Journal of Information Hiding and Multimedia Signal Processing ©2017 ISSN 2073-4212
Ubiquitous International Volume **8**, Number **2**, March **2017**

Behavior Analysis Research of Attackers Based on Path Revenue Calculation

Hui Wang*, Zhe Wang and Fuwang Chen

School of Computer Science and Technology
Henan Polytechnic University
No. 2001, ShiJi Street, GaoXin District, Jiaozuo Henan 454000, China
wanghui_jsj@hpu.edu.cn*, wangxiaozhe1026@163.com, chenfuwang123@163.com

Received July, 2016; revised December, 2016

Abstract. *In order to solve the problem of inaccurate calculation of node belief caused by redundant paths, by evaluating the feasibility of attacks, an attack path analyzing method is proposed. First of all, the method introduces the network vulnerability attack graph (NVAG), which can estimate feasibility of an attack by analyzing the cost-benefit of vulnerable nodes. Secondly, a weight accumulation method is proposed to identify all possible paths, and it will eliminate those potential redundant paths. Finally, the approach improves likelihood weighting algorithm based on Bayesian inference and increases the accuracy of node belief. Experimental results show that the method effectively excludes redundant paths of attack graph so as to improve the accuracy of node belief, and it achieves effective predictive analysis of the attack paths.*
Keywords: Redundant paths; Cost-benefit; node belief; Bayesian inference

1. **Introduction.** In quarterly reports of "Global network security market report [1] "which released by the U.S. Network security company Cybersecurity Ventures pointed out: The Network security market would reach $ 75.4 billion in 2015, while the market demand for information security solutions sustained high growth.

In essence, network attack events happening in computer network system are due to the loopholes in the computer system itself. In [2], the authors proposed a risk management framework using Bayesian networks that enable a system administrator to quantify the chances of network compromise at various levels. In recent years, researchers have begun to apply Bayesian network and attack graph to the prediction of attack behavior [3]. Bayesian network has the characteristics of processing uncertain data [4], and the attack graph can evaluate system based on vulnerability [5, 6, 7]. In [8], the authors built the network's three layers attack graph based on analysis of the underlying alarm data. Based on these graphs, Dantu and Kolan calculated the risk level of a critical resource using Bayesian methodology and periodically updated the subjective beliefs about the occurrence of an attack [9, 10]. Finally, attack graph plays a role of comprehensive evaluation system security trends. The NAGD algorithm was defined in [11] which simultaneously decompose network attack graph into several sub-attack graphs which one-one corresponding to a specific vulnerability exploiting threat.

The thesis of [12] introduced the attack graph model of judgment for internal attack intention. Based on this model, the author presented an algorithm to infer the internal attack intention and a method of maximum probability paths aiming at the attack target. The algorithm had been tested on simulated networks. The experimental result showed

the approach could be applied to large-scale networks [13]. In [14], they built an example of Bayesian network based on a current security graph model, justified the approach of their model through attack semantics and experimental study, then showed that the resulted Bayesian network was not sensitive to parameter perturbation. In [15], the authors proposed a prediction method of attacker has selective attack based on the attack cost. In [16], it does not account for redundant paths, which not only affected the optimization effect of the attack graph, but made the cost-benefit calculation was not comprehensive. To address above problems, our contributions in this paper are summarized as follows.

(1) Firstly, the feasible calculation method is proposed, which is based on the analysis of the cost-benefit of the vulnerable nodes.

(2) Secondly, our research improves likelihood weighting algorithm and adopts the method of weight accumulation to improve the AND node path selection problem, further more effectively calculates the problem of node belief with the attack paths.

The rest of this paper is organized as follows. In Section 2, we introduce the related works including Network Attack Path (NAP) and Network Vulnerability Attack Graph (NVAG). The analysis of Attack feasibility is presented in Section 3. We propose the improved likelihood weighting algorithm in Section 4. The experimental results and comparing the performance of the proposed algorithm with previously proposed methods are presented in Section 5. Finally, conclusions and future works are discussed in Section 6.

2. **Related works.**

2.1. **Network Attack Path (NAP).**

The Attack graph is a network vulnerability analysis model in [17].

Definition 1. When an attacker attacks the network target resources, firstly, the attacker attacks the initial resource node, and then attacks other resource nodes. Repeat these action, until the attacker possessed of target node. The running track in this process of attacker is network attack path (NAP).

