Maximization of Network Robustness Considering the Effect of Escalation and Accumulated Experience of Intelligent Attackers

Frank Yeong-Sung Lin, Po-Hao Tsang, Pei-Yu Chen, Huan-Ting Chen
Department of Information Management
National Taiwan University
Taipei, Taiwan, R.O.C.
{ yslin, d91002,d96006,r95012}@im.ntu.edu.tw

Abstract

In this paper, we propose a mathematical programming model to describe a network attack scenario. In this problem, the attacker's objective is to compromise multiple core nodes at minimum total attack cost. During the attack actions, the attacker may gain some experience from previous attacks to further reduce the attack costs in the future. Moreover, he can also pay extra fee to escalate his authority on a compromised node to get higher user privileges, so that he will have higher authority to access more information on the node. We measure the impact incurred by information leakage in our model, and adopt some Simulated Annealing-based algorithms to solve the problem.

1. Introduction

Internet provides us with many convenient services, such as video conference, file transfer, and E-commerce applications. However, it also brings us some threats to information security and privacy, because cyber criminals can connect to others' computers to steal information or modify some important data via Internet.

Since the Internet has become essential to business operations, numbers of trades are accomplished through computer network. Some critical data are usually stored in computers or other multimedia devices. Therefore, protecting these data is an important issue for enterprises. A remote backup can ensure that systems can provide uninterrupted services or quickly recover from a disaster or a malicious attack. In this manner, critical data is copied and stored in different centers, all of which can provide the main service independently. Consequently, an attacker who wants to completely crash business service or networks may need to compromise all of these data centers which have the capability of providing essential services.

Furthermore, in an attack scenario, after compromising a node, the attacker can choose whether to probe this node and access valuable information or not. For instance, the attacker may gain some useful information like the routing tables which can help him to get the whole picture of the network topology. Another example is that the attacker may access some vital information like the customer data of an E-commerce company. When this kind of information is stolen, it not only causes some privacy issues but also leads to financial loss.

To access more information, the attacker may need to escalate to gain more user rights. In [1], the author analyzed the attack behavior and found some characters: an attacker in the low user-level may usually exploit several vulnerabilities on a computer system to get a certain privilege escalation. They indicated that the attacker at a certain user- level owned the corresponding user privileges and resources of that system. Thus, how to manage the vulnerability well is important for protecting a node from being attack. According to [2], the author measured the total effect of vulnerabilities in a system. Therefore, managing vulnerabilities well would be an important mission in computer security.

When an attacker compromises a node, he may gain experience and escalated authorities. Using the experience efficiently he can reduce the costs of future attacks. For example, when the attacker compromises a node, he may learn how to intrude other systems via the same kind of vulnerabilities on the compromised node. In [3], the author conducted intrusion experiment for empirical data. By analyzing the collected data, the author split the intrusion process into three phases based on attacker behavior: the learning phase, the standard attack phase, and the innovation phase. McDermott et al. [4] pointed out that potential intelligent intruders will more probably attack the target as time goes by. Therefore, the intruder cannot compromise the target today may be more likely to compromise the target in the future.

Because the budget is finite, it is important for a network operator/defender to allocate his budget efficiently. However, there are seldom theoretical studies modeling the attacker behavior and the offense-defense scenarios in mathematical ways [5]. Therefore, we propose a mathematical model. It describes and formulates the effect of Accumulated Experience and Escalation of attackers (AEE) in a quantitative way.

In the AEE Model, the attacker's objective is to compromise multiple core nodes and minimize the total attack cost, which includes the cost of compromising nodes and escalating his authorities on each compromised node.

The remainder of this paper is organized as follow. In Section 2, a mathematical formulation of the attack scenario is proposed. In Section 3, solution approaches

based on the Simulated Annealing methods are presented. The computational results of the experiments are showed in Section 4. Finally, Section 5 is the conclusions and future work about this research.

2. Problem Formulation

In this section, we describe the problem and propose a mathematical model with specific assumptions and problem objective to the target network. The attacker's objective is to compromise multiple core nodes in the given network and to minimize the total attack cost as possibly as he could. In addition, he may gain experience from his previous attacks to reduce the costs of the future attacks.

