Hindawi
Security and Communication Networks
Volume 2021, Article ID 4292670, 12 pages
https://doi.org/10.1155/2021/4292670

Research Article

A Defense Strategy Selection Method Based on the Cyberspace Wargame Model

Yuwen Zhu,[1] **Lei Yu**,[2] **Houhua He**,[2] **and Yitong Meng**[1]

[1]*State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou 450001, China*
[2]*Institute of Information Engineering Chinese Academy of Sciences, Beijing 100093, China*

Correspondence should be addressed to Yuwen Zhu; sss39639@163.com

Received 16 July 2021; Accepted 1 October 2021; Published 27 October 2021

Academic Editor: Mamoun Alazab

Network defenders always face the problem of how to use limited resources to make the most reasonable decision. The network attack-defense game model is an effective means to solve this problem. However, existing network attack-defense game models usually assume that defenders will no longer change defense strategies after deploying them. However, in an advanced network attack-defense confrontation, defenders usually redeploy defense strategies for different attack situations. Therefore, the existing network attack-defense game models are challenging to accurately describe the advanced network attack-defense process. To address the above challenges, this paper proposes a defense strategy selection method based on the network attack-defense wargame model. We model the advanced network attack-defense confrontation process as a turn-based wargame in which both attackers and defenders can continuously adjust their strategies in response to the attack-defense posture and use the Monte Carlo tree search method to solve the optimal defense strategy. Finally, a network example is used to illustrate the effectiveness of the model and method in selecting the optimal defense strategy.

1. Introduction

With increasingly complex network environment and diverse attack methods, network defenders with limited defense resources can hardly solve all network flaws and defend against all attacks. Therefore, the key to network defense is to make the most use of limited resources for the most reasonable defense decisions. Selection of network defense strategy seeks the equilibrium point for network attack-defense game, with attack-defense revenue thoroughly considered [1]. Currently, network defense strategy selection technologies mainly include attack-defense models [2, 3], strategy quantification and selection [4], and game theory [5–7]. The network attack-defense game model is an effective means to solve this problem [8]. In order to quantitatively analyze the game process elements such as attack-defense scenarios, processes, and cost-revenue, the attack-defense game model is indispensable.

Nevertheless, the existing network attack-defense game model usually assumes that the defender will no longer change after deploying the defense strategy. However, in advanced network attack and defense confrontation, the defender usually redeploys defense strategies for different attack situations. Therefore, how to accurately reflect the dynamic interactions and deduction processes between attackers and defenders and how to select the optimal defense strategy have drawn extensive attention from domestic and foreign scholars.

According to the abovementioned problems, this paper aims to study the defense strategy selection method in the dynamic game network. We model the high-level network attack-defense confrontation process as a turn-based wargame in which both attackers and defenders can continuously adjust their strategies in response to the attack-defense posture and use the Monte Carlo tree search method to solve the optimal defense strategy. We conclude our contributions as follows:

(i) We propose a formal description method for the selection of optimal defense strategies, which formally defines the selection of optimal defense strategies for network security

(ii) We propose a network attack-defense wargame model, which is a turn-based wargame and both attackers as defenders can continuously adjust their strategies in response to the attack-defense posture

(iii) We propose a defense strategy selection method based on Monte Carlo tree search, using artificial intelligence methods to analyze the attack-defense strategies

(iv) We design a simulation instance which is used to illustrate the effectiveness of the model and algorithm in selecting the optimal defense strategy

The rest of this paper is structured as follows. The second section discusses the related work. The third section details the formal description of the optimal defense strategy. The fourth section discusses the network attack-defense wargame model. Continuing on this model, the fifth section uses Monte Carlo tree search to select the defense strategy method. The sixth section proposes an example to illustrate the effectiveness of the model and algorithm. The seventh section gives the comparison of related work. Finally, the eighth section summarizes the paper and proposes future work.

2. Related Work

Although some research results have been achieved about attack-defense models [2, 3], strategy quantification and selection [4], and game theory [5–7], it is still in its infancy and no systematic theoretical methods have been formed.