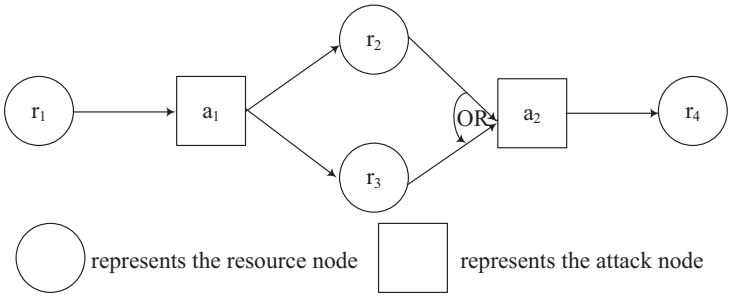

FIGURE 1. A simple network attack graph

In Figure 1, the attacker starts from the start node r_1, through a_1, r_2, a_2 or a_1, r_3, a_2, and finally reaches the target node r_4. Among them, the nodes sequence are composed of r_1, a_1, r_2, a_2, r_4 is a NAP. This ordered node that composed of r_1, a_1, r_3, a_2, r_4 is also a NAP.

2.2. **Network Vulnerability Attack Graph (NVAG).**

The model of Network Vulnerability Attack Graph (NVAG) is a depiction of the relationship among the state of the network resource, the vulnerable nodes, the attack behavior and the attack revenue. The definition combined with actual in this paper, when the attacker's interest is less than the cost of attack, the attacker will not attack the target node.

Definition 2. NAVG is a directed acyclic graph with one or more AND-OR nodes. The AND relationship represents all the child nodes ($\mathrm{Kid}_{(n)}$) need to meet the directive

condition simultaneously before reach the parent node ($\text{Father}_{(n)}$). The OR relationship indicates that all the child nodes meet any directive condition can to reach the parent node. The definition of NAVG is as follow: NVAG = (R, A, E, Val(r_i), P, C).

1) $R = \{r_i \,|i = 1, 2, 3 \cdots N\}$ is a set of nodes standing for network resource. The node variable r_i values 1 or 0. $r_i = 1$ represents that the attacker has succeeded in possessing this resource nodes. $r_i = 0$ denoted that the attacker did not succeed in possessing this resource nodes. r_o indicated the resource nodes already obtained by the attacker in the start state. r_g represents the resource nodes that the attacker is ultimately to possessed.

2) $A = \{a_j | j = 1, 2, 3, \ldots N\}$ stands for a collection of attack nodes, it is a non-empty and definite AND-OR set and the node variable valued true or false. The set is described by 3-tuple (r_i, a_j, r_k): when the resource node r_i is acquired, the occurrence condition of attack behavior a_j is fulfilled (a_j = true), and then the attacker can launch attacks to get the resource node r_k. On the contrary, when the resource node r_i is not satisfied, the occurrence condition of attack behavior a_j is not fulfilled (a_j = false), the attack will not happen, and the attacker do not launch attacks to get the resource node r_k.

3) $E = \{e | e \in (R \times A) \cup (A \times R)\}$ denotes the connection of directed edges between attack nodes and resource nodes in the network attack graph. $e_1 = <a_j, r_i> \in A \times R$ represents a_j is the attack behavior aiming at the possess resource r_i, when the weight $\Phi_j \geq 1$, the directed edge $e_{<aj,ri>}$ = true, otherwise, $e_{<aj,ri>}$ = false. On the contrary, $e_2 = <r_i, a_j> \in R \times A$ means the attacker first possess the resource node r_i, and then attack behavior a_j occurred.

4) $\text{Val}(r_i) = \{r_i | i = 1, 2, 3 \ldots N$, Val($r_i$) is a resource value set of resource nodes.$\}$ Among them, the asset value of resource node r_i is measured by the following four related factors: the disclosure of corporate secrets (CS), the disclosure of personal information (PI), range of influence (RI) and the damage of property (PD). Thus, the value of asset Val(r_i) is calculated as follows:

$$\text{val}(r_i) = \partial\left(\text{CS}(r_i), \text{PI}(r_i), \text{RI}(r_i), \text{PD}(r_i)\right) = w_1 L_{\text{cs}} + w_2 L_{\text{PI}} + w_3 L_{\text{RI}} + w_4 L_{\text{PD}} \quad (1)$$

where W_1, W_2, W_3, W_4 are the weights of asset value of the measure factors that are associated with the resource node r_i. $L_{\text{CS}}, L_{\text{PI}}, L_{\text{DS}}, L_{\text{PD}}$ are the degrees of equivalence of the elements.

