Mobile Networks and Applications (2021) 26:1630–1637
https://doi.org/10.1007/s11036-019-01500-7

Network Risk Assessment Based on Baum Welch Algorithm and HMM

Chundong Wang[1,2,3] · **Kongbo Li**[2] · **Xiaonan He**[4]

Published online: 6 February 2020

Abstract

With the increasingly extensive applications of the network, the security of internal network of enterprises is facing more and more threats from the outside world, which implies the importance to master the network risk assessment skills. To improve the accuracy is an importent issue. In the big data era, there are various security protection techniques and different types of group data. Meanwhile, Online Social Networks (OSNs) and Social Internet of Things (SIoT) are becoming popular patterns of meeting people and keeping in touch with friends (Jiang et al. ACM Comput Surv 49:10:1–10:35 2016; Shen et al. 2017). Risk assessment, as a bridge between security experts and network administrators, whose accuracy can influence the judgment of administrators to the entire network state. In order to solve this problem, this essay uses the Baum Welch algorithm to optimize the risk assessment process by establishing the HMM model, which can improve the accuracy of the evaluation value. Firstly, behavior of the attacker is described in-depth by the attack graph generated through MulVAL framework. Then, the nodes on the attack path can will be evaluated and the value will be further evaluated by the Bayesian model. Finally, by establishing the hidden Markov model, the corresponding parameters can be defined and the most likely probabilistic state transition sequence can be calculated by using the Viterbi algorithm and Baum Welch algorithm to deduce the attack intent with the highest possibility.

Keywords Network security assessment · HMM · Baum Welch · Attack graph

1 Introduction

Increasing cyber attacks have attracted high attention in contemporary data security and network security studies. In wireless sensor network, target tracking [10] and data gathering and aggregating [22]have became more and more concerned. In relaying network, researchers guaranteed data transmission from the source to the destination [2, 8] and the research on performance of primary transceiver and performance of secrecy and kept data from eavesdropping and [3, 21]. There are more and more technologies which can prevent us fromy potential to threats and protect our private information, such as information entropy modularity and biometrics-based authentication [16, 24]. Furthermore, machine learning as a new technonlogy can be widely used to analyse and evaluate in many computer yields and research directions, such as the visual tracking [5], which is used to identify IP traffic [1]. It is argued that contemporary personal data can hardly be safe both in the hard-storage and the soft-storage such as RAM and network [9]. This is because personal data which is available in the social network could be gathered extensively by attackers [11]. The main factor which causes this problem is that the large-scale computer network and enterprise network have relatively more or less vulnerabilities. Eexternal attackers can easily take advantage of these vulnerabilities; therefore, security policies are particularly important. A detailed

✉ Chundong Wang
michael3769@163.com

Kongbo Li
vincy3319833@163.com

Xiaonan He
hexn@ehualu.com

[1] Key Laboratory of Computer Vision and System, Ministry of Education, Tianjin University of Technology, Tianjin, 300384, China

[2] Tianjin Key Laboratory of Intelligence Computing and Novel Software Technology, Ministry of Education, Tianjin University of Technology, Tianjin, 300384, China

[3] School of Computer Science and Engineering, Tianjin University of Technology, Tianjin, 300384, China

[4] Tianjin E-Hualu Information Technology Co., Ltd, Tianjin, China

vulnerability analysis of the complex network can cost a lot of time, funds, and resources, so the most effective strategy can be the network risk assessment. Based on above situations, two approaches have been considered: (1) assess the potential risk one by one; (2) detecting existing vulnerabilities which can be used by attackers through the overall deduction of a series of vulnerabilities.

Cyber attack is the process conducted by an attackers who is implementing the information access and enhancing the information permission based on the attack conditions and goal. The attack depends on the ability of attackers, experiments and the control environment. Prerequisites regarding cyber attacks are shortcoming in the contemporary network(or system). In addition, due to the interrelations among these vulnerabilities, the host devices have established mutual trust, which can be used by cyber attackers to continue the attack after a specific completed attack. Therefore, cyber attacks are usually a complex, multi-step process. In order to explain the process of cyber attack, a number of researchers have proposed risk assessment methods by building security models of network systems through paradigms such as attack graphs.

