Journal of Information Hiding and Multimedia Signal Processing ©2017 ISSN 2073-4212
Ubiquitous International Volume **8**, Number **3**, May **2017**

A Generation Algorithm INAG for Attack Graph Based on NST Model

Hui Wang, Zhe Wang and Fuwang Chen

School of Computer Science and Technology
Henan Polytechnic University
Jiaozuo Henan 454000, China
wanghui_jsj@hpu.edu.cn

Received October, 2016; revised February, 2017

Abstract. *In order to reduce the size of attack graph, attack cost is used as a main indicator to generate network attack graph in traditional methods. However, it is assumed that all key elements, such as cost, benefit and so on, are independent of each other, and ignore the relevant quantitative analysis among key elements. To this end, this paper proposes an improved generation algorithm for the attack graph of network. First, this paper proposes a novel NST model; secondly, we introduce two quantitative indicators; then, a concept of relative intensity coefficient ($\rho\partial$) is given; finally, we propose an improved generation algorithm based on the cost for attack graph of network (INAG algorithm). Simulation experiment results show that this algorithm not only reduces the size of attack graph, but also makes attack graph generated more compact.*

Keywords: Attack graph, Cost, Attack complexity, Risk value, Intensity coefficient

1. **Introduction.** According to "China Internet Network Information Centre Report" [1], from 2011 to 2015, the proportion of computer used in enterprises has been maintained at a high level. According to "Tencent Internet Security Report" [2], in 2015, 145 million new viruses of computer were found by the computer laboratory of Tencent, which increased by 5% compared with 2014. According to "2015 China Internet Network Security Report" [3], in 2015, the security incidents of network had occurred more than 120 thousand, and there was an increase of 125.9% compared with 2014. In addition, according to "Internal Threat Report of Vormetric in 2015" [4], 89% of the respondents thought that their institutions were now facing greater danger from internal attacks, and 34% of people thought that the internal network security was very fragile. At present, the internal network security begins to be attention by enterprises, while industry network security managers also attach great importance to it.

2. **Related Works.** In modelling terms, Cunningham [5] first proposed attack graph model. He believed that network was composed of a variety of components, and these components were connected by physical or logical ways. Ritchey [6] proposed an automatically generate model of attack graph. Although this method could generate attack graph automatically, it was not suitable for large-scale network. Because the model contained all states that were very easy to cause state explosion problem. Then, based on the assumption of expert experience, Ammann [7] proposed a monotonic assumption of network attack to solve the state explosion problem. But the assumption only limited and simplified the generation process of an attack graph. While a generation method of

network attack graph based on multiple preconditions was proposed by Ingols Kyle [8]. But there was a problem with this approach, which had a great increase in the scale of attack graph when the network scale increased. Chen Xiaojun [9] proposed an attack graph model based on the judgment of probability about internal attack intention. But the model relied too much on the expert knowledge base, so there were some limitations when the model was applied in large scale network. In terms of cost, based on the cost and benefit of an attacker, Dapeng Man [10] gave a generation algorithm of global attack graph based on the breadth first search to reduce the size of the attack graph by setting threshold. Although this method could reduce the size of attack graph, it had some limitations due to the breadth first. So an attack graph generated by this method was not suitable for large scale network. Tan Jun [11] proposed a selective attack strategy based on the cost of an attacker, it eliminated cycle path through attacker's subjective choice, but ignored the role of an attacker as the main body of intelligent network attack. And it only took into the cost of an attack, did not consider the benefits.

However, when senior experts are in the process of modelling, existence of redundant paths in the attack graph is inevitable. Therefore, compared with existing ones, our contributions in this paper are summarized as follows.

(1) First, a kind of model based on state transition for network attack is defined (NST model). In order to optimize the cost of the model, two quantitative indicators are introduced, they are the value of attack complexity (Acx) and the risk of display (Dsk). Then, the relationship between Acx and Dsk are quantitatively analysed.

(2) Then, we also give a calculation formula based on the related coefficient of strength ($\rho\partial$). And an improved generation algorithm (INAG algorithm) is proposed based on the calculation formula.

(3) Finally, the experimental results show that the algorithm which based on the improved value of cost can effectively simplify the original attack graph.

3. **Preliminaries Study.**

3.1. **Definitions and Overview.**

In order to describe the model, some basic definitions are given in this section. Definition 1 and Definition 5 are from previous research [12].