5) $P = \{p | p \in P_1 \cup P_2$, where P_1 means the conditional probability distribution of the attack behavior a_j occurred, P_2 stands for the conditional probability distribution of the attack behavior a_j succeed$\}$. Triple $<r_i, a_j, r_k>$ is described as: When r_i is occupied, the conditions of attack behavior a_j is satisfied, a_j can choose to attack or not, so p = (a_j = true $|r_i$ is occupied) $\in [0, 1]$, where $p \in P_1$. If a_j chooses to attack and occupy the resource node r_k. There are two kinds of results, namely, success and failure. Thus, p = $\{a_j$ succeed $|a_j$ attack$\} \in$ [0, 1], where $p \in P_2$.

6) $C_{(n)}$ is the distribution of node belief, $n \in$ A$\cup$ R$\wedge C_{(n)} \in$ [0, 1]. Where $C_{(aj)}$ means the probability of attack when the condition is satisfied, $C_{(aj)} \in$ [0, 1]. $C_{(ri)}$ indicates the probability possessed r_i successfully under the premise of the occurrence of attack behavior, $C_{(ri)} \in$ [0, 1]. In addition, $C_{(ro)} = 1$ is stand for the resource node which was occupied in initial condition.

3. The feasibility analysis and generation algorithm of NAP.

3.1. The feasibility analysis of NAP.

(1) Income analysis of vulnerability nodes

Attack behavior make the use of the vulnerability node to attack, rules of use is Rule = (Pre-resource, Vul, Post-resource). When the Pre-resource of attack is met, the attack can be initiated in the network according to the vulnerability of the node, and when

the attacker launches the attack successfully and the resources after the attack can be obtained.

Pre-resource is the node to earn the resource by vulnerable of nodes, and combines the acquired nodes with node vulnerability for the next attack, and then obtains the target resource ultimately. And the common vulnerability utilization approaches include: MCPr (control the tamper of program), MCPa (change control parameters), MMPa (change the measurement parameters), SPa (intercept key data information), GPr (indirect access to the server to hunt for control authority or the password).

In order to prohibit illegal behavior that unauthorized user, and to ensure the safety of the equipment and the controlled objects, different levels of authority managements are often carried out in [18]. (As shown in Table 1)

Table 1. Classification and description of control authority

Control authority	Description of control authority
FCC	The attacker has the ability to fully control the component
MCP	The attacker can modify the parameters of the control component.
RE	The attacker has the ability to read and execute the control component
LCC	The attacker has the ability to list control components
W	The attacker has the ability to write to the control component
R	The attacker has the ability to read the control component
N	The attacker has no control over the control component

Attack result is Post-resource = (Authority, Gain), which means the control authority levels of network component obtained by attacker and the benefit used the vulnerability successfully. For the benefit of an attack path with j times of network attack (that is, a_j to r_i):

$$\text{Gain}_j = \text{Val}(r_i)\lambda_j\alpha_j \ 1 \le i \le N, \ 1 \le j \le N \tag{2}$$

In (2): $\text{Val}(r_i)$ is the corresponding asset value that the network attack node a_j to the resource node r_i; λ_j is the success of the j times attack, and it obtains the corresponding level of weight of network components; α_j is defined as an influence factor that attacker will be benefited from the vulnerable utilization in a piece of attack behavior.

The calculation method of the Gain_j obtained by the single use of the vulnerable success is given: Firstly, all parameters are given in the form of rank, and they are carried out the initial quantization. The weight value disposable should follow the below partial order relation: PCC > MCP > RE > LCC > W > R > N. The quantitative of vulnerability influence coefficient should follow the partial order relation: MCPr > MCPa > MMPa > SPa > GPr; (As shown in Table 2)

Secondly, we determine the weight w in assets value of corresponding consequences factors with estimation-matrix method in [19], We select m (m = 10–30) field experts, the ratio of importance degree of each two consequence factors is given by them, so as to construct the judgment matrix of m with $4 \times 4 S^{(e)}$ ($e = 1, 2, \ldots, m$).

$$S^{(e)} = \begin{bmatrix} S_{11}^{(e)} & \cdots & S_{14}^{(e)} \\ \vdots & & \vdots \\ S_{41}^{(e)} & \cdots & S_{44}^{(e)} \end{bmatrix} \tag{3}$$

The element equation of $S^{(e)}$ express consequence attribute w_q given by field expert e relative to the important degree of consequence attribute w_p. After obtain the every two judgment matrixes $\{S^{(1)}, S^{(2)} \ldots S^{(m)}\}$ given by m experts, the geometric average method

