Dynamic Trilateral Game Model for Attack Graph Security Game

Chenao Hu¹ and Xuefeng Yan²,*

¹,²College of Computer Sciences and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China

*Corresponding author

Abstract. Internal threats have a huge impact on the attack graph security game. The failure of the MTD model defence measures would be caused by the existence of internal users with certain authority. Dynamic trilateral game model was proposed to extend the original two-part game model. By materializing internal threats, the uncertainty of two-part game model was eliminated, which was expressed as the probability equation used by the players in the observation state process. And the relationship between the offensive and defensive sides became indirect. The user strategy, based on mixed strategy game model, was proposed to increase the coupling between stealth attack and internal threats. The income matrix was dynamically constructed to measure the behavioural outcomes of users and attackers. User behavioural references were obtained through dynamic programming. For the defender, the heuristic strategy in the model reduces the complexity of parties' behaviour through random sampling. Experiments were carried out on the attack graph model under various game settings. Compared with the two-part game model, our model's experimental results showed that the cyber security risks were reduced by 17.9% and 18.8% respectively on the strong and weak structural attack graph.

1. Introduction

To defend against possible attack situations of attackers, the existed framework does not provide a good defense strategy. Relevant work in this field is as follows. Attack graph technology can be used to analyze the vulnerabilities existed in the target network. With regard to the research of attack graphs, the existed research directions were divided into three aspects: attack graph construction [1-4], attack graph analysis [5-12] and target model construction. Target model construction mainly tended to model the attacker model. Attack graph analysis tended to establish a corresponding attack graph for the target network topology. The main direction of attack graph analysis was to analyze the security problem of the network based on the attack graph.

The difficulty in generating a security policy was the multi-step nature of malicious attacks. The specific information of the internal vulnerabilities may be exposed by the internal users because attacker does not have permissions. In this regard, relevant scholars have proposed a number of solutions. Attack graph analysis research was divided into static strategy research [5-6] and dynamic strategy research [11-12]. However, the framework based on the static defence strategy has undergone major changes and cannot adapt to the real-time network environment. In order to cope with the dynamic changes of information systems, it is a very effective method to integrate MTD (Motion Target Defence) [8] into attack graph technology. The main idea was to improve the anti-aggressiveness of information systems by constructing, evaluating and deploying a variety of continuous transformation mechanisms and strategies [9]. The existed research could be divided into
two categories: combined with the game model [10-11] and combined with the advanced model [13]. However, the above models has a common problem, which is to ignore the influence of users in the offensive and defensive game. Some scholars used an uncertain variables represent the unknown effects that may exist in a two-party game, and these effects were caused by the third party participating in the game. The results obtained in this way were not accurate.

In summary, the reason for hindering further development of defence strategy was that the game framework was not comprehensive enough to obtain more accurate results. At the same time, the traditional multi-party game model could not adapt to the game problem in the security field, so it was very important to propose a new security game framework.

2. Dynamic Trilateral Game Model for Bayesian Attack Graph

The multi-step nature of attack behaviour brings many difficulties for security protection. In order to accurately calculate the possibility of nodes being attacked in the network, this section extends the definition of Bayesian attack graph [11].

Bayesian attack graph: Bayesian attack graph is a directed acyclic graph, denoted as $G = (V, E, \theta, p)$

1. $V$ is a non-empty set of node, indicating a set of all vulnerable nodes of the system.
2. $E$ is a set of all edges in the attack graph, indicating the association between the vulnerable nodes.
3. $e = (u, v) \in E$ is the attack path from node $u$ to node $v$. $\pi^-(v) = \{u | (u, v) \in E\}$ denotes the set of parent nodes of $v$, $\pi^+(v) = \{u | (v, u) \in E\}$ denotes a set of child nodes of $v$.
4. $\theta$ represents the node type, $\theta(v) \epsilon \{\Lambda, V\}$ for each node $v \epsilon V$, where $\Lambda$ is the set of "and"-type nodes and $V$ is the set of "or"-type nodes.
5. $p(e^{u,v}) \epsilon (0,1)$ represents the attack probability of attacking node $v$ from node $u$.

The state change of the dynamic attack graph changes as the discrete time metric changes. Between $t=0$ and $t=T$, each game subject determines the behaviour at the moment by observing the state of the attack graph at the previous moment. The behaviour of these subjects comprehensively determines the state of the next moment of the attack graph.