Definition 3.1. *Vulnerability: Let V be the set of vulnerabilities and $V = \{v_1, v_2, v_3, \ldots, v_i\}$. For each $v_i \in V$, we define v_i as a vulnerability in the set V. For this paper, vulnerabilities refer to the technical defects of the hardware and software, and also refer to the tactical deficiencies in the management of computer system.*

Definition 3.2. *Vulnerability Attack is a 3-tuple $V_A = (A_P, t, A_S)$. We use the 3-tuple to represent vulnerability attack set, where:*

A_P is a set of input preconditions for the vulnerability attack. t represents a state transition, which represents a change in the state of the network after an attack; A_S represents the consequences of vulnerability attacks, and the consequences include obtaining new permissions, adding new connections and obtaining new vulnerability information.

A concrete example of a vulnerability attack to gain new permissions is shown in Figure 1. The attacker starts from controlling host H1 to getting the root permission of host H2. During the process of attacking, he exploits the vulnerability (2016-1194 CVE) in Web IIS service which exists on host H2. The vulnerability attack should meet four preconditions: (1) The attacker's permission at the source host H1 is at least User level (i.e. U_H1); (2) The host H1 has a permission to access Http service on the host H2 (i.e. Http_H1_H2); (3) Host H2 is running Web IIS services (i.e. Web_H2); (4) There is a vulnerability in Web IIS service (i.e. V1194), whose number is 2016-1194 CVE. Only if the above four conditions all are met, the attacker will successfully launch an attack on H2 (i.e. V1194_H1_H2) from H1. The result is that the attacker can get root permission on the H2 (i.e. R_H2).

Definition 3.3. *Attack Complexity is a statistical value of probability when one vulnerability is successfully exploited (i.e., Acx). It serves as a quantitative indicator of a successful attack in this paper. In order to determine the attack complexity of v_i, we also define that $\forall v_i \in V, \exists P_A(i) = Acx(v_i)$, $P_A(i)$ is the attack complexity of v_i.*

Definition 3.4. *Risk Value of Display is a statistical value of probability when a vulnerability attack is found (i.e., Dsk). It is used as a quantitative indicator to reflect the failure for an attack. In order to determine the risk value of display of v_i, we also define that $\forall v_i \in V, \exists P_D(i) = Dsk(v_i)$, $P_D(i)$ is the risk value of display of v_i.*

Definition 3.5. *Cost is a 2-tuple Cost = (Acx, Dsk). For this paper, it is all efforts to complete an attack from the attacker to the target when the attack is completed (i.e. Cost).*

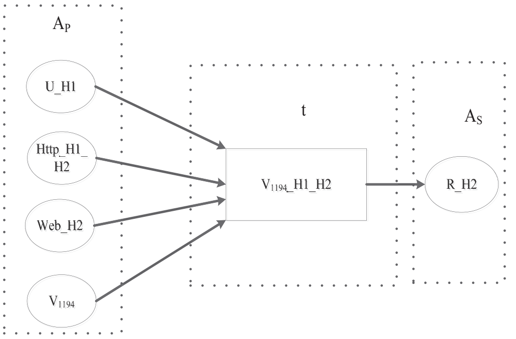


Figure 1. The instance of vulnerability attack.

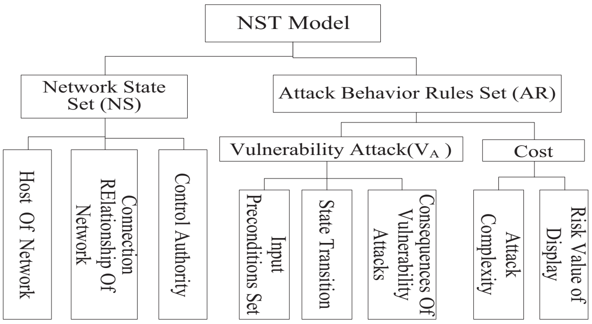


Figure 2. The composition of the model.

3.2. NST Model Definition.

Definition 3.6. *NST Model is an attack model based on state transition for internal network. For this paper, the model can be represented by a 2-tuple. The 2-tuple is NST = (NS, AR). Among them, NS is composed of network state set of the model, and AR is a set of attack behaviour rules.*

The composition of this model is shown in figure 2. The previous text has made an overview and definition for vulnerability attack (i.e. V_A) and the cost (i.e. Cost). Below, this paper defines network state and attack behavior rules in the model.

Definition 3.7. *Network State is a 3-tuple NS = (Hd, Nl, Crol) and describes necessary related elements in attack graph based on the model in this paper, where:*

Hd represents the host in a network, and it also is a 4-tuple, Hd = (Hostid, Os, Sers, Vu). In the 4-tuple Hd: 1) Hostid is the unique identifier of a host in network; 2) Os is a version of the operating system running on a host; 3) Sers represents an open list of available services on a host; 4) Vu represents a list of vulnerabilities in a host.

