ldots, \sum_{j=1}^{\infty} p_{ij} (n, n + 1) = 1. \)

The CVSS defines the complexity of the attack. Most of the researches use it to classify the attack difficulty, as shown in Table 1.

### Table 1 The CVSS standard

| Id | Weight | Level                                      |
|----|--------|--------------------------------------------|
| E0 | 0.35   | Network risk level is high, the penetration of low degree of difficulty |
| E1 | 0.61   | Network risk level medium, easy to penetrate the degree of moderate difficulty |
| E2 | 0.71   | Network risk level is low, the penetration of high degree of difficulty |
| E3 | 0.71   | Network risk level is undefined |

But the CVSS standards have few levels of definition which applies to a simple network environment. Therefore, this paper adopts an improved CVSS to calculate the risk probability of nodes, as shown in Table 2.

### Table 2 The CVSS standard defines the ease of vulnerability penetration attacks

| Id | Weight | Level                                      |
|----|--------|--------------------------------------------|
| E0 | 0.1    | Public reporting but the attack is only theoretically possible or not publicly reported weaknesses |
| E1 | 0.2    | Public report but no attack method         |
| E2 | 0.3    | Publicly report and refer to possible attack methods |
| E3 | 0.4    | Public report and a rough description of the attack method |
| E4 | 0.5    | No ready attack tool but with more detailed attack steps |
| E5 | 0.6    | There is no ready attack tool but there are very detailed attack steps |
| E6 | 0.7    | There is a corresponding attack code and use the method |
| E7 | 0.8    | Customize the available attack tools with detailed attack steps |
| E8 | 0.9    | Ready-to-use attack tools with detailed attack steps |

After an attack action \( G_i \in A \) is selected, the state transfers from node \( Z_i \) to the other one \( Z_x \) where \( 0 < i, j < n, 0 < r < k \). If the penetration difficulty of \( G_r \) is \( E(G_r) \) in CVSS, the transition probability from \( Z_i \) to \( Z_j \) is defined as:

\[ P_{ij} \left( Z_i \xrightarrow{G_r} Z_j \right) = \frac{E(G_r)}{\sum_{r=0}^{\infty} E(G_r)} \] (5)

In the attack graph, I am divided into two sets: target node set \( I_z \) and non-target one \( I_x \), where \( I_z \cup I_x = I \). The \( V(Z_i) \in [0,1] \) is the probability from \( Z_i \in I \) to \( Z_j \in I_x \) define as follows:

\[ V^n(Z_i) = \sum_{z \in \text{STATUS}(Z_i)} \sum_{g_i \in \text{REGULATION}(Z_i \rightarrow Z_j)} P(Z_i \xrightarrow{G_i} Z_j) \times V^{n-1}(Z_j) \] (6)

In (6), \( \forall Z_i \in I_z \), then \( V^0(Z_i) = 1 \), and \( \forall Z_i \in I_x \), then \( V^0(Z_i) = 0 \). \text{STATUS}(Z_i) \) is the node set that successful attack is accomplished by only one attack action. \text{REGULATION}(Z_i \rightarrow Z_j) \) is optional rule from \( Z_i \) to \( Z_j \). After the \( n \) iteration, the iterative process is considered convergent if \( V^n = V^{n-1} \). For example, there is a directed attack graph with transition probability shown in Figure 3.

It is assumed that the target node is 4 in Figure 3, then \( I_x = \{0, 1, 2, 3, 4, 5\} \), \( I_z = \{4\} \), \( I_0 = \{0, 1, 2, 3, 5\} \), \( A = \{G_1, G_2, G_3, G_4, G_5, G_6, G_7\} \). The directed graph is divided into two layers, and to attack the target can be done at most two steps: \( Z_0 \rightarrow Z_1 \rightarrow Z_4 \) and \( Z_0 \rightarrow Z_2 \rightarrow Z_4 \). Table 3 illustrates the iterative process of attack graph.
Table 3 Iterative Process