An attacker can gain two kinds of experience during an attack, one comes from compromising a node, and the other comes from escalating his authority on the compromised node. The first kind is gained from previous attacks, and is used to reduce the costs of future attacks. The second kind of experience is gained from the escalation on the compromised node. After compromising a node, the attacker may pay an extra fee to conduct some authority escalation on the node to get more powerful user rights, which allow him to access more useful information to further reduce the costs of attacks and accumulate impact incurred by information leakage on the compromised nodes. An attacker could pay various extra fees in order to have different levels of escalation, because there are several levels of user privileges in a system. The user privileges increase as the attacker pays more of his budget for them.

The information derived from a compromised node may include important financial data of an enterprise or secret files, such as personnel data, or the password of a network administrator. This may cause critical loss of the network and serious damage to the enterprise. For this reason, our model also considers the information value corresponding to an impact factor to evaluate the damage incurred by information leakage.

We assume that the target network is at the Autonomous System (AS) level network; hence, an attacker needs to compromise the core node step-by-step. Because the number of vulnerabilities on each node is different, an attacker who wants to compromise a node may need to pay different costs related to the defense budget allocated to the node and the vulnerabilities on it. We also assume that there are several levels of user privileges on a system. Thus, an attacker could pay various levels of extra budget to do different levels of escalation. The more costs he pays, the more user rights he could gain. We also use an impact factor to evaluate the information an attacker access from a compromised node.

2.1 Problem Formulation of the AEE Model

We model the above problem as a mathematical programming problem. The given parameters are defined as Table 1.

Table 1 Given Parameters

Notation	Description
N	The index set of all nodes in the network
D	The index set of all core nodes in the network
W	The index set of all Origin-Destination pairs (O-D pairs), where the origin is node o; and the core nodes are d (where $d \in D$)
E_i	The index set of all the privilege levels on node i (e.g., 0, 1, 2, …), where $i \in N$ and level 0 means node i is compromised without escalation.
L_i	The index set of all the level on node i exclusive of level 0, where $i \in N$
P_w	The index set of all candidate paths of an O-D pair w, where $w \in W$
δ_{pijl}	An indicator function, which is 1 if node j (at privilege level l) is the pervious node of node i on path p, and 0 otherwise (where $i, j \in N, p \in P_w, l \in E_i$)
σ_{pil}	An indicator function, which is 1 if of node i (at privilege level l) is on path p, and 0 otherwise (where $p \in P_w, i \in N, l \in E_i$)
S	The index set of all stages
$S_{(k)}$	The index set of the stage 1 to stage k-1, where $k \in S$
e_{il}^{e}	The experience gained by the attacker after escalating to level l on node i, where $i \in N, l \in E_i$
e_{i}^{c}	The experience gained by the attacker after compromising node i, where $i \in N$
I_{il}	The impact incurred by accessing information from level l on node i after escalation, where $i \in N, l \in E_i$
T	The threshold of total impact, which is the damage level that the attacker needs to reach.
B	The total defense budget

In this formulation, the attack sequence is represented by a term, *stage*. Stage n means the attack is launched on the n-th step of the attack action. As noted earlier, once the attacker escalates to a higher level on a compromised node, he might know some links he did not know before escalating to the level. Thus, the network we modeled here can be viewed as an artificial two-dimensional network. The decision variables are defined as Table 2.

Table 2 Decision Variables

Notation	Description
y_{sil}	1 if node i is compromised at stage s and escalated to level l of the node, and 0 otherwise (where $s \in S, i \in N, l \in E_i$)
x_p	1 if path p is selected as the attack path, and 0 otherwise (where $p \in P_w$)
b_i^c	The defense budget allocated to protect node i from being compromised, where $i \in N$
b_{il}^e	The defense budget allocated to protect node i from being escalated, where $i \in N, l \in L_i$

$\hat{a}_i^c(b_i^c)$	The threshold of the attack budget required to compromise node *i*, where $i \in N$
$\hat{a}_{il}^e(b_{il}^e)$	The threshold of the attack budget required to escalate to level *l* on node *i*, where $i \in N$, $l \in L_i$

The objective is to minimize attack cost by adjusting which nodes and levels to attack and which attack sequence to adopt. In this problem, an attacker tries to compromise multiple core nodes using the minimized total attack cost. Thus, we formulate attacker behavior as an optimization problem, the AEE Model.