Lee et al. [9] first proposed a cost-sensitive model as the basis of response decisions in 2002, which determined whether to respond or not according to the attack cost and revenue. The decision-making idea was relatively simple, and the quantification of cost-revenue was relatively rough. However, the ideas and methods of cost-revenue quantification, classification, and attack classification can be used for reference. Li et al. [10] established a noncooperative game model between attacker and sensor trust node and gave the optimal attack strategy by calculating Nash equilibrium. Because of the complexity of the restriction conditions of Nash equilibrium, Serra et al. [11] used the Pareto optimization method to calculate the Nash equilibrium solution of the game. Esmalifalak et al. [12] took the attack and defense times as the basic strategy of both sides, established a complete information two-person zero-sum game model, and verified it in the system. Wu et al. [13] used the reinforcement learning algorithm to solve Nash equilibrium and realize security situation analysis and prediction of an intelligent system. Liu et al. [14] investigated how to achieve such a trade-off optimally of cost-revenue by proposing a two-player strategic game model between the attack and the defender. Then, a graph-based simulated annealing algorithm is proposed to derive the utility-maximising strategy. Wang et al. [15] analyzed the influence of the selection of cooperation strategy on the cooperation effect of sensor network nodes with the help of an evolutionary game. Na et al. [16] and Abass et al. [17] calculated the optimal evolutionary stability strategy against DoS attack and APT attack by using replication dynamic equation, respectively. Hayel and Zhu [18] established an evolutionary Poisson game model between malware and antivirus programs and analyzed the program opening strategy about the replication dynamic equation. Consider that the randomness of attack and defense means would inevitably lead to the state jump of the game system.

3. Formal Description of Optimal Defense Strategy Selection

The complex network topology, coupled with the various network node states, is difficult to describe in real-time. For example, in the network structure of n nodes, there are $2^{1/2(n-1)n^2}$ kinds of different situation combinations, where 2^n represents the authority status of the node and $2^{1/2(n-1)n^2}$ represents the state type of the network topology. Therefore, a formal description of the network attack-defense environment can greatly reduce the computational complexity.

Definition 1 (network topology matrix). For the network topology, it can be represented by a two-tuple $G = (V, E)$. If an n-order square matrix $A_{[n\times n]}$ is used to represent the network topology, the square matrix satisfies the following formula:

$$A_{[i,j]} = \begin{cases} 1 < v_i, \quad v_j > \in E, \\ 0 < v_i, \quad v_j > \notin E, \end{cases} \tag{1}$$

where the two-tuple G represents the network topology $G = (V, E)$, V represents the set of nodes in the network $V = \{v_1, v_2, \ldots, v_n\}$, and E represents the set of edges $E = \{e_1, e_2, \ldots, e_m\}$.

Definition 2 (vulnerabilities of network nodes). Vulnerability(v) represents the vulnerability $\mathrm{Vul}_1, \mathrm{Vul}_2, \mathrm{Vul}_3, \mathrm{Vul}_4, \ldots\ldots, \mathrm{Vul}_n$ of node v in the network as follows:

$$\text{vulnerability}(v) = (\mathrm{Vul}_1, \mathrm{Vul}_2, \mathrm{Vul}_3, \mathrm{Vul}_4, \ldots\ldots, \mathrm{Vul}_n). \tag{2}$$

Definition 3 (network attack reachable node vector). For the network attack reachable node vector at time t, it is represented by the following vector:

$$\overrightarrow{R}(t) = (r_1, r_2, r_3, r_4, \ldots\ldots, r_n), \tag{3}$$

where n is the number of nodes in the network. Each node vector is calculated as follows:

$$r_i = \begin{cases} 1, & \text{node } i \text{ is reachable in the next attack,} \\ 0, & \text{node } i \text{ is unreachable in the next attack.} \end{cases} \tag{4}$$

Definition 4 (attack-defense posture vector). Attack-defense posture vectors can reflect the permission of each node in the network. In this paper, we believe that the permission of nodes in the network does not belong to the attacker; it must belong to the defender. It is represented by the following vector:

$$\overrightarrow{S}(t) = (s_1, s_2, s_3, s_4, \ldots\ldots, s_n), \tag{5}$$

where n is the number of nodes in the network.