In order to build such a comprehensive model regarding network attack relationships, a range of challenges have to be overcome. We have to correlate data from numerous resources, which include topology, vulnerabilities, and configurations, into an integrated model. The construction of the model representation and persistence must be flexible and can be easily extended.

However, it is very difficult to use only one method to process the system vulnerability analysis and generate optimal safety management strategies. Since the test result remains uncertain, it is not possible to accurately infer the attack intention. Thus, information and probabilities of the attack graph are further explored by using Hidden Markov Model. HMM is applied to detect uncertainties of those observable states and attack states. Then, a probabilistic mapping between network observations and attack states can be generated by HMM. Parameters of the model are redefined through the Baum Welch algorithm and the maximum probability state transition sequence is further calculated by using the Viterbi algorithm. Based on these processes, the intention of attackers has been finally inferred. According to the experimental results, the maximum probability path with the network topology and configurations has been demonstrated.

The attack intention can be accurately inferred by this comprehensive model. This method provides a good representation of network security administrators and equips them with some security strategies to overcome existing shortcoming in the enterprise network.

2 Related work

Network security risk is propagative and network security risk will be the target in network through its multiple vulnerabilities between relevant services and hosts. Wang et al. [12] considered the difficulty of attack, the cost of reconfiguration of the network and the value of key information assets in the network based on the attribute attack graph, put forward the network security measurement method. Feng et al. [4] put reliability ideas into the attack graph to analyze the vulnerability of the network. Shi jin et al. [18] proposed a kind of intrusion response model based on attack graph, taking into account the factors such as system, attacker's profit and so on. Mehta and Sawilla [15, 19] considered different nodes in attack graph because of different location in attack path, such as some atomic attacks are a number of key points in the attack path, based on the Google Rank (web-level) to calculate the importance of each node. There may be a circular path in the attack graph, when the network security probability calculation is carried out, the repeated calculation of the cyclic node probability value will result in the error probability value which does not match the actual situation. Most literatures did not consider this situation.

Ou et al. [23] first proposed that one of the reasons for the complex attack pattern is a cyclic path problem in the attack graph, and it is found that the circular path in the graph can not be solved simply by deleting some atomic attacks, otherwise some important unconfirmed attacks. Wang [20] discussed the impact of three different types of circular paths on the risk assessment, and eliminated the loops by removing the succeeding nodes and edges of each node in loop path, the method is very complicat-ed to deal with the nodes in the loop path. At the same time, Wang does not give a detailed algorithm to calculate the probability of each node, nor does it consider the probability error calculation caused by the correlation between infiltration. Attack path analysis technology, takes forward search mode and depth-first search strategy to find the effective attack path of each node, through a collection of intermediate nodes to prevent the generation of a circular path, the algorithm's time complexity is exponential, nor does it apply to large-scale networks; Homer et al. [6] used an equivalent attack path without a circular path, so that a node in arbitrary path only appears one time. This method has added a lot of nodes and directed edges, the attack map will become very complicated. Based on the segmentation theorem in Bayesian networks, he also gave a way to solve the problem of correlation between osmosis. This method does not take into account the penetration of nodes. The results of the situation are only for small-scale network while the joint probability of access and calculation is also very complex. In attack graph, three uncertain sources are discussed: the uncertainty of the attack graph structure, the attacker's behavior and the intrusion alarm. The dynamic Bayesian network to capture the hidden uncertainty in the attack graph. Qi [14] used the Markov chain attack graph model to improve the accuracy of attack graphs. The degree of vulnerability attack is considered to determine the standard probability of state transition. The attack graph can statically evaluate the security of the network system, but it is difficult to dynamically deduce the attack intent and evaluate the next attack state based on the current system.

In this paper, these ideas of probability dependence to the Baum Welch algorithm [13] are used, and the probability generated by this method will be more realistic in representing the real network environment. Also, the use of HMM would be expanded, not only to establish the probability of mapping between the network observation and attack state, but also to calculate the use of HMM maximum probability state transition sequence. This method will be used to infer the attacker's attack intent more accurately.

3 Model establishment

3.1 Common vulnerability scoring system

Common Vulnerability Scoring System consists of three metrics: baseline score, time score, and environment score. The benchmark score metric represents the inherent and basic characteristics of the constant time and the vulnerability of user environment. The value of the time scale measure is the change in the value of vulnerability over time. The environmental score measure is fragile depending on the particular implementation environment.