| Iterative | \( Z_0 \) | \( Z_1 \) | \( Z_2 \) |
|-----------|---------|---------|---------|
| 0         | \( V^0(0) = 0 \) | \( V^0(1) = 0 \) | \( V^0(2) = 0 \) |
| 1         | \( V^1(0) = 0 \) | \( V^1(1) = 1\ast E(G_4) \) | \( V^1(2) = P_4\ast E(G_3) \) |
| 2         | \( V^2(0) = P_1 \ast 1 \ast E(G_3) \ast E(G_4) + P_2 \ast E(G_2) \ast P_4 \ast E(G_5) \) | \( V^2(1) = 1\ast E(G_4) \) | \( V^2(2) = P_4\ast E(G_3) \) |
| 3         | \( V^3(0) = P_1 \ast 1 \ast E(G_3) \ast E(G_4) + P_2 \ast E(G_2) \ast P_4 \ast E(G_5) \) | \( V^3(1) = 1\ast E(G_4) \) | \( V^3(2) = P_4\ast E(G_3) \) |

Figure 3 Attack Graph Example

5. Experimental Analysis

5.1 Experimental Environment

In the experimental environment, there are three hosts (\( H_3, H_4, H_5 \)), one server (\( H_1 \)), one router and one firewall (\( H_2 \)) shown in Figure 4. The operating systems of three hosts and server are windows XP. The attack host (\( A \)) is Linux. In \( A \), an automated test platform is installed which is developed by Ruby script to automatically call security tools to complete network scanning and penetration. Security tools are installed including burp suite, nikto, metasploit and so on. The three hosts are located in Intranet protected by firewall and server provides some network services to extranet.

Figure 4 Experimental Environment

In the simulation experiment, there are five typical vulnerabilities exist in network, listed in Table 4.
Table 4 Vulnerability Selection

| Host_id | Host type      | Host_vulSet | Att_getAuth |
|---------|----------------|-------------|-------------|
| H1      | server         | ServU5.0    | Root        |
| H2      | Firewall       | telnet      | Root        |
| H3      | Notebook host  | SQL         | Root or user|
| H4      | Desktop host   | RPC         | Root        |
| H5      | Desktop host   | Remote login| Root        |

5.2 Generate Attack Graphs
By the breadth traversal algorithm proposed in Figure. 2, the attack graph is generated in Figure.5 when the target node is H3.

Figure 5 Generated Attack Graph

The probability of a successful attack is determined by the probability that the vulnerability is breached, which can be defined by the penetration attack complexity in CVSS, as shown in table 5.

Table 5 Penetration Attack Complexity Table

| Attack action   | id | weight |
|-----------------|----|--------|
| ServU5.0        | E3 | 0.4    |
| telnet          | E7 | 0.8    |
| SQL             | E5 | 0.6    |
| RPC             | E4 | 0.5    |
| Remote login    | E8 | 0.9    |

Based on table 5, the corresponding attack graph with transition probability is calculated by (5). Then the transition probability of A → H1 is 0.4/ (0.8+0.4) =0.33, A → H2 is 0.8/ (0.4+0.8) =0.67, and so on. By (4), the transfer probability matrix is obtained as shown in Table 6.

Table 6 Transfer probability matrix

| End state | Start state | A   | H1  | H2  | H3  | H4  | H5  |
|-----------|-------------|-----|-----|-----|-----|-----|-----|
| A         | 0           | 0.33| 0.67| 0   | 0   | 0   |
| H1        | 0           | 0   | 0   | 0.3 | 0.25| 0.45|
| H2        | 0           | 0   | 0   | 0.3 | 0.25| 0.45|
| H3        | 0           | 0   | 0   | 0   | 0   | 0   |
| H4        | 0           | 0   | 0   | 1   | 0   | 0   |
| H5        | 0           | 0   | 0   | 1   | 0   | 0   |
Attack graph with transition probability is shown in Figure 6.