$$s_i = \begin{cases} 1, & \text{Node } i \text{ permission belongs to the defender,} \\ -1, & \text{Node } i \text{ permission belongs to the attacker.} \end{cases} \tag{6}$$

Definition 5 (the attack strategy node vector $\overrightarrow{A}(t)$). It represents the next attack node under the state $\overrightarrow{S}(t)$. The attack strategy vector is expressed as follows:

$$\overrightarrow{A}(t) = (0, 0, 0, \ldots, a_i, \ldots, 0), \tag{7}$$

where each node vector is calculated as follows:

$$a_i = \begin{cases} 1, & \text{Node } i \text{ permission belongs to the defender after the game,} \\ -1, & \text{Node } i \text{ permission belong to the attacker after the game,} \end{cases} \tag{8}$$

where there is only $a_i \neq 0$ in $\overrightarrow{A}(t)$, and a_i means that node i is the node of the current round of confrontation game.

From the definition, we can know that $\overrightarrow{A}(t) \cap \overrightarrow{R}(t) = \overrightarrow{A}(t)$. At time t, the nodes involved in the next attack strategy must belong to the reachable nodes of the network attack.

Definition 6 (target vector $\overrightarrow{T}$). After several strategy combinations, the target is reached, and the target vector is expressed as follows: $\overrightarrow{T} = (t_1, t_2, t_3, t_4, \ldots, t_{\text{target}}, \ldots, t_n)$ where $t_{\text{target}} = -1$ represents that the goal of the attacker is to obtain the permission of the node target.

According to the above description, under certain attack-defense resources, based on the attack-defense game rules, the attacker's permission to seize the node target can be described as given network topology $A_{[n\times n]}$, an initial attack to reach the node vector $\overrightarrow{R}(t)$, and the initial attack-defense situation vector $\overrightarrow{S}(t) = 1$, after several strategies $\overrightarrow{A}(t)$, and the target $\overrightarrow{T}(t)$ is reached. The defense strategy selection problem implemented by the defender can be described as how the defender allocates the defense resource for each attack strategy $\overrightarrow{A}(t)$ of the attacker.

4. Network Attack-Defense Wargame Model

Wargame [19] is a kind of turn-based, role-playing, strategy game. This paper models the high-level network attack-defense confrontation process as a turn-based wargame in which both attackers and defenders can continuously adjust their strategies in response to the attack-defense posture.

4.1. Model Hypothesis. The hypothesis of the network attack-defense wargame model is as follows.

Hypothesis 1 (rational hypothesis). Assuming that the attacker and the defender are completely rational, the attacker will not launch unprofitable attacks and the defender will not defend at all costs.

Hypothesis 2 (cost assumptions). The goals of the attacker and the defender are to obtain and protect their network equipment. Both parties can be quantified and measured during the game to the offensive and defensive costs.

Hypothesis 3 (game hypothesis). Assume that the winner can replace the loser in the attack-defense game to obtain all the permissions of the node. If the attacker succeeds, it means that he will not be discovered by the defender and will proceed to the next round of attack-defense games. If the attacker fails, the defender can redeploy defense measures which are very important and effective. The existing network attack-defense game model usually assumes that the defender will not change after deploying the defense strategy, but in a high-level network attack and defense confrontation, the defender usually redeploys defense strategies for different attack situations. Last, if the game is flat, it means that the attacker is not discovered by the defender, and this node can be used as a springboard node for the next round of the game.

Hypothesis 4 (attack hypothesis). Assume that at the beginning of the attack, the permissions of all nodes in the network belong to the defender and no less than two network devices are exposed to the external network. If there is only one entry node, the defender only needs to protect the node, and there is no game process.

4.2. Formal Description. In the network attack-defense game, the attacker finally obtains the target node permissions by obtaining the node permissions of the defender on the attack path. The following shows the network attack-defense wargame model in the multigame state.

Definition 7. The network attack-defense wargame model (NADWM) is given below. It can be represented by the eight-tuple.