Nl is the representation of connection relationship in a network, and it also is a 3-tuple, NL = (Src, Dst, Conn_pro). In the 3-tuple Nl: 1) Src represents the source host; 2) Dst represents the destination host; 3) Conn_pro represents a protocol or port for connecting to another host, if there is no connection between two hosts, Conn_pro = null, and when the two hosts are the same host, Conn_pro = localhost.

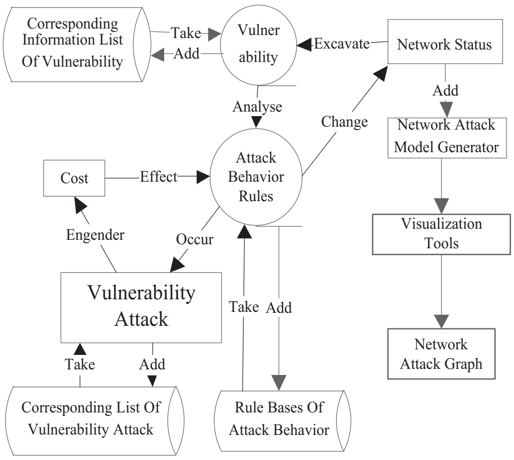


Figure 3. The Architecture Diagram of Attack Graph Generation.

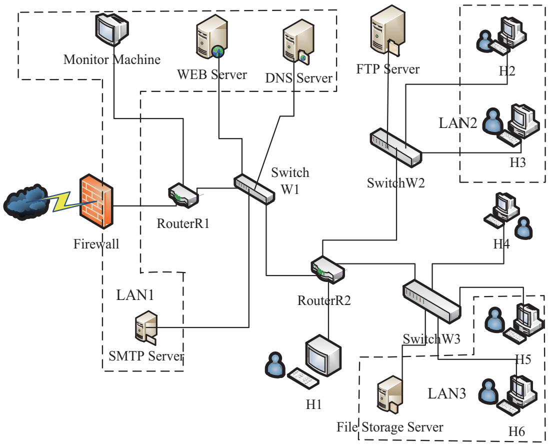


Figure 4. The Map of Network Topology.

Crol represents the control authority that an attacker has in a host. In this model, the authority is divided into three levels: system administrator privileges, general system user rights and visitor rights for remote network. Let us restate the description above as root, user and access.

Definition 3.8. *Attack Behavior Rules are expressed as follows: AR = (V_A, Cost), where: $V_A = (A_P, t, A_S)$, Cost = (Acx, Dsk). It describes the conditions and results of an attack in this paper.*

We use a 4-tuple to represent the A_P, A_P = (Src_access, Conn_pro, Dst_sers, Dst_v), where:

1) Src_access indicates that an attacker in the source host should have the lowest access rights; 2) Conn_pro indicates that the connection between a source host and a destination host should be satisfied, which guarantees the reachability of an attack; 3) Dst_sers represents open services in a host, which ensure that vulnerabilities are available; 4) Dst_v indicates corresponding vulnerabilities in the destination host.

We also use a 3-tuple to represent the A_S, A_S = (Rslt_access, Rslt_conn, Rslt_vs), where:

1) Rslt_access indicates an attacker's access to the target host after a successful attack; 2) Rslt_conn represents a change of connection relationship in a network after a successful attack; 3) Rslt_vs indicates vulnerabilities of a successful attack.

The t of V_A represents a state transition which has been explained above. In this model, an attacker's attack behavior is regarded as a use of v_i, which reflects the attack path through a vulnerability attack.

3.3. Attack Graph Generation Framework Based on NST Model.

According to these definitions of the NST model, its specific architecture is shown in figure 3. The architecture of network attack graph is summarized as follow in this paper:

1). Administrator collects network status information of the target network by using a variety of scanning, sniffer and data mining; 2). Administrator creates a corresponding information list of vulnerability through the analysis of the target network state information to identify the existence of vulnerability information; 3). Administrator analyzes vulnerabilities to identify the vulnerability that might be implemented by an attacker to attack, and establishes corresponding rule bases of attack behaviors; 4). Administrator sets up a corresponding vulnerability attack list for those possible vulnerabilities, and analyzes the impact of the cost on attack behaviors; 5). Network status changed by vulnerability attack is added to a network attack model generator.

The steps of above is form the model of network attacks. When these steps are completed, a network attack graph is generated.