TABLE 2. Equivalent classification of parameters

Parameter	Equivalent level
CS The importance level of CS:	1, public; 2, insider; 3, secret; 4, classified; 5; top secret;
PI The leakage range of PI:	1, none; 2, samll range; 3, medium range; 4, big range; 5; maximum range;
RI The importance range of RI:	1, none; 2, samll range; 3, medium range; 4, big range; 5, maximum range;
PD The level of PD:	1, one hundred thousand blew; 2, one hundred thousand to five hundred thousand; 3, five hundred thousand to one million; 4, one million to ten million; 5, more than ten million;
α	MCPr:3; MCPa:2.5; MMPa:2; Spa:1.5; GPr:1;
λ	PCC:1.2; MCP:1.0; RE:0.8; LCC:0.6; W:0.4; R:0.2; N:0;

is first used to synthesize the matrix, and the matrix S is obtained. The elements S_{pq} of S are calculated as follows:

$$S_{pq} = \sqrt{\prod_{e=1}^{m} S_{pq}^{(e)}} \; p, q = 1, 2, 3, 4 \tag{4}$$

Finally, we solve the problem of eigenvalues and eigenvectors $Sw = \gamma_{\max} w$ and normalize the main eigenvector w can get the consequence attribute weight coefficient vector $w' = (w_1, w_2, w_3, w_4)^{\mathrm{T}}$. Then, w_1, w_2, w_3, w_4 can be obtained.

(2) The cost breakdown of Vulnerability nodes

Vulnerability attack cost is mainly determined by the following three factors: The difficulty degree of the attack D, the hidden degree of vulnerability H and the time to attack successfully T. The attack cost of a single vulnerable point can be expressed as $\mathrm{Cost}_i = \beta_1 D + \beta_2 H + \beta_3 T$, among them, $\beta_1, \beta_2, \beta_3$ are the relative weight of the corresponding factors. In this paper, the algorithm assumes that a maximum attack cost $\mathrm{Cost}_{\max}$, on the one hand, we can find the path of high benefits compared to the attack benefits. On the other hand, we can limit the depth of attack, and reduce the path of the attack who makes little sense.

(3) The analysis of Attack feasibility

Before the implementation of network attacks, the attacker will evaluate and analyze the cost-benefit of attack nodes. Only when the benefits of the attack behavior in its acceptable range, the attacker will think the attack is feasible. Therefore, the following formula can be used to determine the feasibility of this sub attack path.

$$\Phi_j = \frac{\mathrm{val}(r_i)\lambda_j \alpha_j}{\beta_1 D + \beta_2 H + \beta_3 T} = \frac{\mathrm{Gain}_j}{\mathrm{Cost}_j} \approx \frac{\mathrm{Gain}_j}{\mathrm{Cost}_{\max}} \; 1 \le i \le N, \; 1 \le j \le N \tag{5}$$

As shown in (5), the attack path feasibility Φ_j is the ratio of the attack benefit and attack cost of attacker. The attack behavior will occur when $\Phi_j \ge 1$ that the attacker can gain more than the cost of his own.

3.2. **The generation algorithm of NAP. Definition 3**. For any two adjacent nodes m, n in the attack graph, if there is a directed edge from m to n, then there exist the partial order relation between m and n, represent by $<m, n>$. The set composed of a variety of partial order is called POS, and every NAP is a POS.

The generating process of NAP as follows: in the first place, the weight $\Phi(a, b) = a/b$ is gave in attack graph, a means the weight of the attack benefit in $\Phi(a, b)$, b means the

weight of attack cost. In the second place, r_0 as the started node and cut an edge which is the child node, add to the collection POS_i, which the AND relationship indicated by the symbol of "$\wedge$". The POS that cut off each node in turn according to the topological order of the child node to the target parent node in attack graph, denote as POS_j, and credited to the $\text{POS}_{(j+1)}$. We continue to repeat the operation, and obtain the attack path ultimately.

4. **Improved likelihood weighting algorithm.** In the attack graph, the computation of node belief is an important basis to judging the attack paths. Logic sampling method needs to abandon the sample and results in a waste of resource. The traditional likelihood weighting method in [20] cannot solve the problem of node weight and edge identification of AND relationship very well, and this method cannot determine the redundant paths pretty good. This research develops an improved method of cumulative calculation, which can increase the accuracy of removing redundant paths. Node belief calculation problem is to obtain the probability distribution $C_{(n)}$ of all nodes on this path. As a result, the node belief of the attack graph can be described as follows:

Algorithm 1 describes the likelihood weighted of generative process of NAP, traversal of each node variable (X) in attack graph according to the topological order of nodes. First of all, determine the direction of the directed edge, if it is the directed edge from attack node to the resource node $<a_j, r_i>$, and judge the weight, if $\Phi_j \geq 1$, then $e_{<aj,ri>} =$ true, otherwise $e_{<aj,ri>} =$ false, and then identified the node. If $e_{<aj,ri>} =$ false, then it means to give up the attack (a_j). Then before sampling, the method gives X with value of false, and make this node as a random variable with a fixed value, and then sample. The specific algorithm 1 as follows:

Algorithm 1 The generation of NAP with a state label

Input: The NVAG, the weight, linear order relation set NAP, $<a_j, r_i>$ directed edge set e_1, partial ordered set (POS), AND relation set M, topological order Ψ, arbitrary node X and Y.
Output: The linear relationship set NAP with state flag.