The status of the dynamic attack graph can be represented by the following tuple information:

1. $t \epsilon (0, ..., T)$, the time information of the attack graph status;
2. $S_t = \{s^a_t, s^d_t, s^u_t\}$, the result information of all game subjects at this moment;
3. $\Lambda_t \subseteq \Lambda$, the security situation value of the current attack graph at time.

The behavioral constraints of each strategic decision maker are as follows:

Attacker Constraint: At time $t$, the attacker strategy set is defined as $A_t$, and the attack strategy constraints have the following points. First, when $v \epsilon s^a_{t-1}$, if the attacker intends to add the $v$-node to the set $A_t$, move the node $v$ out of $s^a_t$. Secondly, when $v \epsilon s^d_{t-1}$ or $\pi^-(v) \notin s^a_t$, the attacker cannot join a $v$-node to the set $A_t$.

Defender Constraint: At time $t$, the defender strategy set is defined as $D_t$, and the defender strategy constraints have the following points. First, when $v \epsilon s^u_{t-1}$, if the defender intends to add the $v$ node to the set $D_t$, then this defense is invalid, and the node $v$ is removed from the $s^u_t$. Secondly, when $v \epsilon s^d_{t-1}$ or $s^u_{t-1}$, the defender cannot join a $v$-node to the defender strategy set $D_t$.

User Constraint: At time $t$, the user strategy set is defined as $U_t$, and the user strategy constraints have the following point. When $v \epsilon s^u_{t-1}$ or $s^d_{t-1}$, the user cannot join a $v$-node to the user strategy set $U_t$.

At time $t$, every strategic decision maker can only add one node to its own strategy set.

2.1. Game State Transition Model

The three-party game can be seen as a non-cooperative game in which the three roles are non-zero and completely dynamic information. The defender's behaviour is to spread the threat to the inactive node. The defender's behaviour is that the administrator fixes the vulnerability on the system node, and the user's behaviour is to gain revenue by exposing system information. The parties in the game dynamically select their own behaviour according to the state and behavioural strategies of the system. The system jumps to the next state according to the influence of party's behaviour.
2.1.1. Attacker Strategy Based On Equilibrium Value Propagation

The vulnerability nodes in the network topology interact with each other. The attacker is a rational decision-maker who only focuses on the utilization of the target vulnerability nodes, and expects to use these target vulnerability nodes to profit, while other nodes are not in the attacker's intention. Therefore, the value metric of each vulnerability node is only related to the target node.

Since attacker's attack strategy is subject to the activated node, the attacker can only attack the child nodes of the activated node. For the "or" node, the attacker can directly activate, and for the "and" node, which can only be activated after all parent nodes have been activated. The attack candidate node set $\Psi^a(S_t)$ is constructed as follows:

$$\Psi^a(S_t) = \{(u, v) \in E^v | u \in S_t, v \notin S_t \} \cup \{v \in V \setminus S_t | \text{pre}(v) \subseteq S_t \}$$

Attack value calculation: For each node in the attack graph, its value is always related to the target node. At some point, the attacker is always more inclined to select nodes that are beneficial to the target from the candidate nodes. Nguyen et al. [12] consider that the value of non-target node is indirect value, but node value is the highest value of a single target node, and does not take into account that each target node has an impact. Based on the above considerations, this research improves the node vulnerability propagation algorithm and uses the following formula to calculate:

$$\hat{r}(v) = \gamma(\sum_{u \in \pi^+(v), u \in V^v} \hat{r}(u) \cdot p^{\text{act}}(e_{u,v}) + c^{e_{u,v}}(v)) + \sum_{u \in \pi^+(v), u \in V^v} \frac{r(u) \cdot p^{\text{act}}(e_{u,v}) + c^{e_{u,v}}(v)}{|\pi^-(u)|} / |u|$$

Due to the uncertainty of the attacker strategy, in order to verify the impression of different attacker strategies on the overall network security situation, this research uses the following three attacker strategies:

1. random attack strategy
   At time $t$, the attacker randomly selects an attack node from the candidate nodes as the next attack target.