Objective function:

$$\min_{y_{sil},\, x_p} \sum_{i\in N, l\in E_i} \hat{a}_i^c(b_i^c) \sum_{k\in S} y_{kil} \prod_{s\in S_{(k)}} \sum_{j\in N} \sum_{m\in E_i} \left(e_j^c y_{sjm}\right) + \sum_{i\in N, l\in L_i} \hat{a}_{il}^e(b_{il}^e) \sum_{k\in S} y_{kil} \prod_{s\in S_{(k)}} \sum_{j\in N} \sum_{m\in L_i} \left(e_{jl}^e y_{sjm}\right), \quad \text{(IP 1)}$$

subject to:

$$\sum_{p\in p_w} x_p \sigma_{pil} \le \sum_{s\in S} y_{sil} \qquad \forall i\in N, w\in W, l\in E_i \quad \text{(IP 1.1)}$$

$$\sum_{p\in P_w} x_p = 1 \qquad \forall\ w \in W \quad \text{(IP 1.2)}$$

$$x_p = 0 \text{ or } 1 \qquad \forall\, p\in P_w, w\in W \quad \text{(IP 1.3)}$$

$$\sum_{l\in E_i} y_{kil} \le \sum_{s\in S_{(k)}} \sum_{j\in N} \sum_{l\in E_i} y_{sjl}\delta_{pijl} \qquad \forall\, i\in N, k\in S,\ p\in P_w, w\in W \quad \text{(IP 1.4)}$$

$$\sum_{s\in S} \sum_{l\in E_i} y_{sil} \le 1 \qquad \forall\, i\in N \quad \text{(IP 1.5)}$$

$$\sum_{i\in N} \sum_{l\in E_i} y_{sil} \le 1 \qquad \forall\, s\in S \quad \text{(IP 1.6)}$$

$$y_{sil} = 0 \text{ or } 1 \qquad \forall\, i\in N, s\in S, l\in E_i \quad \text{(IP 1.7)}$$

$$T \le \sum_{i\in N} \sum_{l\in E_i} \sum_{s\in S} I_{il} y_{sil} \quad \text{(IP 1.8)}$$

In this model, the attacker's objective is to compromise multiple core nodes using the minimized total attack cost. The compromised costs and escalation costs would be reduced by experience factor, e_i^c and e_{il}^e, which are values between 0 and 1. The effect of the experience would be showed as accumulated multiplied forms. Constraint (IP 1.4) requires that if a node is compromised at stage *k*, the ancestor node of that node on the selected attack path must have been compromised at one of the stages 1 to *k*-1 before.

3. Solution Approach

3.1 Solution Approach for the AEE Model

In this section, the AEE Model [6] is solved by Simulated Annealing (SA) based heuristics. We develop two-phase SA-based approaches to solve the problem. The first phase involves implementing the SA procedure and adjusting the attack sequence, attack tree and escalation level. During the first phase, the 10 best solutions are saved as the first step results. Next, in the second phase, we use the results from the first phase as initial solutions, and search for neighbor solutions by adjusting the attack sequence only. Finally, the smallest objective value is saved and which is the best solution we found by this approach.

We use three different initial solutions and several methods to search for the neighbor solutions based on the approach we developed in last paragraph. The first initial solution is an algorithm that is similar to Prim's algorithm, which first generates a minimum cost spanning tree. Next, we prune unnecessary nodes, i.e., nodes that are not core nodes or intermediate nodes on the paths towards core nodes. Finally, the escalation levels are adjusted to satisfy the corresponding constraints. The second solution is a random-based algorithm. The difference between this solution and the first one is the criteria for choosing the next node. In this case, when choosing the next target node, we always choose a reachable node at random as the next attack node, instead of choosing the node with the smallest weight. The third solution is similar to the first one; the weight of each node is the ratio of the cost to experience. The time complexity of all the initial solutions is $O(|N|log|N|)$.