4. INAG Algorithm Based on NST Model.

4.1. Cost Attribute Value Quantization in NST Model.

Based on the NST model, this paper proposes the improved network attack graph generation algorithm (INAG algorithm). According to related definitions of the NST model, the INAG algorithm mainly uses cost value as a main indicator to generate network attack graph. However, Acx is affected by many factors, such as automation degree of attack tools, an attack's time, attackers' experience and so on. So it is difficult to accurately describe every vulnerability's Acx, and the different Acx of vulnerabilities can only be expressed by some variables. According to survey statistics, researchers found that there was a mapping between a vulnerability used of the cycle and Acx, through the analysis of a large number of security incidents. This mapping can express Acx difference from every vulnerability. The quantization standard of Acx is shown in table 1.

Table 1. The Value of Acx Quantization Table

Grades	Acx	Description
1	0.1	There is a detailed attack method, and no need to own an attacking tool.
2	0.3	There is a detailed method of attack and a readily available attacking tool.
3	0.5	There is a detailed method of attack, but without attacking tools.
4	0.7	There is a public reporting vulnerability, and only a rough attack method.
5	0.9	There is a public reporting vulnerability, but no attack method.

At the same time, Acx is a quantitative indicator for a successful attack, Dsk is used as a quantitative indicator to reflect the failure of an attack. So Acx also affects Dsk during an attack. According to the National Infrastructure Advisory Council (NIAC), the aim is to provide a set of open common vulnerability scoring mechanism CVSS [13] (Common Vulnerability Scoring System). The CVSS consists of three metrics for computing vulnerability scores, which are basic evaluation, life cycle assessment, and environmental assessment. But this paper does not consider the life cycle and the environmental assessment, because they are influenced by larger uncontrollable factors. The basic evaluation consists of six indexes, they are the vector of access (Av), the identity authentication (Au), the attack complexity (Acx), the impact of confidentiality (CI), integrity (II) and availability (AI). Each metric value is shown in table 2.

Table 2. The Value of Basic Group CVSS Table.

Indicators	Grade one	Grade two	Grade three
Av	Remote (0.395)	Near (0.646)	Local (1.0)
Au	Multiple (0.45)	Single (0.56)	Unwanted (0.704)
Acx	High (0.35)	Medium (0.61)	Low (0.71)
CI/II/AI	Nothing (0)	Slightly (0.275)	Serious (0.64)

As can be seen from the table 2, when an attack position at the beginning of vulnerability attack is lower, the network is more requirements to the identity authentication for attacker, and the attack complexity of a vulnerability is higher. Meanwhile, the lower the measure of corresponding indicator is, and the smaller the impact of CI/II/AI is. Therefore the probability that an attacker was found is smaller. So, this paper uses the value of CI/II/AI to evaluate the value of Dsk. In addition, we introduce the concept of correlation strength coefficient ($\rho\partial$) to recalculate the cost, and its main function is to quantitatively analyze the relationship between Acx and Dsk. This concept ($\rho\partial$) is based on the Pearson Product Moment Correlation Coefficient for the quantitative analysis of the relationship between two variables A and B.

4.2. **Calculation of Correlation Strength between Acx and Dsk.** Pearson Product Moment Correlation Coefficient is a well-known formula, which is usually used to measure the relationship between two fixed distance variables (e.g. weight and height). This paper uses the formula to calculate the correlation between Acx and Dsk, which are two important indicators for cost. Pearson Product Moment Correlation Coefficient formula is as follows:

$$\rho_{X,Y} = \frac{\sum XY - \frac{\sum X \sum Y}{N}}{\sqrt{(\sum X^2 - \frac{(\sum X)^2}{N})(\sum Y^2 - \frac{(\sum Y)^2}{N})}}$$

In the above formula, N is the number of experiments which determine the value of correlation strength, Y and X are two variables, $\rho_{X,Y}$ is the calculation of correlation value between two variables. According to Pearson's formula.

$$\rho\partial = \frac{\sum XY - \frac{\sum X \sum Y}{N}}{\sqrt{(\sum X^2 - \frac{(\sum X)^2}{N})(\sum Y^2 - \frac{(\sum Y)^2}{N})}}$$

$$\rho\partial(i) = \frac{\sum P_A(i)P_D(i) - \frac{\sum P_A(i) \sum P_D(i)}{N}}{\sqrt{(\sum (P_A(i))^2 - \frac{(\sum P_A(i))^2}{N})(\sum (P_D(i))^2 - \frac{(\sum P_D(i))^2}{N})}}$$

In the formula, X is the value of Acx, Y is the value of Dsk, and N is the number of probability and statistics which is to obtain the number of times when the attack complexity and the value of risk of same vulnerability are found. $\rho\partial(i)$ is the correlation strength coefficient of $P_A(i)$ and $P_D(i)$, and $\rho\partial \in [-1, 1]$.