$\overrightarrow{\text{NADWM}}$ = (Participant, Time, State, Strategy, Resource, $\overrightarrow{\text{Resource}}$, Cost, Revenue), where

(i) Participant = $\{p1, p2, p3, \ldots\ldots, p_n\}$ is the set of players in the game, where $n \geq 2$. Participants are individuals or organizations that participate in the game and independently make decisions and bear the results. In this paper, participants are the attacker p_A and the defender p_D.

(ii) Time = $\{t_1, t_2, t_3, \ldots\ldots, t_n\}$. Network attack-defense is a dynamic and continuous confrontation process, which needs to be modeled and analyzed from the time dimension. Time is a set of attack-defense time series.

(iii) State = $\{\text{state}_1, \text{state}_2, \text{state}_3, \ldots, \text{state}_n\}$. There are many states of network attack-defense games. State_t is the state of the player at time t in the current attack and defense process

(iv) Strategy = $\{\text{strategy}_A, \text{strategy}_D\}$. Strategy represents the strategy space. strategy_A represents the strategy space of the attacker and strategy_D represents the strategy space of the defender.

(v) Resource = $\{\text{Resource}_A(t), \text{Resource}_D(t)\}$. Resource represents the resource of the attacker and the defender. $\text{Resource}_A(t)$ presents the resource value of the attacker at time t, and $\text{Resource}_D(t)$ represents the resource of the defender at time t. When $\text{Resource}_A(t) \leq 0$, the attacker is not enough to launch an attack, and the game is judged to be a failure.

(vi) $\overrightarrow{\text{Resource}} = (\overrightarrow{\text{Resource}_A(t)}, \overrightarrow{\text{Resource}_D(t)})$. $\overrightarrow{\text{Resource}}$ represents resource allocation vector.

Attacker resource allocation vector $\overrightarrow{\text{Resource}_A(t)} = (a_1, a_2, a_3, a_4, \ldots\ldots, a_n)$, where n is the number of nodes in the network and $\overrightarrow{\text{Resource}_A(t)}$ represents the attack force distribution of the attacker at each node at time t. There is one and only one $a_i \neq 0$ in $a_1, a_2, a_3, a_4, \ldots\ldots, a_n$, which represents that the attacker attacks at node i at time t.

Defender resource allocation vector $\overrightarrow{\text{Resource}_D(t)} = (d_1, d_2, d_3, d_4, \ldots\ldots, d_n)$, n is the number of nodes in the network and $\overrightarrow{\text{Resource}_D(t)}$ represents the defense force distribution of the defense side at each node at time t.

(i) Cost = $\{\text{cost}_A(v_i, \text{Vul}_j, t), \text{cost}_D(v_i, t)\}$ represents the attack-defense cost function at time t in the network attack-defense game process, including the attacker cost $\text{cost}_A(v_i, \text{Vul}_j, t))$, which represents at time t the attacker attack the defender in node v_i by using vulnerability Vul_j and the defender cost $\text{cost}_D(v_i\, t)$ represents the defense force deployed by the defender on node v_i at time t.

(ii) Revenue = $\{\text{revenue}_A(t), \text{revenue}_D(t)\}$ represents the attack and defense revenue function at time t in the network attack-defense game process, including the attacker revenue $\text{revenue}_A(t)$, which represents the revenue of the attacker attacking the defender at time t, and the defender revenue $\text{revenue}_D(t)$, which represents the revenue of the defender defending the attacker at time t.

Table 1: Vulnerability price and level correspondence table.

Level	Price
0	\$0
1	\$0-\$1 k
2	\$1 k-\$2 k
3	\$2 k–\$5 k
4	\$5 k–\$10 k
5	\$10 k–\$25 k
6	\$25 k–\$50 k
7	\$50 k–\$100 k
8	≥\$100 k

4.3. Cost-Revenue Quantification and Attack-Defense Strategy

4.3.1. Cost-Revenue Quantification. In the process of the network attack-defense game, the cost-revenue function needs to be quantified.