2. probability attack strategy
   This selection method of the attack node calculates the attack probability of all nodes according to the node value, and the selection probability of each candidate attack node is calculated according to the following formula.

   $$p(v) = \frac{r(v)}{\sum_{u \in \Psi^a(S_t)} \hat{r}(u)}$$

3. greedy attack strategy
   The selection method of the attack node calculates the attack value of all nodes according to the node value, and activate the node with the highest value among the attack nodes.

2.1.2. User Strategy Based On Mixed Strategy Game

The user has the corresponding authority to scan and analyze the vulnerability in the scope of the authority. Users are not concerned about the security of the entire system, only consider their own gains and losses, their means is to expose the analyzed vulnerability information to the attacker. For each user's accessible vulnerability node, the revenue is the value of this node, and the cost is legal cost. The legal cost of the user's exposure to vulnerability information and the gains obtained are related to the attacker's behaviour. For this problem, the mixed strategy game can be used to model the attacker-user model. The resulting mixed strategy Nash equilibrium is the user's behavioural guidance.

Legal cost: The user exposed the information about the vulnerability node has a legal cost, the cost is positively related to the distance between the node and the target node. The legal cost of the node is the sum of the risk propagation values of all the important nodes here, which is expressed as follow:

$$c^u(v) = \sum_{u \in \pi^+(v), u \in V^v} (c^u(u) \cdot p^{\text{act}}(e_{u,v})) + \sum_{u \in \pi^+(v), u \in V^v} \frac{c^u(u) \cdot p^{\text{act}}(e_{u,v})}{|\pi^-(u)|} / |u|$$
Full benefit: When user's interference policy and attacker's attack strategy are the same as $v$, the net income value that the user can obtain is recorded as $r(u)(v) = r^u(v) - c^u(v)$, where $r^u(v)$ is the user benefit of node $v$, $c^u(v)$ is the legal cost of node $v$.

Revenue Matrix: For each node, different behaviours of attackers and users will have different income functions. Assume that the income matrix is $Q$, which is expressed as $Q = \{q_{ij}\}_{i=1}^{n}$, where $n$ represents the number of vulnerable nodes. The mathematical expression is as follows:

$$seq(v) \equiv \{v\} \cup pre(v) \cup \{pre(v)\} ...$$

$$q_{ij} =
\begin{cases}
\delta^k y^u(v), & \text{if } v_i \in seq(v_j) \\
0, & \text{if } v_i \notin seq(v_j) \\
\delta y^u(v), & \text{other}
\end{cases}$$

Where $k$ is the distance between the attack node and the user exposed node.

User-attacker game model: The outcome of each of the choices of the attacker and the user is represented by revenue matrix, and the user wishes to have a strategy that maximizes the expected value of the mixed strategy game. For the above problem, it is specified that the user's strategy set is $U$, and the mixed strategy expectation is $v$, wherein each element in the user strategy set represents the user's tendency to select a node mode. The sum of the user's propensity for all the selected node modes is 1. The above can be expressed using the following mathematical expression:

$$\sum_{i=1}^{n} u_i j \geq v(j = 1, ..., n) \text{ for } u_1 + u_2 + \cdots + u_n = 1 \text{ and } u_i \geq 0 (i = 1, ..., m)$$

For the above formula, the user's target can be expressed as $\max v$, since $v$ is a number greater than 0, so the target can be converted to $\min \frac{1}{v} = \min \frac{u_1 + u_2 + \cdots + u_n}{v}$, let $u'_i = \frac{u_i}{v}$ can represent the above equation into another form:

$$\sum_{i=1}^{n} u'_i j \geq 1 (j = 1, ..., n), \text{ for } u'_i \geq 0 (i = 1, ..., m)$$

The linear programming problem can be solved to obtain the mixed strategy expectation value and the user strategy set, wherein the largest propensity strategy in the user strategy set is the user selected strategy.

2.1.3. Defender Strategy Based On Trilateral Game

As a defensive party, the defender always expects a better security posture in the entire network structure. After the entire network topology is reinforced, the entire network topology has higher risk resistance capability. The defender needs to consider the user's exposure strategy to deploy the defence strategy based on the attacker's attack strategy. The defence strategy of the defender can be expressed as the following steps.

1. At the time $t+1$, randomly select $N^a$ attack path and the set $[A^K]$ represents attack paths, where $k = 1, 2, ..., N^a$. The set of policies in $A^K$ is represented as $ra(A^K)$.