The approaches for searching for neighbor solutions can be divided into three parts: change of the attack sequence, change of the attack tree, and change of the escalation levels of compromised nodes. We use two different techniques to change the attack sequence. The first rearranges the whole traversed sequence. We start from the source node and randomly choose a compromised node that can be reached from the source node and re-label its sequence. Then, we repeat this process until all the compromised nodes are visited again. The second technique randomly exchanges the attack sequences of two compromised nodes. We also divide methods of changing the attack tree into two parts. The first a compromised node is randomly chosen on the attack tree by the attacker and the compromised nodes, which were compromised after the choose node, are reset to uncompromised. Then, we start from the chosen node to find other paths randomly in order to complete the attack. Therefore, the new attack tree will be the same with the original tree before the node was chosen. The second method only adjusts small parts of the original attack tree. We use two methods for this task. One changes the path between two compromised nodes that are adjacent to each other in the attack tree and adjust the attack sequence if necessary. The other compromises an additional node that is not necessary for the original attack tree or it removes an unnecessary node from the original attack tree. This is reasonable because the attacker may gain some additional experience from the extra attack and that experience may help him reduce the cost of future attacks. Finally, we also develop two ways to change the escalation levels. One randomly exchanges the escalation levels on two compromised nodes. The other escalates to a higher level or drops to a lower level on a randomly chosen compromised node. Although we

develop several methods to search neighbor solutions, we only choose one approach at random to search for neighbor solutions in each loop. The time complexity of searching for neighbor solutions is $O(|N|log|N|)$.

4. Computation Experiments

4.1 Computation Experiments with the AEE Model

To measure the effective of our proposed algorithms, we design the simple algorithm 1 *S1*. It can be divided into an outer part and an inner part. At outer loop, we ignore the effect of experience and then run the Prim's Algorithm and calculate the total cost. Next, starting from the source and replacing the weight of each node with the value that the original nodal cost subtracts the effect of its experience. We start from the source and replace the weight of each node with the value that the original nodal cost subtracts the effect of its experience. The effect would be calculated by the experience factor of the current target node multiplying the total cost of all the nodes which are compromised after the current node on the attack tree. Then, we run the SA procedure which is the inner loop of this heuristic to adjust the sequence and escalation levels.

We use different initial solutions to distinguish our approaches. The first one is the Prim-based algorithm, and its corresponding SA approach is denoted as *TSA_Prim*. The second one is the approach which randomly chooses the next node. Its corresponding SA solution is denoted as *TSA_Random*. The last one is the solution using the ratio between the experience and the cost of a node. Its corresponding approach is denoted as *TSA_Weight*.

4.2 Experiment Environment

The SA parameter α is set to 0.7, and β is set to 1.3. The initial temperature T_0 is initialized to 1.0 and the final temperature is set to T_0/1000. At each temperature, we control the SA to repeat b_0 times, and initialize b_0 to 1000. We randomly assign the experience value and the number of vulnerabilities on each node.

To evaluate the quality of our approaches, we compare our solutions to the exhaustive search in three small networks with 9 nodes which are grid, random and scale-free networks. We consider one escalation level in the three small networks. In other larger size networks, we use two ways to evaluate the quality of our solutions. One is to compare our solutions with *S1*. The other is to design two networks as showed in figures 1 and 2, in which we can find the optimal solutions intuitively. In Figure 1, we set the second type and the third type nodes and the last node of the first type nodes as core nodes.

In order to evaluate the robustness of networks, we also consider three types of networks (i.e., grid, random, and scale-free networks). Each network could consist of 25, 49, 81, 100, or 144 nodes, each of which could have three escalation levels.

The cost function here is defined as a concave form, $\ln(\frac{b_i \times M}{V_i}+1)$, where b_i is the budget allocated to node i and V_i, a given parameter here, is the number of the vulnerabilities on node i and M is a constant to adjust the proportion of b_i and V_i. The cost functions of different escalation levels on nodes are also defined as this form.

We also design three budget allocation strategies. The first policy is a uniform allocation strategy. Each node is allocated the same defense budget. The second one is a degree-based budget allocation. Each node is allocated budget according to the percentage of its degree over the total degree of the network. The last one is the vulnerability-based budget allocation. Budget allocated to each node depends on the ratio of the vulnerabilities on each node over total vulnerabilities in the networks. As noted earlier, the network can be viewed as a two-dimensional network. Thus, while allocating defense budget to escalation levels, we can treat the different levels on a node as different nodes in this artificial two-dimensional network. Consequently, we can use this property to allocate budget to each level in degree-based and uniform defense budget allocation strategies.