$\text{Cost}_A(v_i, \text{Vul}_j, t)$ means the cost that the attacker is attacking the defender in node v_i by using vulnerability Vul_j at time t. The attack cost in this model represents the value of vulnerabilities used by the attacker, assuming that the attacker purchases vulnerabilities and exploits tools through the vulnerability market. So, higher value of vulnerability leads to higher cost of attack cost, which further leads to greater difficulty and higher cost of defense. However, once the vulnerability is found by defenders, it will lose value as defenders can simply deploy firewall rules.

This paper uses the vulnerability price proposed in the VulDB library to evaluate the attack cost. VulDB is the number one vulnerability database worldwide with more than 178000 entries available. Its specialists work with the crowd-based community to document the latest vulnerabilities every day since 1970, providing technical details, and additional threat intelligence such as current risk levels and exploit price forecasts. Their price estimations are calculated via mathematical algorithms developed by their specialists over the years through observing the exploit market and exchange behavior of involved actors. It allows the prediction of generic prices by considering multiple technical aspects of the affected vulnerability [20].

This paper quantifies the cost of attack into 9 different levels according to the price range proposed by VulDB, $\text{cost}_A(v_i, \text{Vul}_j, t) \in \{0, 1, 2, 3, \ldots, 8\}$. Higher level represents higher cost of attack as well as higher value of vulnerability. The price and level of the vulnerability are shown in Table 1.

(1) $\text{Revenue}_A(t)$. The revenue of the attacker represents at time t. In the attack-defense game, the attack revenue increases by k when they get the permission of the nontarget node, which is the path node revenue. When the attacker obtains the permission of the target node, the attack revenue

increases by $n \times k$, where n is the number of nodes in the network. In the process of attack, the ultimate goal of the attacker is to obtain the permission of the target node, so the sum of attack revenue obtained by the node on the attack path should be less than or equal to the attack revenue obtained by the target node. Otherwise, the attacker can achieve the maximum attack revenue as long as he focuses on the attack process. In this paper, for the convenience of calculation, $k = 1$:

$$\text{revenue}_A(t) = \begin{cases} 1, & \text{Attacker obtains the permission of the nontarget node,} \\ n, & \text{Attacker obtains the permission of the target node,} \\ 0, & \text{Attacker do not obtain the node permission.} \end{cases} \tag{9}$$

(2) $\text{Cost}_D(v_i, t)$. The cost of defender represents the defense cost of the defender deployed on the node v_i. There are many kinds of defender costs, such as manpower, equipment, and resources. In this paper, we do not consider the specific methods of defense but only pay attention to the allocation of limited defense resources to the network topology. Corresponding to the attack cost, the defense cost is quantified into 10 different levels, $\text{cost}_D(v_i, t) \in \{0, 1, 2, 3, \ldots, 10\}$. The higher the level, the more defense resource deployed in the node.

(3) $\text{Revenue}_D(t)$. The revenue of the defender represents the defense revenue at time t. Since both attack and defend belong to a zero-sum game, $\text{revenue}_A(t) + \text{revenue}_D(t) = 0$ [21]:

$$\text{revenue}_D(t) = \begin{cases} 1, & \text{Attacker obtains the permission of the nontarget node,} \\ n, & \text{Attacker obtains the permission of the target node,} \\ 0, & \text{Attacker do not obtain the node permission.} \end{cases} \tag{10}$$

4.3.2. Attack-Defense Strategy. In the process of network attack-defense game, it is necessary to define the attack and defense strategy.

Definition 8. Attack and defense strategy set $S^k = (S_A^k, S_D^k)$. It represents the set of action strategies taken by the attacker or defense in the game state$_k$.

$$\text{Attack strategy set } S_A^k = \left(S_A^k(a_1), S_A^k(a_2), S_A^k(a_3), \ldots, S_A^k(a_n)\right), \tag{11}$$

where $k = 1, 2, 3, \ldots, K$, $S_A^k \in \text{strategy}_A$, and $S_A^k(a) = S_A^k(\overrightarrow{\text{resource}_A})$

$S_A^k(a)$ represents the strategy of attack resource allocation, where $r_1, r_2, r_3, r_4, \ldots\ldots, r_n$ has one and only one $r_i \neq 0$. It means that the attacker launches an attack at node i at this time, and $\text{cost}_A = r_i$.

$$\text{Defend strategy set } S_D^k = \left(S_D^k(d_1), S_D^k(d_2), S_D^k(d_3), \ldots, S_D^k(d_m)\right), \tag{12}$$

where $k = 1, 2, 3, \ldots, K$, $S_D^k \in \text{strategy}_D$, and $S_D^k(d) = S_D^k(\overrightarrow{\text{resource}_D})$.