- $\Psi \leftarrow$ NVAG
- $\text{POS}_i \leftarrow \emptyset$, $\text{NAP}_i = \emptyset$, $M = \emptyset$
- For (each node variable X in Ψ)
- To find node variable X that has a partial order relation with Y
- IF $(<X, Y> \in e_1)$
- IF $\Phi_{<X,Y>} \geq 1$
- $e_{<X,Y>} \leftarrow$ true
- ELSE
- $e_{<X,Y>} \leftarrow$ false
- $<a_j, r_i> \leftarrow$ give up
- IF $(<X_1 \cup X_2, Y>)$
- Algorithm 3
- $M \leftarrow M \cup \{X_1 \cap X_2\}$
- $\text{POS}_{i+1} \leftarrow \text{POS}_i \cup \{<M, Y>\}$
- ELSE
- $\text{POS}_{i+1\leftarrow}$ $\text{POS}_i \cup \{<X, Y>\}$
- END IF
- END IF
- END IF
- END FOR
- NAP $\leftarrow$ POS
- Iterating through all edges $<a_j, r_i>$ that the value of attack indicator is false.
- RETURN NAP_i

Algorithm 2 depicts the improved likelihood weighting algorithm, if X is the evidence variable, operate the observed value x of X as the sampling result, and act the probability of sampling as the value of the sample weight. If X is not the evidence variable, according to the logic sampling, sample the remaining nodes without initial node according to the probability distribution $P(X|\text{Kid}_{(n)})$. n samples is obtained through the sampling, among them, there are n_z samples which meet the evidence variables $z \in Z$, and the corresponding weight is Δ_z; there are n_{zo} samples which meet the query variables $o \in O$,

and the corresponding weight is Δ_{zo}; Then, the posterior probability obtained according to the Bayesian inference and the prior probability, the formula is as follows:

$$P\left(O=o|Z=z\right)=\frac{P(O=o)P(Z=z|O=o)}{P(Z=z)}\approx\Delta_{zo}/\Delta_{z} \tag{6}$$

Algorithm 2 Improved likelihood weighting algorithm

Input: the NVAG, effective sample size n, evidence variables set Z, evidence variables value z, query variables set O, query variable value o, node topological order Γ, arbitrary node X.
Output: Δ_{zo}/Δ_{z}.

- $\Gamma \leftarrow$ NVAG
- $i \leftarrow 0$, $\Delta_z \leftarrow 0$, $\Delta_{zo} \leftarrow 0$
- WHILE $(i < n)$
- $P_i \leftarrow \emptyset$
- FOR (each node variable X in Γ)
- $G_x \leftarrow$ false
- IF $(X \in G_x)$
- $P_x \leftarrow 0$
- ELSE
- IF $X \in O$ and X is a root note.
- Mark X as sample
- ELSE
- $x \leftarrow$ the sampling result according to $P(X|\mathrm{Kid}(n))$
- END IF
- END IF
- END FOR
- $p_i \leftarrow p_i \cup \{X = x\}$
- $\Delta_i \leftarrow \prod_{x\in Z} P(X|\mathrm{Kid}_{(x)})|p_i$
- $\Delta_z \leftarrow \Delta_z + \Delta_i$
- WHEN $(X \in Z)$
- IF $(X \in O)$ then
- $\Delta_{zo} = \prod_{x\in O} P(X \in O|X \in Z|P_{x\in Z}$
- $\Delta_{zo} \leftarrow \Delta_{zo} + \Delta_i$
- $C_{(x)} \approx \Delta_{zo}/\Delta_z$
- END IF
- $i \leftarrow i + 1$
- END WHILE
- RETURN Δ_{zo}/Δ_z

Nodes of AND relationship in attack graph, this study adopts the concept of weighted accumulation to judging whether the label is false. The specific algorithm 3 is as below:

Algorithm 3 The give up note select of AND relationship

Input: Linear order relation set POS_i, directed edge set of $<X_1 \cup X_2, Y> \in e_1$, directed edge set M of AND relationship, arbitrary node X and Y.
Output: give up attacking node set of $M_x \leftarrow M$.