We calculate the probability of each vulnerability from this base metric. To obtain the probability, the value of the base metric should be normalized by dividing its value with 1 while the base metric has maximum value of 1.

In other words a vulnerability can only be exploited provided that an attacker could gain access to other vulnerability. If an attacker succeeded in breaching one system vulnerability, then the probability of breaching the second system will increase while the attacker has experience to exploit vulnerability of a system with similar characteristic.

3.2 Hidden Markov Model

Basic theory The hidden Markov model is a model with a double stochastic process, where the first stochastic process is the Markov chain, which describes the state sequence. Another random process describes the relationship between the state and the observed variable. The state is not visible to the observer. And the state and its characteristics can only be observed by a random process, which reflects the relationship between the state and the observed variables. Implicit state S: Set up a set S = [S1,...,SS], where S is a model of a set of hidden states. Once the network system state is exploited for exploits, the S would be denoted this event. For example: = Exploit (Ha, Hw, Vi) that the host Ha through the loophole Vi on the host Hw attack, s is the number of state in the model. These states satisfy the Markov nature, which is the actual implied state in the Markov model. These states are usually not obtained by direct observation (E.g.,S1, S2, S3, etc).

Define observable state Y: Associated with the implicit state in the model can be obtained by direct observation (E.g.,Y1,Y2,Y3, ..., YT, etc), the number of observable states does not necessarily coincide with the number of implied states.

Define The initial state probability matrix π The initial state probability matrix π = [p1, p2, p3] is the initial state distribution, for example, t = 1, P (S1) = p1, P (S2) = p2, P (S3) = p3.

Define state transition probability matrix A that describes the transition probability between the states in the HMM model.

Define the state transition probability distribution matrix A: The observation set V indicates the exploit used by the attacker. For example, V1 is CAN-2003-0252.

Define observed state transition probability matrix B: Let N be the number of im-plicit states, and M be the number of observable states, then, B = {Mv /)}.

Define Oi: The probability of observing the state is Oi at the time t, the implied state is Sj.

Define tri-tupleλ: We use λ = (A, B, π) tri-tuple to concisely represent a hidden Markov model. The hidden Markov model is actually an extension of the standard Markov model, adding a set of observable states and the probabilistic relationship between these states and implicit states.

Define DVi: Indicating that the possibility of being attacked is the use of vulnera-bility Vi, it is clear that the greater the value, the greater the probability of occurrence, the attack will be less difficult. The system state is shifted from state S0 (normal state) to S1, and a new vulnerability has occurred. This process continues until the target state SS is achieved with the observation VS. Therefore, if Vi is successfully used, then the system state will turn to S.

Define DWi: Its weight of the system state will go through the loophole vi to the Si state. Hidden state setting S = S1, ..., SS. If the system state is transferred from state S to SS by exploiting the vulnerability, then the corresponding weight is IS. If the system state transitions from Vulnerability V1, V2, ..., Vs to another state SS, its corre-sponding weights are I1, I2, .., IS.

The state transition probability distribution matrix A formula is shown as follows:

Define the observed state transition probability matrix B. The detailed parameters are calculated as follow equation (1):

$$A=\{Aij\}=\left\{Wj/\sum_{t=1}^{n}Wt,\, Si\overset{Vj}{\rightarrow}Sj\right\} \tag{1}$$

When the system state is S, the attacker will attack the target successfully through the vulnerability. So we set the observations of these loopholes Vi to 1; When the system state is Si, the system state cannot be transferred from Si to Sj through Vulnerability V, then we put the probability to DVi accordingly; When the system state is Si, the system state can be transferred from Si to Sj through Vulnerability V, then we put the probability to DVi+DVi*DVj accordingly.

Finally, the data of the probability matrix was standardized.

Viterbi algorithm Viterbi algorithm is a dynamic programming algorithm. It is used to find the Viterbi path-implicit state sequence, which is most likely to produce the sequence of observed events, especially in the Markov information source context and the hidden Markov model. The Viterbi algorithm is a special but most widely used dynamic programming algorithm, which can solve the shortest path problem in any graph by using dynamic programming. And the Viterbi algorithm is proposed for the shortest path problem of a special graph - directed graph of the fence network. We want to find the hidden state sequences behind the observation sequence, and the hidden sequence of the largest probability of occurrence of the observation sequence, that is the result we need to find out.