Reflected by network attacker's experience, the stronger correlation degree between Acx and Dsk is, the higher cost of an attack is. On the contrary, the weaker correlation degree is, the lower cost of an attack is. Sample data are given in this paper as shown in table 3, according to the definition of Pearson's correlation coefficient, the greater absolute value of $\rho\partial$ is, the closer value of the coefficient is to 1 or -1 and the stronger correlation degree is. Conversely, $\rho\partial$ is closer to 0, the degree of correlation is weaker.

Table 3. Absolute Value $\rho\partial$ and Description of Correlation Strength.

$\rho\partial$	Description of Correlation Strength
0.8–1.0	Extremely Strong Correlation
0.6–0.8	Strong Correlation
0.4–0.6	Moderate Correlation
0.2–0.4	Weak Correlation
0.0–0.2	Extremely Weak Correlation or No Correlation

However, network attack is usually a complex multi-step process. Price of Vulnerability Attacking (pe$_i$) which affects the calculation of cost is defined as follows.

4.3. **Quantitative Calculation of Cost for INAG Algorithm.**

Definition 4.1. *Price of Vulnerability Attacking* (pe$_i$) *is the cost of that an attacker need to pay when one vulnerability has been successfully used.*

Assume that attackers are wise, if there are two attacks or multiple attacks against the same vulnerability in an attack path, the cost of subsequent repeated attacks will certainly be less than the first. Therefore, in a process of an attack, calculating the price of repeated attacks about the same vulnerability is also related to attacker's experience. So, this paper gives a calculation formula of cost which is based on $\rho\partial$. The formula is as follows:

$$Cost = \begin{cases} \sum_{j=1}^{n} \mathrm{pe}_{ji} * (\xi\rho\partial(j)) & (\text{No Vulnerability Repeat}, 1 \leq j \leq n) \\ \sum_{j=1}^{n} (K^{time-1} * \mathrm{pe}_{ji}) * (\xi\rho\partial(j)) & (\text{Vulnerability Repeat}, 1 \leq j \leq n) \end{cases}$$

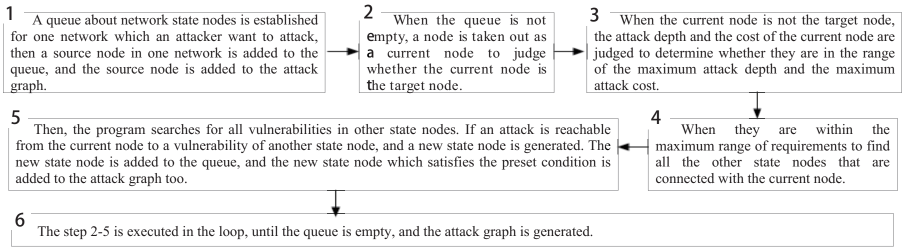
Among them, n represents the number of nodes in entire attack path from the source node to the node of j which is attacked. pe_{ji} represents the cost of a successful attack for the attacked node j which is using a vulnerability with maximum permission on the attacked node j. $0 < i < u$, and u represents the total number of vulnerabilities on the attacked node j. $\rho\partial$ represents the correlation coefficient of the attack complexity and the risk value of display. K $(0 < K < 1)$ represents the dependence on an attacker's experience during an attack, and the value is determined by expert experience. And time represents the number of repeated attacks during an attack from the source node to the attacked node j. ξ represents the compactness coefficient of network, which reflects the robustness of a network, and its value is also determined by expert experience.

However, an internal network is smaller relative to an external network, and attackers are considered to be wise. So when the depth of attack is too long or the cost beyond the default threshold, then the attack is considered to be non-reachable.

Algorithm 1 Attack graph generation algorithm INAG.

```
Algorithm: INAG
Enter the initial network state: init_state
Maximum attack depth: Max_Depth
Maximum attack cost: Max_Cost
Output attack graph: A_Graph
Begin:
Queue State_queue = new Queue;
State_queue <− EnQueue(init_state);
While(State_queue.IsEmpty())
{N_cunt <− dequeue(State_queue);
 If (N_cunt =! N_goal && N_cunt.depth < MaxDepth && N_cunt.cost < MaxCost)
   {for each vul in N_cunt.vuls
     {if(AR.preconditions = true)
       {Nas.depth = N_cunt.depth + 1;
        Nas.cost = N_cunt.cost + cost;
        Graph.Addedge(N_cunt, Nas);
        State_queue <− EnQueue(Nas);
}}}}
```

4.4. **Algorithm INAG Algorithm Based on Cost Value.** Attack graph generation based on the INAG algorithm uses queue structure to save new nodes. First, initial state of network is added to the queue, and then the algorithm is executed. A state node is taken out of the queue by the algorithm. The depth and the cost of the node are calculated, if any one of them does not exceed the default threshold. To determine whether attack conditions are established for each vulnerability of the state node, if one of them is formed, a new state node is generated and added to the queue. And then a new node is taken from the queue until the queue is empty. Finally, the cycle is over when the queue is empty. The thesis studies the specific approach of the algorithm as follows.