- $M_x \leftarrow \emptyset, \Phi \leftarrow \emptyset$
- FOR $(<X1 \cup X2, Y> \in e1)$
- $\Phi_1 \leftarrow$ Calculate the weight of $<X_1, Y>$
- $\Phi \leftarrow \Phi_1 + \Phi$
- $\Phi_2 \leftarrow$ Calculate the weight of $<X_2, Y>$
- $\Phi \leftarrow \Phi_2 + \Phi$
- IF $\Phi \geq 2$
- $e_{<X1\cup X2,Y>} \leftarrow$ true
- ELSE
- $e_{<X1\cup X2,Y>} \leftarrow$ false
- $e_{<X1\cup X2,Y>} \leftarrow$ give up
- $M_x \leftarrow_{<X1\cup X2,Y>}$
- END IF
- RETURN M_x

Algorithm 3 aims to finding out the resource status node with state of false in set M. First of all, the algorithm judges the directed edge with M relationship in set e_1, and calculates Φ of two edges, and then adds them up. If $\sum \Phi_j \geq 2$, then the mark of directed edge is false, otherwise the tag is true. Store the directed edge with mark of false into M_x. This algorithm can obtain the node set of redundant paths more accurately.

5. **Node belief computation examples.** For example, in Figure 2, the calculation results of the node belief by using the traditional Bayesian inference algorithm are shown in Table 3 (The number of effective sample is 5000 in the experiment. c and d are assumed, c represents $P_1 = 0.6$ and $P_2 = 0.9$, d means $P_1 = 0.9$ and $P_2 = 0.6$).

TABLE 3. Traditional Bayesian inference result

	$c(r_0)$	$c(a_1)$	$c(a_2)$	$c(a_3)$	$c(r_1)$	$c(r_2)$	$c(a_4)$	$c(a_5)$	$c(a_6)$	$c(r_3)$	$c(r_4)$	$c(a_7)$	$c(r_5)$
c	1	0.57	0.58	0.56	0.26	0.49	0.16	0.28	0.05	0.40	0.04	0.26	0.23
d	1	0.88	0.86	0.87	0.28	0.52	0.25	0.46	0.11	0.43	0.06	0.44	0.27

As example of taking the weight parameters given in the attack graph is shown in Figure 2, the computational results of nodes belief by using improved likelihood weighting algorithm are shown in Table 4.

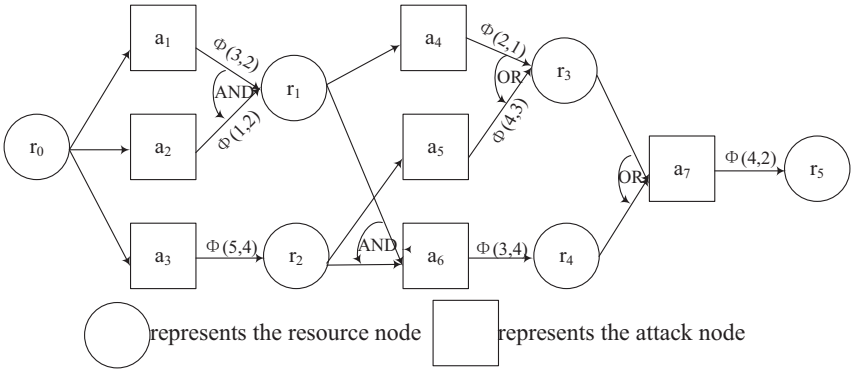

FIGURE 2. A typical network attack graph with weight

TABLE 4. Improved likelihood weighting inference result

	$c(r_0)$	$c(a_1)$	$c(a_2)$	$c(a_3)$	$c(r_1)$	$c(r_2)$	$c(a_4)$	$c(a_5)$	$c(a_6)$	$c(r_3)$	$c(r_4)$	$c(a_7)$	$c(r_5)$
c	1	0.56	0.54	0.58	0.24	0.52	0.14	0.31	0	0.40	0	0.24	0.21
d	1	0.84	0.87	0.85	0.26	0.50	0.22	0.44	0	0.38	0	0.34	0.20

As shown in Table 3 and Table 4, compared with the results between improved algorithm and traditional algorithm, the change of node belief can be seen. In order to observe the change of data more intuitively, this paper takes NAP_1 and NAP_2 as an example, and draws the graph node belief variation graph Figure 3 and Figure 4 represent the two paths node structure and the distribution of the node belief respectively. (Where Tc means the change of node belief in condition c of traditional algorithm, Td represents the change of node belief in condition d of traditional algorithm, Ic and Id denote the change of node belief under two different conditions of improved algorithm).