2. For the blocking of each attack path, security situation assessment is conducted. The security situation assessment in this paper uses the evaluation indicators of Yang [14] to evaluate the overall network security situation. The calculation formula is as follows:

$$\Lambda(A^K) = \sum_{j=1}^{m} p_j(s) \text{Impact}(v)\text{Weight}$$

Where $p_j(s)$ is the realization probability of the attack stage, which refers to the possibility that the attacker has successfully invaded the state of a certain stage; $\text{Impact}(v)$ is the score of all the individual vulnerability threats used in the attack phase; $\text{Weight}$ is the node weight value during the attack phase.

3. Select a path with the largest value of $\Lambda(A^K)$ from $ra(A^K)$, select the non-exposed node in the intersection for the intersection of path nodes set and the attack candidate node. if there is no other node in the intersection, move $A^K$ out of $ra(A^K)$ and repeat step 3 until the corresponding defense harden node is selected.
2.2. Security Situation Algorithm Based On Trilateral Game
Exploit Probability: For the path $e_{u,v}$, the vulnerability $v$ is a possible vulnerability. The probability of attack on the path is the probability of exploiting the vulnerability. The probability is calculated based on the three attributes: AV, AC and AU. Calculated as follows:

$$p(e_{u,v}) = 2 \times AV \times AC \times AU$$

Node threat score: According to the official CVSS document [15], the threat score for the target node is calculated as follows:

$$\text{Impact}(v) = 10.41 \times (1 - (1 - C)(1 - I)(1 - A))$$

Node weight value: In the entire network topology, every node has a certain weight ratio to prove the importance of these nodes in this topology, so as to comprehensively quantify the node importance of the attack phase.

Node attack cost: The attacker's attack requires cost. The main reason for the cost is the attacker's understanding of the vulnerability and how the vulnerability is displayed on the network. For known vulnerabilities, the attack cost is mainly considered from three aspects:
1. Whether the exploit method is released;
2. Whether the exploit tool is used to spread on the network;
3. Whether the vulnerability is needed to exploit the vulnerability.

The above three aspects of comprehensive measurement can be used as a node attack cost calculation indicator.

![Figure 1](image.png)

**Figure 1.** Schematic diagram of the state change of the tripartite game.
At time $t$, the three parties simultaneously observe the attack graph. By observing the state of the attack graph, the behaviours of the parties in the game comprehensively determine the next state of the attack graph. The game state changes as shown in the Figure 1.

The game is played as follows:
1. According to the behavioural constraints, the defender selects the defense strategy $D_t$.
2. Under $D_t$, the user selects the state of the corresponding attack graph to select the $U_t$.
3. Under the joint action of $D_t$ and $D_t$, the attacker observes the current attack graph state and performs policy selection to obtain $A_t$.

The security situation value of the network attack topology at each moment in the game process is calculated by the following formula until the arrival time maximum $T$.

$$\Lambda_t = \sum_{i=1}^{n} A_i$$

Where $A_i^t$ is the arbitrary attack path at the current time, and $n$ is the number of attack paths.

3. Experiment
In order to eliminate random conditions, this section conducts experiments through the random network structure. This section of the experiment is divided into two groups of experiments, which use different network topologies. The two topologies are: (i) A layered directed attack graph model; (ii) a random directed attack graph [12]. The control experiment applied the method of the paper [12] to the framework.
3.1. Random Network Validation

In order to eliminate random conditions, this section conducts experiments through the random network structure. This section of the experiment is divided into two groups of experiments, which use different network topologies, which are mainly used to verify the validity of the model in the case of large-scale order and large-scale disorder. The two topologies are: (i) A layered directed attack graph model; (ii) a random directed attack graph [12].

3.1.1. Layered Attack Graph

The structural model is an ordered structure model, and it is a 5-layer directed attack graph, where the $k^{th}$ ($k = 1, \ldots, 5$) layer has $25 \times 0.8^{k-1}$ nodes, and the last layer's nodes are the target node. Each layer node is connected to 50% of the next level node, and the nodes are randomly selected. Since the node attribute has a certain influence on the behavior of the game, the attack graphs of the 30%, and 50% three 'and' node proportions are experimentally verified, as shown in Figure 2. The meaning of the x-axis in the subgraph is the proportion of the root node in the attack graph, and the meaning of the y-axis is the security situation value. The smaller security situation value, the smaller the damage to the system. In order to prevent extreme experimental data from appearing in the experiment. In the same experimental environment, the result is a 100-fold average of the results produced by the environment. In this structure, there are three types of attacker attack strategies, and the results of these attack strategies are verified. Combining the results of each strategy, the model improves the security of the layered directed attack graph by 17.9% compared to the two-player game model.