Its algorithm is shown below:

The observation space is O, the state space is S, the observation sequence is Y, A is the transfer matrix, where Aij is the transition probability from the state Si to Sj, and the state transition probability matrix B is observed, where the state is observed in the state Si The probability of Sj, the initial probability of K, and the path X is the state sequence of the observed value Y. Output: Most likely implied state sequence X.

Algorithm 1 VITERBI(O, S, Y, π, A, B).

```
Input: input parameters A, B, C
Output: output result
for each state $S_i$ do
    $T_1[i,1] \Leftarrow \pi_i B_i Y_1$
    $T_2[i,1] \Leftarrow 0$
end for
for $i \Leftarrow 2, 3, ..., T$ do
    for each state $S_j$ do
        $T_1[j,i] \Leftarrow \max_k (T_1[k,i-1] * A_k j * B_j y_i)$
        $T_2[j,i] \Leftarrow \operatorname{argmax}_k (T_1[k,i-1] * A_k j * B_j y_i)$
    end for
end for
$z_j[i,1] \Leftarrow \operatorname{argmax}_k (T_1[k,T])$
$x_i[i,1] \Leftarrow S_{zT}$
for $i \Leftarrow T, T-1, ..., 2$ do
    $z_i - 1 \Leftarrow T_2[z_i, i]$
    $x_i - 1 \Leftarrow S_{z_i - 1}$
end for
return X
```

In the approach proposed by Jake, the introduction of CVSS and CCSS will represent a more realistic model. In order to calculate the vulnerability variables, their probabilities can be calculated using CVSS. For CVE-ID for CAN-2002-0392, the vulnerability has been confirmed and its identity becomes CVE-2002-0392: Apache packet encoding memory corruption vulnerability. The basic vector for this vulnerability is (AV: N / AC: L / Au: N / C: P / I: P / A: P). Through the above basic vector, the formula (2), (3), (4) were be calculated the following results:

$$Base=(0.6*Imp+0.4*Exp-1.5)*f(Imp),\ f(Imp)=1.176 \tag{2}$$

$$Imp=10.41*(1-(1-ConImp)*(1-IntImp)*(1-AvaImp)) \tag{3}$$

$$Exp=20*AccessComplexity*Authentication*AccessVector \tag{4}$$

Define three node types as vL for LEAF nodes, vA for AND nodes, and vO for OR nodes, then the probability of each node p(vL), p(vA), p(vO), in MulVAL attack graphs G can be derived using general theory of probability, as follow equations(5), (6), (7).

$$p(vL)=p(v)(forLEAFnodes) \tag{5}$$

$$p(vA)=p(v)\prod_{i=1}^{N}p(vI)(ConjuctiveprobabilityforANDnodes) \tag{6}$$

$$p(vO) = p(v)\prod_{i=1}^{N} p(vI)(Disjunctiveprobabilityfor ORnodes) \quad (7)$$

CAN-2002-0392:

A. Access Vector: Network (Score = 1),
B. Access Complexity: Low (Score = 0.71),
C. Authentication: None (Score = 0.704),
D. Confidentiality: Partial (Score = 0.275),
E. Integrity: Partial (Score = 0.275),
F. Availability: Partial (Score = 0.275).

Exp = 20 x AV x AC x Au = 20 x 1 x 0.71 x 0.704 = 9.9968

Imp = 10.41 * (1 - (1 - ConfImpact) * (1 - IntegImpact) * (1 - AvailImpact)) = 1.041 x (1- (1-0.275) x (1-0275) x (1-0275)) = 6.443

Base = (0.6Imp + (0.4Exp -1.5)) x f (Imp) = ((0.6 x6.443) + (0.4 x 9.9968) - 1.5) x 1.176 = 7.5

CVE-2009-3586:

A. Access Vector: Network (Score = 1),
B. Access Complexity: Low (Score = 0.71),
C. Authentication: None (Score = 0.704),
D. Confidentiality: Partial (Score = 0.275),
E. Integrity: Partial (Score = 0.275),
F. Availability: Partial (Score = 0.275).