The results is shown in Figure 4 and Figure 5, the node belief will be different through different algorithms. There are two alterations can be found in Figure 4 and Figure 5. The first one, the node whose belief is 0 has occurred in the improved algorithm, but it isn't appeared in the traditional algorithm. The next one, the node belief obtained by improved algorithm is significantly smaller than that obtained by traditional methods. The traditional algorithm does not consider redundant paths into account, therefore, it thinks that all attacker would choose to attack and not gave up any paths. And the traditional algorithm doesn't consider the benefit and cost of attack. Furthermore, the belief of target node is obviously higher. However, the improved algorithm considers the redundant paths into account. The attacker would choose the most beneficial attack paths to themselves and eliminate the pointless ones, so the belief of target node will naturally decrease due to the reduction of attack paths.

As shown in Figure 3, the attack graph includes all data in Table 4 obtained from the weight parameters. It can be observed that the node belief of a_2 is not set to 0, which is

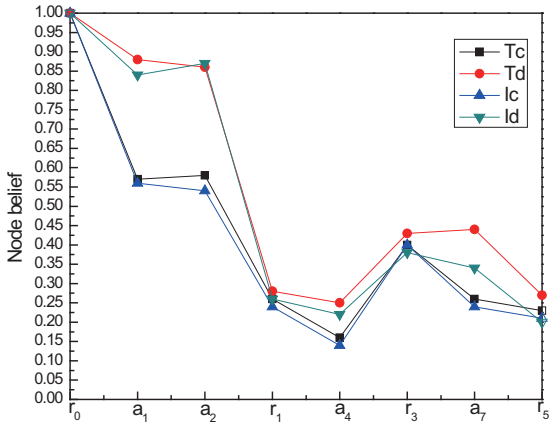

FIGURE 3. Comparison diagram of node belief in NAP_1

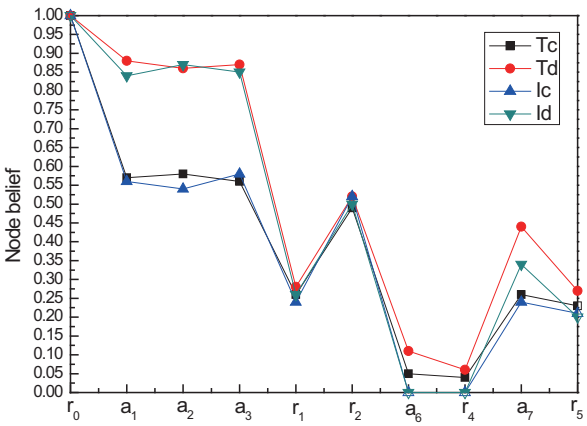

FIGURE 4. Comparison diagram of node belief in NAP_2

another improvement of this paper. The premise of AND node with the false mark is not to calculate the weight of one side, but accumulate the weight of two or more edges, and then determine whether to give up the attack.

The method proposed in this paper is more effective to predict the attack path than other methods in [6, 8, 20]. And in section 3 of this paper, different from [8], we put the value of node vulnerability into the calculation of the attack revenue, and accurately analyze the attack feasibility. Then, we can determine the possibility of network attacks more precisely. In [20], in the calculation of the confidence of the attack node, it doesn't think of the problem of the path selection of the AND node in the attack graph, so the attack path is not accurate. In this paper, we propose a new algorithm to solve the problem of AND node path selection, as shown in algorithm 3.Combined with improved likelihood weighting algorithm, this paper not only reduces the redundant paths effectively, but also avoids ignoring of loophole of AND node relationships. Thus, the algorithm improves the accuracy of the predicted path effectively.

6. **Conclusion.** Combined with the node vulnerability, this paper translates the node confidence calculation into attack behavior cost-benefit calculation by defining the model of NVAG. The method reduces the redundant paths by identifying nodes of lower weights. Then, Bayesian inference algorithm is proposed in the method, and it further improves the accuracy of node confidence. Finally, this paper puts forward the weight accumulation method which can solve the node identification problem in AND relationship. In this way, the algorithm neither increases redundant paths, nor miss the effective paths of attacks which the attackers may choose.

The experimental results show that the proposed method can be more effective to predict the attack paths and calculate the node confidence to reduce the redundancy paths. In a word, a better preventive strategy for network security management is provided in this paper.