$S_D^k(d)$ represents the strategy of defense resource allocation at each node in this state.

Definition 9 (Nash equilibrium [22]). The game model of NADWM is a zero-sum random game. In the game state$_k$, attack and defense strategy set can be expressed as follows:

$$\begin{aligned} S_A^k &= \left(S_A^k(a_1), S_A^k(a_2), S_A^k(a_3), \ldots, S_A^k(a_n)\right), \\ S_D^k &= \left(S_D^k(d_1), S_D^k(d_2), S_D^k(d_3), \ldots, S_D^k(d_m)\right). \end{aligned} \tag{13}$$

The stable mixed strategy $((S_A^i(a^*)), (S_D^i(d^*)))$ is a Nash equilibrium if and only if the mixed strategy is the optimal response of both offense and defense.

Due to the huge number of strategy choices, this paper uses the Monte Carlo tree search method in the fifth part to solve the Nash equilibrium. The final task of Monte Carlo tree search is to find the Nash equilibrium. The way to find is through continuous selection, expansion, and simulation and finally backpropagation whether the strategy on this path is appropriate. The victorious party increases all the utility, while the losing party reduces the action utility, and a balanced state can be reached finally.

4.4. State and State Transition Rules. The construction of the network attack-defense game process consists of three steps: determining the initial state of the model, setting the rules of game state transition, and the termination state of the game.

4.4.1. The Initial State of the Model. To start the game, the following conditions must be determined:

(i) In the game process, there is only one attacker p_A and one defender p_D in the target network structure.

(ii) In the initial stage, the permission of all nodes belongs to the defender, and the defender has no less

than 2 network devices exposed in the outer network. If only one device, the optimal strategy of the defender is to put all attack resources in this outer network node, and there is no intelligent game.

(iii) The defender distributes the defense costs for all nodes in the network.

4.4.2. Game State Transition Rules. The game state transition rules satisfy the following two conditions:

(i) After the game starts, the attacker can move randomly in any direction within the network topology reachable nodes, and each move step needs to consume the attack cost cost_A.

(ii) In the process of network attack-defense, the attacker has a strong pertinence when attacking a target, while for the defender, it has a wide range of universality in the defense process. Simply put, an exploit attack may only apply to a certain environment of a certain node in the network, and in most cases, the defensive tools apply to both computers and servers. Therefore, during each attack, only the attacker will deduct the cost of the attack from its total resource. Then, the game is played according to the defense cost and attack cost. When $\text{cost}_A > \text{cost}_D$ attacker wins, node permissions belong to the attacker, the defender does not detect the attacker, and judge whether it is a target node. If it is not a target node, $\text{revenue}_A = \text{revenue}_A + 1$ and $\text{revenue}_D = \text{revenue}_D - 1$. Then, the attacker will proceed to the next attack. When $\text{cost}_A = \text{cost}_D$, both attacker and defender are tied and the node belongs to the defender, but the attacker can reach the next node from this node. When $\text{cost}_A < \text{cost}_D$, defender wins. Defenders can redistribute the defense revenue during each round of the game.

4.4.3. Game Termination State. The game ends if any of the following conditions were met:

(i) The attack resource is less than or equal to 0.

(ii) The attacker has no new reachable nodes to choose from.

(iii) The attacker arrives at the target node.Revenue is updated $\text{revenue}_A = \text{revenue}_A + n$ and $\text{revenue}_D = \text{revenue}_D - n$.