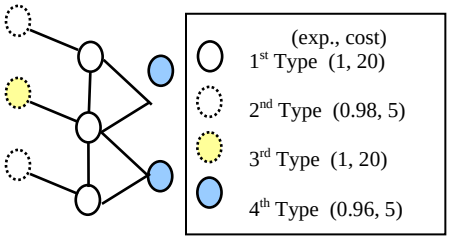

Fig. 1 Experiment Topology 1

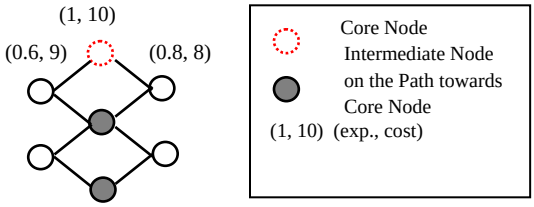

Fig. 2 Experiment Topology 2

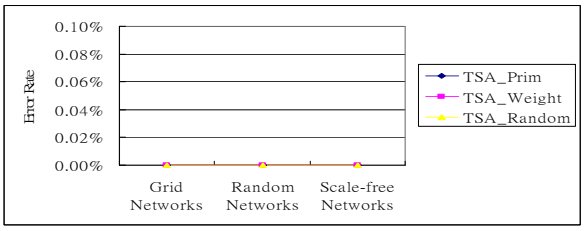

Fig. 3 The Error Rate of Networks with 9 nodes

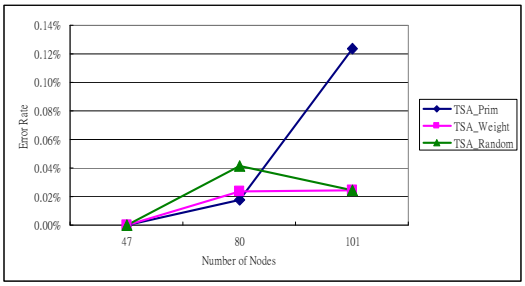

Fig. 4 The Error Rate of Proposed Approaches under Experimental Networks 1

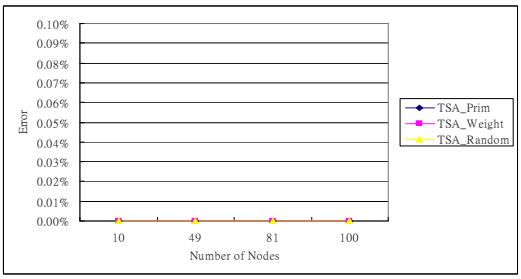

Fig. 5 The Error Rate of Proposed Approaches under Experimental Networks 2

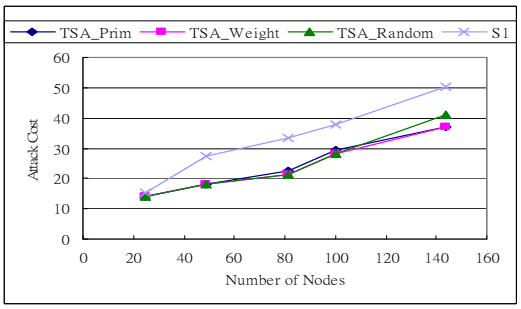

Fig. 6 Comparison of Proposed Approaches under Scale-free Networks with Degree-based Defense Budget Allocation Strategy

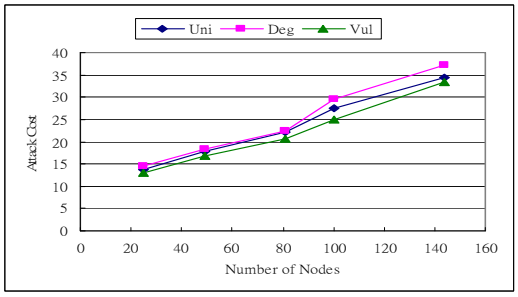

Fig. 7 Comparison of Random Networks with Different Defense Budget Allocation Strategies

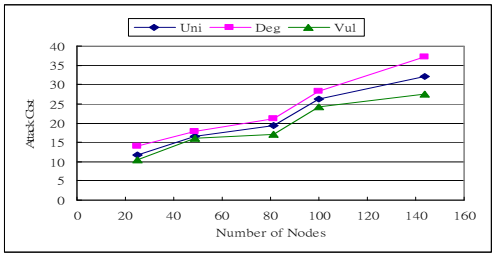

Fig. 8 Comparison of Scale-free Networks with Different Defense Budget Allocation Strategies

4.3 Discussion of Results

Figures 3 to 5 show the quality of our solutions under the target networks. The error rate of our solutions under these experiment networks and the networks with 9 nodes is approximate under 0.1%.