1 A queue about network state nodes is established for one network which an attacker want to attack, then a source node in one network is added to the queue, and the source node is added to the attack graph.

2 When the queue is not empty, a node is taken out as a current node to judge whether the current node is the target node.

3 When the current node is not the target node, the attack depth and the cost of the current node are judged to determine whether they are in the range of the maximum attack depth and the maximum attack cost.

4 When they are within the maximum range of requirements to find all the other state nodes that are connected with the current node.

5 Then, the program searches for all vulnerabilities in other state nodes. If an attack is reachable from the current node to a vulnerability of another state node, and a new state node is generated. The new state node is added to the queue, and the new state node which satisfies the preset condition is added to the attack graph too.

6 The step 2-5 is executed in the loop, until the queue is empty, and the attack graph is generated.

5. Experimental results.

5.1. The Design of Experimental.

In order to verify the feasibility of the optimized algorithm, a corresponding simulation experiment is done in this paper. It is assumed that attackers are wise, and always get the maximum profit with a minimum cost. Under feasible conditions of multiple attack paths, an attacker will choose a best path to attack which is in accordance with his inclination. The dependence coefficient K on attacker's experience during an attack is set to 0.5 in the experiment, it expresses that the cost of a second similar attack is half a cost of the original. The compactness coefficient of network ξ is set to 3. The network topology used in the experiment is shown in Figure 4. We also specialize in the use of expert experience and refer to the data and functions which are provided in the literature [12] and literature [14, 15]. There are seven workstations and five servers in the topology of network. The DNS server, WEB server, SMTP server and FTP server and monitoring host are connected to a same network through virtual LAN technology. Host H1 and H4 are distributed in different segments, host H5 and H6 can be accessed by host H4. Host H2 and H3 are connected to a same network, and host H5 and H6 and IDS file storage server are connected to another same network. All servers in the experiment are using Linux system, and all workstation hosts are using Windows system. All servers provide corresponding services, while a host can access another host of service vulnerability through the corresponding service. Host H1 open services with SMTP and FTP services, host H2 open HTTP service, host H3 open FTP and HTTP services, host H4 open TENET and SSH services, host H5 open FTP service, host H6 open HTTP service. All open services have corresponding vulnerabilities, and can be acquired through sophisticated vulnerability scanning tools [16, 17] such as Nessus, Xsan and so on. Some vulnerability information of host and server are shown below in table 4. Now host H2 hopes that through the vulnerability information which exist in the network accessing the file storage server S.

Table 4. Vulnerability Information of Host and Server

Number	Open Services	Vulnerability Number of CVE	Acx	Dsk	Description of Permission
H1	Smtp Web	{2015–7776/2015–7712}	0.1/0.3	0/0.275	Access to Smtp and Web Servers
H2	Http	{2016–1194}	0.7	0.64	Access to Web and Dns Servers
H3	Ftp Http	{2015–8878/2016–1194}	0.5/0.3	0.64/0.275	Access to Ftp, Web Servers
H4	Telnet Ssh	{1999–1211/2014–8730}	0.5/0.5	0.275/0.64	Access to Smtp and Dns Servers
H5	Ftp	{2015–8878}	0.5	0.64	Access to Ftp Servers
H6	Http	{2016–3385}	0.3	0.275	Access to Web Servers
Web	Http Telnet	{2016–1194/1999–1211}	0.7/0.5	0.64/0.64	Open Web and Telnet Services
Ftp	Ftp Http	{2015–7712/2016–3385}	0.3/0.3	0.275/0.275	Open Ftp and Web Services
Dns	Http Ssh	{2016–1194/2014–8730}	0.7/0.5	0.64/0.64	Open Web and Ssh Services
Smtp	Telnet Http	{1999–1211/2016–1194}	0.5/0.7	0.275/0.64	Open Telnet and Web Services
S	Http Ssh	{2016–3385/2014–8730}	0.3/0.5	0.275/0.64	Open Web and Ssh Services