Exp = 20 x AV x AC x Au = 20 x 1 x 0.71 x 0.704 = 9.9968

Imp =(1 - Availability) x (1 - Availability)) = 1.041 x (1- (1-0.275) x (1-0.275) x (1-0.275)) = 6.443

Base = (0.6Imp + (0.4Exp -1.5)) x f (Imp) = ((0.6 x 6.443) + (0.4 x 9.9968) - 1.5) x 1.176 = 7.5

CVE-2003-0252

A. Access Vector: Network (Score = 1),
B. Access Complexity: Low (Score = 0.71),
C. Authentication: None (Score = 0.704),
D. Confidentiality: Complete (Score = 0),
E. Integrity: Complete (Score = 0),
F. Availability: Complete (Score = 0).

Exp = 20 x AV x AC x Au = 20 x 1 x 0.71 x 0.704 = 9.9968

Imp =(1 - A) x (1 - A)) = 10.41 x (1- (1-0) x (1-0) x (1-0)) = 10.41 Base = (0.6Imp + (0.4Exp -1.5)) x f (Imp) = ((0.6 x 10.41) + (0.4 x 9.9968) - 1.5) x 1.176 = 10

CVE-2009-4776

A. Access Vector: Network (Score = 1),
B. Access Complexity: Medium (Score = 0.61),
C. Authentication: None (Score = 0.704),
D. Confidentiality: Complete (Score = 0),
E. Integrity: Complete (Score = 0),
F. Availability: Complete (Score = 0).

Exp = 20 x AV x AC x Au = 20 x 0.61 x 0.71 x 0.704 = 8.6

Imp = 10.41 * (1 - (1 - ConfImpact) * (1 - IntegImpact) * (1 - AvailImpact)) = 10.41 x (1- (1-0) x (1-0) x (1-0)) = 10.41

Base = (0.6Imp + (0.4Exp -1.5)) x f (Imp) = ((0.6 x 10.41) + (0.4 x 8.6) – 1.5) x 1.176 = 9.3

Since MulVAL's attack graph shows that the probability of all LEAF nodes or con-figuration nodes is 1.0, this means that each variable in the LEAF node is assumed to exist and manipulated as an attacking medium. This is not a real case, so in this article the second method of Jake's approach was implemented. Because the display network state does not exist can take advantage of vulnerability variables. If there is a loophole in a node, the probability of other nodes will be higher loopholes. This means that the vulnerability of node N1 depends on the vulnerability at node N3, and the probability vulnerability at node N1 may increase or exceed the original possibility. Node N3 has identity CAN-2002-0392, node N1 has identity CAN-2003-0252. If these two vulnerabilities can be used remotely, access rights state was changed. Thus, the result was came out:

N3 vul=P(CAN-2002-0392)=0.75

Node N3: vulExists (webServer, 'CAN-2002-0392', httpd, remoteExploit, priv Escalation): 0.75

N1 vul=P(CAN-2003-0252)*P(CVE-2009-4436)=0.85

Node N1: vulExists (fileserver, 'CAN-2003-0252', mountd, sqlInject, priv Escalation): 0.85

N2 vul=P(CVE-2009-3586)*P(CVE-2009-4251)=0.8

Node N2: vulExists (webServer, 'CVE-2009-3586', httpd, remoteExploit, priv Escalation): 0.8

N4 vul=P(CVE-2009-4776)*P(CVE-2007-6432)=0.7

Node N4: vulExists (webServer, 'CVE-2009-4776', httpd, bufferOver, priv Escalation): 0.7

4 Experiment And Analysis

4.1 Experimental environment

In a given network topology environment, the proposed model of the experiment is used to verify its feasibility. The topology shown in Fig. 1.

In the network topology, there are three regions (internet, dmz, internal). The Internet is considered a threat from an external network, a potential attacker; the middle area is a DMZ (non-military area), a web server (Web Server) placed in the DMZ and the external network Firewall is fw1; internal (internal), placed a file server (File Server) and a workstation (Work Station), a firewall placed between the network and DMZ. External accesses to the web server through the internet, and cannot directly access the workstations within the network.