5.3 Attack Graph Evaluation
Markov chain is used to calculate the probability of attack of attack graph, and then the complexity of the attack is evaluated. The probability from each node to another node depends mainly on the transition probability $P$ shown in Figure 5. There are 6 attack paths to $H3$, for example the $P$ of the 1st path is calculated as $P=0.33\times0.4\times0.25\times0.5\times1\times0.6 = 0.0099$. Table 7 shows the $P$ values of all attack paths.

| Path NO. | Starting node | Target node | Traverse the node path | Attack probability |
|----------|---------------|-------------|------------------------|--------------------|
| ①        | $A$           | $H3$        | $A\rightarrow H1\rightarrow H4\rightarrow H3$ | 0.0099             |
| ②        | $A$           | $H3$        | $A\rightarrow H1\rightarrow H3$              | 0.02376            |
| ③        | $A$           | $H3$        | $A\rightarrow H1\rightarrow H5\rightarrow H3$ | 0.016038           |
| ④        | $A$           | $H3$        | $A\rightarrow H2\rightarrow H4\rightarrow H3$ | 0.0402             |
| ⑤        | $A$           | $H3$        | $A\rightarrow H2\rightarrow H3$              | 0.09684            |
| ⑥        | $A$           | $H3$        | $A\rightarrow H2\rightarrow H5\rightarrow H3$ | 0.130248           |

According to Figure 6, the target node $H3$ can be reached by 3 steps at most. The attack path ⑥ is more probably of which probability is 0.130248. By (6), the overall risk value is $V^3(H3) = 0.0099 + 0.02376 + 0.016038 + 0.0402 + 0.09684 + 0.130248 = 0.316986$. According to CVSS standard, the network security level is $E0$ level which means that the network danger level is high but penetration degree is low.

6. Conclusion
Network vulnerability is one of main security threats of enterprise network. Mining network vulnerability helps to improve the management of network security risk. Network vulnerabilities exist in hosts and various network services, and are closely related to the current network topology. This paper proposed the NSRAM which can automatically and periodically collect network topology and its vulnerability information. Combining the improved CVSS and probability evaluation of attack path, NSRAM determines the risk level of enterprise network security. Simulation experiments verify its
feasibility and flexibility.

In large and complex network, the changes of network topology at any time lead to frequent migration of network security risk. Incremental evaluation of attack graph and path will greatly improve the efficiency of the model, which is the focus of the future research.

7. Acknowledgment
This research was financially supported by the education research of Zhongnan University of Economics and law (2016J1408) and Youth project of the National Social Science Foundation (16CXW019).

8. References
[1] China Internet Information Center (ENNIC). The 39th China Internet Development Statistics Report[EB/OL], 2017.
[2] Jinsoo Shin, Hanseong Son, Gyunyoung Heo, Cyber Security Risk Evaluation of a Nuclear Instrumentation and Control System Using Bayesian Networks and Event Trees[J]. Nuclear Engineering and Technology, 2017(49):517-524.
[3] Sun F X. Danger Theory Based Model for Network Security Risk Assessment[J]. Advanced Materials Research, 2011(187):148-154.
[4] Li Qun Tang, Jian Ping Ma, Hui Qiang Wang. Research on a Method of Service-Oriented Network Layer Network Security Situation Assessment[J]. Applied Mechanics and Materials, 2013: 260-264.
[5] Fang Yang, Study on the AHP&Dset Telecom Network Risk Assessment Model and The Application[D]. Beijing University of Posts and Telecommunications, 2010.
[6] Yuan H. A Network Security Risk Assessment Method Based on Immunity Algorithm[J]. Advanced Materials Research, 2010:948-953.
[7] Wu Jinyu. Research on Key Technologies of Network Security Risk Assessment [D]. Beijing University of Posts and Telecommunications, 2013.
[8] Yu Yajun, Liu Zeshen, Pan Zhisong, Hu Guyu, Chang Qing. The Network Attack Graph Analysis Based on Graph Kernel [J]. Military Communications Technology, 2016 (03): 20-25.
[9] WANG Jie, LI Qian-guo, QI Yong. A Real-time Analysis Method of Network Security Risk Based on Markov Model [J]. Computer Science, 2016: 338-341.
[10] Qi Yong, LIU Min, LI Qian-guo. An attack graph model based on extended Markov chain [J]. Computer Engineering and Design, 2014, (12): 4131-4135.
[11] K Scarfone, P Mel. An Analysis of CVSS Version 2 Vulnerability Scoring[C], International Symposium on Empirical Software Engineering and Measurement. 2009:516-525.
[12] Wang Zuoguang, WEI Qiang, LIU Wen-wen. Quantitative risk assessment of industrial control systems based on attack-tree and CVSS [J]. Application Research of Computers, 2016 (12):3785-3790.