3.1.2. Random Attack Graph

The structural model is an unordered structural model, and the scale of the attack graph is set to $|V|=100$ and $|E|=300$. In this section, 15 nodes are randomly selected as attack target nodes in the attack graph. Similar to the experiment with layered attack graph, the attack graphs of the 30%, and 50% three 'and' node proportions are experimentally verified, as shown in Figure 3. The result is 100 times the average of the results produced by the environment. Combining the results of the three attack strategies, the model improves the security of the random directed attack graph by 18.8% compared to the two-player game model.
The experimental results show that the dynamic trilateral game model is effective and feasible on various attack graphs.

4. Conclusion
This research focuses on the problem of deploying security policies on attack graphs to deal with cyber attack. In order to improve the effectiveness of cyber defenses, this paper considers the influence of the third player in the offensive and defensive game on both offense and defense, improves the existed two-player game model and proposes a dynamic trilateral game model based on Bayesian attack graph. The experimental verification is carried out under the strong and weak structural attack graph. The results show that the proposed model has feasibility and effectiveness in various attack graph topologies.

Acknowledgments
This work is supported by the 13th Five-Year Equipment Development Project (41401010201); the 13th Five-Year Key Basic Research Project (JCKY2016206B001).

References
[1] Swiler, L.P, & Phillips, C. 1998. A graph-based system for network-vulnerability analysis. Proceedings of the Workshop on New Security Paradigms, 71-79.

[2] Sheyner, O., Haines, J., Jha, S., Lippmann, R., & Wing, J. M. 2002. Automated generation and analysis of attack graphs. IEEE Security and Privacy Magazine, 1971, 273.

[3] Wang, L., Yao, C., Singhal, A., & Jajodia, S. 2006. Interactive analysis of attack graphs using relational queries.

[4] Liu, Y., & Man, H. 2005. Network vulnerability assessment using Bayesian networks. Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security 2005. International Society for Optics and Photonics.

[5] Fredj, O. B. 2015. A realistic graph-based alert correlation system. John Wiley & Sons, Inc.

[6] Liu, S. C., & Liu, Y. 2016. Network security risk assessment method based on HMM and attack graph model. 2016 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD). IEEE.

[7] Gao, L., Wang, F., & Gao, N. 2019. An security hardening measures selection model based on improved ant. Computer Engineering and Applications.

[8] Jajodia, Sushil, Ghosh, Anup K, Swarup, & Vipin. 2011. Moving target defense. , 54, 99-108.

[9] Xu, J., Guo, P., Zhao, M., Erbacher, R. F., Zhu, M., & Liu, P. 2014. Comparing different moving target defense techniques. In Proceedings of the First ACM Workshop on Moving Target Defense (pp. 97-107). ACM.

[10] Poolsappasit, N., Dewri, R., & Ray, I. 2012. Dynamic security risk management using bayesian attack graphs. IEEE Transactions on Dependable and Secure Computing, 9(1), 61-74.

[11] Nguyen, T. H., Wright, M., Wellman, M. P., & Singh, S. 2018. Multistage Attack Graph Security Games: Heuristic Strategies, with Empirical Game-Theoretic Analysis. Security and Communication Networks, 2018.

[12] Lei, C., Ma, D. H., & Zhang, H. Q. 2017. Optimal strategy selection for moving target defense based on markov game. IEEE Access, PP(99), 1-1.

[13] Berg, K. 2019. Set-valued games and mixed-strategy equilibria in discounted supergames. Discrete Applied Mathematics, 255, 1-14.
[14] Hao-Pu, Y., Hui, Q., Kun, W., & University, I. E.. 2017. Network security situation evaluation method for multi-step attack. Journal on Communications.

[15] A Complete Guide to the Common Vulnerability Scoring System [EB/OL]. https://www.first.org/cvss/v2/guide