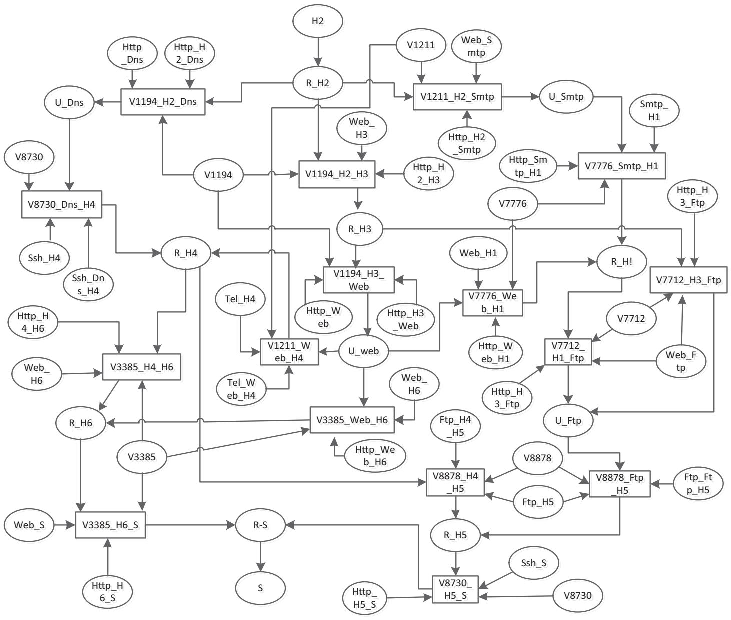


Figure 5. The Original Attack Graph.

5.2. **Experimental analysis and results.** The attack graph based on the NST model is shown in figure 5. According to the experimental design, there are 11 nodes in the network. Generating a complete attack graph should have 17 vulnerability attacks which based on 11 nodes and the vulnerability information of network. So there should be 17 sides. However, according to the cost formula and path analysis, results obtained are given in Table 5. The depth and cost of paths are shown in figure 6. According to expert experience, the Max_Depth value is set to 4, and the Max_Cost value is set to 3. Therefore, only 4, 5 and 6 meet requirements of the attack path. At the same time, according to results of the above table, since the path 5 has the shortest path and the minimum cost. So the path is the most likely attack path of attackers.

Table 5. The Table of Cost and Depth for Each Attack Path

Number	Path	Depth	Cost
1	(H2, Web) → (Web, H1) → (H1, Ftp) → (Ftp, H5) → (H5, S)	5	3.2
2	(H2, Smtp) → (Smtp, H1) → (H1, Ftp) → (Ftp, H5) → (H5, S)	5	3.0
3	(H2, Web) → (Web, H4) → (H4, H5) → (H5, S)	4	3.1
4	(H2, Dns) → (Dns, H4) → (H4, H6) → (H6, S)	4	2.3
5	(H2, Web) → (Web, H6) → (H6, S)	3	2.1
6	(H2, H3) → (H3, Ftp) → (Ftp, H5) → (H5, S)	4	2.7
7	(H2, H3) → (H3, Web) → (Web, H6) → (H6, S)	4	3.5

The optimized attack graph is shown in figure 7. It is through path analysis and correlation calculation, and then obtains the graph based on the INAG algorithm. According to simulation results, the comparison of Figure 5 and Figure 7 shows that the scale of attack graph is reduced based on the INAG algorithm. The optimized attack graph is more concise and more intuitive than the original attack graph, and it also can provide a reliable policy reference for network security managers to make a security policy for network.

6. **Conclusions.** On the basis of previous research results, we study an attack model, and puts forward a new idea for generation algorithm of attack graph. Firstly, quantization

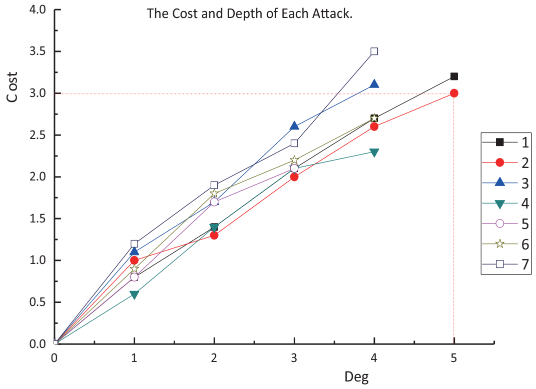


FIGURE 6. The Cost and Depth of Each Attack.