Figure 6 compares the quality of the proposed algorithms with *S1* in scale-free networks. Our heuristics perform better than *S1* obviously. The quality of the results of the approach *TSA_Weight* is better than other approaches on average. Thus, we use *TSA_Weight* as the solution approach in the following comparisons.

For Figures 7 and 8, we could observe that networks with degree-based defense budget allocation strategy are the most robust. If a node with more connectivity, it may be also a shortcut in a network. Thus, the attacker could use this node to reach his targets more quickly. Hence, if the defender protects these nodes more, it would become more difficult for the attacker to reach the target nodes. The vulnerability-based defense budget allocation is the most vulnerable way to protect all the networks. The reason is that if a node is vulnerable, it may be allocated much budget in this allocation strategy. But if the node is on the edge of the network and the attacker could also reach his goal without compromising it.

5. Conclusion

We have addressed the issues of attacker behavior under network defense-attack scenario. We focus on the learning skill of intelligent attackers and how it could help the attackers to reduce their costs in the future. This concept is generalized as a term, *experience*, in this paper. We also modeled the escalation of attackers and evaluated the impact incurred by information leakage. As a result, the attacker would try to minimize the total attack cost under these issues. In response, the network defender would try to maximize the total attack cost by a proper defense budget allocation strategy.

The key contribution of this paper is the development of a max-min mathematical model which well formulated the interaction between attackers and defenders in the real world. We have also solved this model by several proposed heuristics. To the best of our knowledge, very little research is done to model the real-world attack behavior in the offense-defense sceneries by this approach.

Another contribution is we have evaluated the robustness of different networks with different budget allocation strategies by the minimized total attack cost.

In this paper we assumed that information probed from each level on a compromised node would not be duplicated. Thus, the experience and the impact of information would be accumulated continuously. By this assumption, an attacker is very skillful and intelligent that he would not pay any unless fee to gain duplicated information. Therefore, we could further discuss the duplicated information issues in the future.

Moreover, in this paper, we only discuss the behavior of attackers. Considering the attack and defend scenarios, when the attacker decides his attack strategy, the network defender readjusts his resource allocation strategy to resist the attacks. In response, the attacker will change his strategy again to find the most cost efficient approach. There may be some serial interactions between the defender and the attacker. Therefore, we will focus on how to appropriately allocates the limited defense resources of operator.

References

[1] Z. Yongzheng and Y. Xiaochun, "A New Vulnerability Taxonomy Based on Privilege Escalation," *Proceedings of the 6th International Conference on Enterprise Information Systems*, 2004.

[2] O.M. Alhazmi, Y.K. Malaiya, "Quantitative Vulnerability Assessment of Systems Software" *Proceeding of IEEE Reliability and Maintainability Symposium*, pp. 615-620, Jan 2005.

[3] E.Jonsson and T. Olovsson, "A Quantitative Model of the Security Intrusion Process Based on Attacker Behavior", *IEEE Transactions of Software Engineering*, Volume 23, Number 4, pp. 235-245, April 1997.

[4] J. McDermott, "Attack-Potential-Based Survivability Modeling for High-Consequence System," *Proceedings of the Third IEEE International Workshop on Information Assurance (IWIA'05)*.

[5] M.N. Azaiez, V.M. Bier, "Optimal Resource Allocation for Security in Reliability systems" *European Journal of Operational Research*, 181 pp. 773-786, 2007.

[6] S. Kirkpatrick, C.D. Gelatt, Jr., M.P. Vecchi, "Optimization by Simulated Annealing", *Science*, Volume 220, Number 4598, pp. 671-680, May 1983.