Acknowledgement. This project is 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] Global network security market report in 2015[EB/OL] http://www.chinacloud.cn/show.aspx?id=20054&cid=11

[2] N. Poolsappasit, R. Dewri, I. Ray, Dynamic Security Risk Management Using Bayesian Attack Graphs[J]. *IEEE Transactions on Dependable & Secure Computing*, vol. 9, no. 1, pp. 61–74, 2012.
[3] C. Zhao, H. Wang, J. Lin, Lv H & Y. Zhang, A Generation Method of Network Security Hardening Strategy Based on Attack Graphs [J]. *International Journal of Web Services Research*, 12, (2015).
[4] Yue K, Wu H, Liu W & Zhu Y. Representing and Processing Lineages over Uncertain Data Based on the Bayesian Network [J]. *Applied Soft Computing*, 37(C): 345–362, (2015).
[5] M. Keramati, A. Akbari, M. Keramati, CVSS-based security metrics for quantitative analysis of attack graphs. *International Conference on Computer and Knowledge Engineering*, (2013) October 31 & November 1; Mashhad, IRAN
[6] T. Harada, A. Kanaoka, E. Okamoto & M. Kato, Identifying Potentially-Impacted Area by Vulnerabilities in Networked Systems Using CVSS. *2012 IEEE/IPSJ 12th International Symposium on Applications and the Internet IEEE*, (2010) July 19–23; Seoul, Korea.
[7] C. Zhang, L. Hu, Nurbol, Construction Method and Realization of Datasets under Wireless Network WEP Attack [J]. *Journal of Jilin University (Science Edition)*, (1): 71–75, (2014).
[8] Z. Y. Luo, B. You, J. Z. Xu, et al, Automatic recognition model of intrusive intention based on three layers attack graph[J]. *Journal of University (Engineering and Technology Edition)*, 2014, 44(5): 1392–1397.
[9] D. T. Ram, Risk Management Using Behavior Based Bayesian Networks[M]// *Intelligence and Security Informatics Springer Berlin Heidelberg*, 2005.
[10] Z. M. Lu, Y. P. Feng. Information Entropy and Cross Information Entropy Based Attacking Methods for Complex Networks,[J]. *Journal of Information Hiding and Multimedia Signal Processing*, Vol. 7, No. 6, pp. 1243-1253, November 2016.
[11] WU Di, LIAN Yi Feng, CHEN Kai et al. A Security Threats Identification and Analysis Method Based on Attack Graph[J]. *CHINESE JOURNAL OF COMPUTERS*, 2012, 35(9): 1938–1950.
[12] X. J. Chen, B. X. Fang , Q. F. Tan et al. Inferring Attack Intent of Malicious Insider Based on Probabilistic Attack Graph Model [J]. *Chinese Journal of Computers*, (2014).
[13] Y. Ye, X. S. Xu, Z. Qi & X. Wu, Attack Graph Generation Algorithm for Large-scale Network System[J]. *Journal of Computer Research and Development*, 2013, 50(10): 2133–2139.
[14] P. Xie, J. Li, X. Ou, et al., Using Bayesian Networks for Cyber Security Analysis[C]// *2013 43rd Annual IEEE/IFIP Nternational Conference on Dependable Systems and Networks (DSN) IEEE*, 2010: 211–220.
[15] H. R. Wu, J. Qin, B. J. Zheng, Anti-attack Ability Based on Costs in Complex Networks[J]. *Computer Science*, vol. 39, no. 8, pp. 224–227, 2012.
[16] R. Dewri, I. Ray, N. Poolsappasit & D. Whitley, Optimal security hardening on attack tree models of networks: a cost-benefit analysis [J]. *International Journal of Information Security*, vol. 11, no. 3, pp. 167–188, 2012.
[17] Wang X, Sun B, Liao Y et al. Computer Network Vulnerability Assessment Based on Bayesian Attribute Network[J]. 2015.
[18] M. Z. Gao, D. Q. Feng, C. I. Ling, Vulnerability analysis of industrial control system based on attack graph[J]. *Journal of Zhejiang University (Engineering Science)* , vol. 12, pp. 2123–2131, 2014.
[19] Chien-Ming Chen, Linlin Xu, Tsu-Yang Wu, and Ci-Rong Li. On the Security of a Chaotic Maps-based Three-party Authenticated Key Agreement Protocol[J]. *Journal of Network Intelligence*, vol. 1, no. 2, pp. 61-66, May 2016.
[20] Y, F. Wang, H.-Wang, Research on Predicting Attack Paths Based on Bayesian Inference[J]. *Information Technology Journal*, vol. 12, no. 14, pp. 2712–2718, 2013.