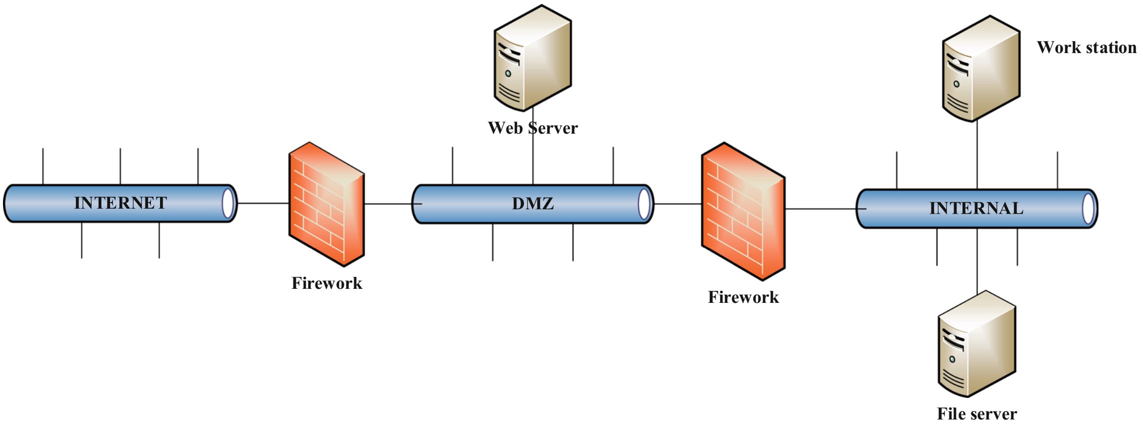

Fig. 1 network topology graph

4.2 Simulation attack flow graph and vulnerability information

The simulation values and descriptions used in our simulation experiments are shown in Table 1.

The weight of each vulnerability would be computed and generated, and the im-proved MulVAL evaluation score which we purposed used as the weight of the new vulnerability. Its values are shown in Table 2.

4.3 HMM and Baum Welch algorithm

According to the network topology and network configuration, the attacker wants to attack the target with vulnerabilities. Therefore the intention of the attacker will be extracted.

S is the system state space, S shows the state of the system in the attacker to make the attack process, the system state of the process of change. Where T0 is the initial state, indicating that the attacker is ready to attack, this time the system is not at-tacked. S1 indicates that the attacker has compromised the web server through the V1 vulnerability. S2 indicates that the attacker had a buffer overflow attack on the web server through the V2 vulnerability. S3 indicates that the attacker by taking the web server and then attack the file server, using V3 vulnerabilities. S4 indicates that the attacker utilize the file server and then attack the workstation with V4 vulnerabilities.

We will use a forward-backward algorithm, also known as the Baum Welch algorithm. The first case is relatively simple, that is, we know D observation sequences of length T and corresponding hidden state sequences, namely (O1, I1), (O2, I2), ... (OD, ID) are known, and we can easily solve the model parameters with maximum likelihood. Assuming that the frequency of the sample moving from the hidden state qi to qj is Aij, then the state transition matrix is evaluated as Eq. 8:

$$A = \left[a_{ij}\right], where a_{ij} = \frac{A_{ij}}{\sum_{s=1}^{N} A_{is}} \tag{8}$$

It can be seen that solving the model in the first case is still very simple. But in many cases, we can't get the hidden sequence corresponding to the HMM sample observation sequence. Only D observation sequences of length T are (O1),(O2),...(OD) is known. The most commonly used solution is the Baum Welch algorithm, which is actually based on the EM algorithm.

Since the Baum Welch algorithm principle uses the principle of the EM algorithm, the joint distribution P$(I \mid O, \lambda)$ is found in the E step based on the conditional probability P$\left(I \mid O, \overline{\lambda}\right)$, where For the current model parameters, then M steps to maximize this expectation, resulting in an updated model parameter. The EM iteration is then continued until the values of the model parameters converge.

Table 1 Vulnerability and descriptions

Host	Node	Vulnerability
Web server	V1	CAN-2003-0252
Web server	V2	CVE-2009-3586
File server	V3	CAN-2002-0392
Work station	V4	CVE-2009-4776

Table 2 Vulnerability and weight

Vulnerability	Improved MulVAL assessment score	Weight
V1	0.85	0.85
V2	0.8	0.8
V3	0.75	0.75
V4	0.7	0.7

First, let's look at the E step. The current model parameter is The joint expression P$(I \mid O, \lambda)$ is based on the conditional probability P$\left(I \mid O, \overline{\lambda}\right)$. The expected equation is Eq. 9:

$$L\left(\lambda, \overline{\lambda}\right) = \sum_{I} P\left(I \mid O, \overline{\lambda}\right) logP\left(I \mid O, \overline{\lambda}\right) \tag{9}$$