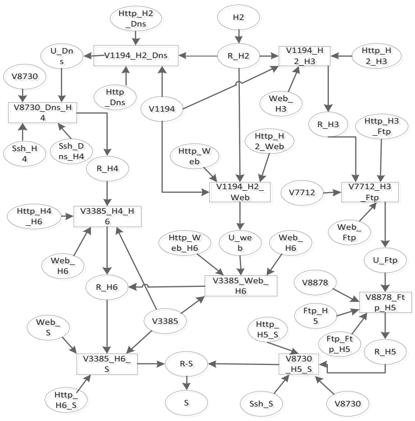


FIGURE 7. Optimized Attack Graph.

indicators of attack graph generation algorithm are given. Secondly, a new concept is introduced to quantitatively analyse the relationship between them. Based on results of quantitative analysis, one key element of the algorithm is calculated again. Finally, an attack graph based on the algorithm is completed. Before using the method provided in this article, the attack graph generated by the existing method is shown in figure 5. And the attack graph generated by this method is shown in figure 7. In this paper, the feasibility of this method is verified by the simulation experiment. The follow-up work will continue to further optimize the algorithm on the basis of existing results, and provide real-time and feasible solutions for the decision maker of network security.

Acknowledgement. This work is financially supported by Research Fund for the Doctoral Program of Higher Education of China (No. 20124116120004), and supported by Educational Commission of Henan Province of China (No. 13A510325).

REFERENCES

[1] China Internet Network Information Centre Report [EB/OL].
http://cnnic.cn/gywm/xwzx/rdxw/2015/201601/t20160122_53283.htm.

[2] Tencent Internet Security Report [EB/OL].
http://guanjia.qq.com/act/brand/201601safe/?ADTAG=web.gj_index.daohang_2015n.

[3] China Internet Network Security Report [EB/OL]. http://www.cert.org.cn/publish/main/index.html.

[4] 2015 Insider Threat Report | Insider Threat Security Statistics | Vormetric [EB/OL].
http://www.vormetric.com/campaigns/insiderthreat/2015/.

[5] W. H. Cunningham Optimal attack and reinforcement of a network, *[J]. Journal of the Acm,* 1985, 32(3): 549-561.

[6] R. W. Ritchey , P. Ammann, Using model checking to analyse network vulnerabilities , pp.156-165, 2000.

[7] P. Ammann, d. Wijesekera, S. Kaushik, Scalable, graph-based network vulnerability analysis, *ACM Conference on Computer and Communications Security, CCS 2002, Washington, Dc, Usa,* pp. 217-224, November. 2002.

[8] K. Ingols, R. Lippmann, Piwowarski K. Practical Attack Graph Generation for Network Defense, *Computer Security Applications Conference. IEEE Computer Society*, pp. 121-130, 2006.

[9] X. J. Chen , B. X. Fang , Q. F. Tan, et al., Inferring Attack Intent of Malicious Insider Based on Probabilistic Attack Graph Model , *Chinese Journal of Computers*, vol. 11, no.1, pp. 62-72, 2014.

[10] D. Man ,B. Zhang , W. Yang , et al., A Method for Global Attack Graph Generation, *IEEE International Conference on Networking, Sensing and Control. IEEE*, pp. 236-241, 2008.

[11] Q. Jun, W. Hongrun, Y. Yunfei, Z. Bojin, Effectiveness of Attack Strategies of Complex Networks with Cost , *Transactions of Beijing Institute of Technology*, vol. 33, no. 1, pp. 67-72, 2013.

[12] H. Wang, F. W. Chen, Y. F. Wang, An Approach of Security Risk Evaluation Based on the Bayesian Attack Graph , *Open Cybernetics & Systemics*, vol. 9, no. 1, pp. 953-960, 2015.

[13] A. Younis, Y. K. Malaiya, I. Ray, Evaluating CVSS Base Score Using Vulnerability Rewards Programs, *[M]// ICT Systems Security and Privacy Protection.,* 2016.

[14] C. Z.Wang , G. Q. Huang Network threat analysis based on vulnerability relation model , *International Journal of Security & Its Applications*, vol. 9, no. 1, pp. 357-368, 2015.

[15] K. Zhang, Analysis Method based on Rough Attack-defense Bayes Game Model , *International Journal of Security & Its Applications*, vol. 9, no. 1, pp. 109-118, 2015.

[16] N. I. Daud, K. A. A. Bakar, M. S. M. Hasan A case study on web application vulnerability scanning tools, *Science and Information Conference. IEEE*, pp. 595-600, 2014.

[17] H. Holm, Performance of automated network vulnerability scanning at remediating security issues , *Computers & Security*, vol. 31, no. 2, pp. 164-175, 2012.