In step M, we maximize the above formula and then get the updated model parame ters as follow Eq. 10:

$$L\left(\lambda, \overline{\lambda}\right) = argmax \sum_{I} P\left(I \mid O, \overline{\lambda}\right) logP\left(I \mid O, \overline{\lambda}\right) \tag{10}$$

Through the continuous iteration of E step and M step, we show the derivation of the Baum Welch algorithm. We can deduce the iterative formula for A, this part of the formula Eq. 11 can be organized as:

$$\sum_{d=1}^{D}\sum_{I}\sum_{t=1}^{T-1} P\left(I \mid O, \overline{\lambda}\right) a_{it}a_{it+1} = \sum_{d=1}^{D}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{t=1}^{T-1} P\left(O, i_t^d = i, i_{t+1}^d = j \mid \overline{\lambda}\right) loga_{ij} \tag{11}$$

According to the method mentioned in above equations, the HMM parameters were calculated as follows:

$$A = \begin{bmatrix} 0.42 & 0.58 & 0 & 0 \\ 0.47 & 0.53 & 0 & 0 \\ 0 & 0 & 0.33 & 0.67 \\ 0 & 0 & 0.33 & 0.67 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.28 & 0.24 & 0.32 & 0.32 \\ 0.24 & 0.28 & 0.32 & 0.30 \\ 0.24 & 0.28 & 0.22 & 0.22 \\ 0.24 & 0.24 & 0.14 & 0.16 \end{bmatrix}$$

As shown in the HMM state transition diagram, S0 can be directly converted to SI, S2 or S3. We use python to implement the Viterbi algorithm, enter the observation sequence VI, V2, V3, V4 and model parameters = (A, B, π). The results are as follows:

A. If = (1,0,0,0), the optimal state sequence is S1, S2, S3, S4, the probability is 0.2255;
B. If = (0,1,0,0), the optimal state sequence is S2, S4, the probability is 0.2876;
C. If = (0,0,1,0), the optimal state sequence is S3, S4, the probability is 0.2886;

Summarizing all the results to arrive at the most likely sequence, we can see the system state transition sequence is S3, S4. Therefore, the most likely path is to crack the file server FS through exploit V3 and V4.

Liu considers the difficulty of calculating the vulnerability as a probability of determining the state transition, using the state transition probability directly as evidence of the decision of the network security administrator. The state transition probability matrix is calculated using the Liu method [11].

The state transition probability matrix is as follows:

$$C = \begin{bmatrix} 0 & 0.34 & 0 & 0.24 & 0.42 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.36 & 0.64 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

In contrast, the actual network situation must be considered and the CVSS assessment scores were adjusted. To use the HMM's maximum probability state transition sequence analysis problem, network security administrators view the result as evidence for making decisions will become more reliable.

The algorithm includes using the HMM model to calculate the probability that the host is attacked and determining the value of network security risks based on the importance of the network assets. The HMM model parameters are determined by a simple CVSS scoring system. In contrast, our approach more concerned about the state transition and set up parameters with state transition in the model.

5 Conclusion

The generation of attack graphs is part of the network risk assessment, and the model of the attack network makes network assessment more accurate. From the perspective of network experts and network administrators, the implementation of this model allows them to take more effective measures with changes between networks and threats from external networks. Based on the MulVAL framework, this paper uses the Viterbi algorithm and Baum Welch algorithm to deduce the most probable state transition sequence, which is the path of the most likely attack through simulation experiments. Also, this paper uses the combination of Baum Welch algorithm and Markov Model to make a more accurate prediction of the entire network and risk assessment.

This approach is not easy to deploy in the super large scale network environment, the future work is researching about how to work effectively with these two models deployed in a larger network environment, or in the real business network.

Acknowledgements Our work was supported by The United Foundation of General Technology and Fundamental Research (No.U1536122), the General Project of Tianjin Municipal Science and Technology Commission under Grant (No.15JCYBJC15600), the

Major Project of Tianjin Municipal Science and Technology Commission under Grant (No.15ZXDSGX00030). The authors would like to give thanks to all the pioneers in this field, and also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the quality of this paper.

References

1. Chen Z, Peng L, Gao C, Yang B, Chen Y, Li J (2017) Flexible neural trees based early stage identification for ip traffic. Soft Comput 21(8):2035–2046
2. Fan L, Lei X, Yang N, Duong TQ, Karagiannidis GK (2016) Secure multiple amplify-and-forward relaying with cochannel interference. IEEE J Selected Topics Signal Process 10:1494–1505
3. Fan L, Lei X, Yang N, Duong TQ, Karagiannidis GK (2017) Secrecy cooperative networks with outdated relay selection over correlated fading channels. IEEE Trans Veh Technol 66:7599–7603
4. Feng P, Lian Y, Dai Y (2006) A vulnerability model of distributed systems based on reliability theory. J Softw 17(7):1633–1640
5. Gao Y, Shan X, Hu Z, Wang D, Li Y, Tian X (2016) Extended compressed tracking via random projection based on msers and online ls-svm learning. Pattern Recogn 59(11):245–254
6. J H, XM O, D S (2009) A sound and practical approach to quantifying security risk in enterprise networks. Technical Report 2009-3 Kansas State University
7. Jiang W, Wang G, Bhuiyan MZA, Wu J (2016) Understanding graph-based trust evaluation in online social networks: methodologies and challenges. ACM Comput Surv 49:10:1–10:35
8. Lai X, Zou W, Xie D, Li X, Fan L (2017) Df relaying networks with randomly distributed interferers. IEEE Access 5:18:909–18:917
9. Li B, Huang Y, Liu Z, Li J, Tian Z, Yiu SM (2018) Hybridoram: practical oblivious cloud storage with constant bandwidth. Information Sciences. https://doi.org/10.1016/j.ins.2018.02.019
10. Li Y, Wang G, Nie L, Wang Q (2014) Collaborative target tracking in wireless sensor networks. J Ad hoc Sensor Wireless Netw 23:117–135
11. Liu Q, Wang G, Li F, Yang S, Wu J (2017) Preserving privacy with probabilistic indistinguishability in weighted social networks. IEEE Trans Parallel Distrib Syst 28(5):1417–1429
12. Wang L, Singhal A, Jajodia S (2007) Toward measuring network security using attack graphs. In: Proceedings of the 3rd international workshop on quality of protection(QoP), pp 49–54
13. Qi Y, Liu M, Li Q (2011) Baum-welch algorithm. In: 2011Baum
14. Qi Y, LIU M, LI Q (2014) An attack graph model based on extended Markov chain. Comput Eng Des 35(12):4131–4135
15. Sawilla RE, Ou X (2008) Identifying critical attack assets in dependency attack graphs, pp 18–34
16. Shen H, Gao C, He D, Wu L (2015) New biometrics-based authentication scheme for multi-server environment in critical systems. J Ambient Intell Human Comput 6(6):825–834
17. Shen J, Zhou T, Wei F, Sun X, Xiang Y (2017) Privacy-preserving and lightweight key agreement protocol for v2g in the social internet of things. IEEE Internet of Things Journal. https://doi.org/10.1109/JIOT.2017.2775248
18. Shi J, Guo S (2008) An intrusion response method based on attack graph. J Softw 19(10):2746–2753
19. Mehta V, Bartzis C, Zhu H (2006) Ranking attack graphs. In: Proceedings of the 9th international symposium on recent advances in intrusion detection(RAID), pp 127–144
20. Wang L, Islam T (2018) An attack graph—based probabilistic security metric. In: Proceedings of the 22nd annual IFIP WG 113 working conference on data and applications security(DBSec), pp 1–1
21. Wang X, Zhang H, Fan L, Li Y (2017) Performance of distributed switch-and-stay combining for cognitive relay networks with primary transceiver. Wireless Personal Commun 97:3031–3042
22. Xing X, Xie D, Wang G (2015) Energy-balanced data gathering and aggregating in wsns: A compressed sensing scheme. Int J Distrib Sensor Netw 2015:1–10
23. XM O, WF B (2006) A scalable approach to attack graph generation. In: Proceedings of the 13th ACM conference on computer and communications security(CCS), pp 336–345
24. Yang W, Wang G, Bhuiyan MZA, Choo KKR (2017) Hypergraph partitioning for social networks based on information entropy modularity. J Netw Comput Appl 86:59–71