In the process of attack-defense games, the goal of the defender is to ensure the security of the target node, as much as possible to ensure the security of other nodes in the network. In other words, the defender is to reduce the attack revenue under the premise of a certain amount of attack-defense resources. The simulation operation process of NADWM is shown in Figure 1.

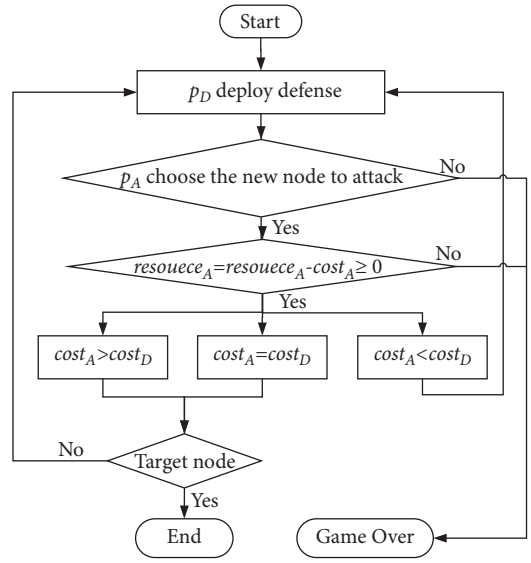

Figure 1: Flow chart of NADWM simulation operation.

5. Defense Strategy Intelligent Selection Method Based on Monte Carlo Tree Search

5.1. Algorithm Idea. In the model of NADWM, as a large number of game states lead to a huge amount of computation, it is difficult to select the optimal defense strategy by using the enumeration method. This paper adopts the Monte Carlo tree search method, which is a heuristic search algorithm based on the tree data structure, which is still effective in the huge search space [23]. For different attack states, this scheme provides a relatively suitable defense strategy for the defender.

5.2. Algorithm Description. Recently, Monte Carlo tree search is boosting the performance of computer Go playing programs which is a tree search strategy that balances the history and future returns. The basic principle is to randomly select the maneuver strategy and then update the value of the originally selected strategy through expected return. This algorithm makes a large number of repeated random simulations until the best strategy appears. Specifically, MCTS is divided into 4 parts, Selection, Expansion, Simulation, and Backpropagation. It is empirically proved that the performance of MCTS scales well against the number of simulations to select an optimal move in computer Go. In addition, developing efficient parallel MCTS (PMCTS) algorithms is important to improve the performance because single processor's performance may not be expected to increase as used to [24]. The PMCTS principle is shown in Figure 2.

5.2.1. Monte Carlo Tree Search Steps. Monte Carlo tree search is essential to maintain a tree, in which each node corresponds to a specific situation state R. The edge of this

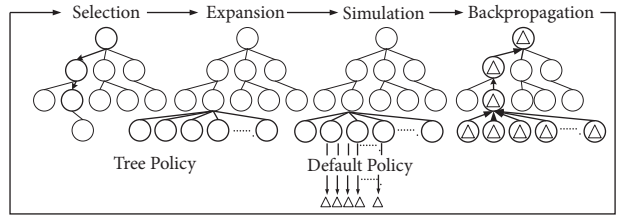

Figure 2: The principle of the PMCTS algorithm.

node is composed of all game actions of both attack-defense parties in state R [25], as shown in Figure 3.

Step 1. Selection. Select the node to go next.

Definition 10. $N_{(R,p)}$ represents the number of times the node performs action p in state R

Definition 11. $\text{Revenue}_{(R,p)}$ represents the attack revenue of the node to perform action p in state R.

Definition 12. $Q_{(R,p)}$ represents the average attack revenue obtained by the node performing a series of actions p in state R and reflects the level of attack revenue that state R can provide, which is calculated as the following equation:

$$Q_{(R,p)} = \frac{\text{revenue}_{(R,p)}}{N_{(R,p)}}. \tag{14}$$

Definition 13. $T_{(R,p)}$ represents the relative defense revenue. Because the attack-defense parties belong to a zero-sum game [19], the higher the revenue of the attack, the worse the defense strategy, and $T_{(R,p)}$ is calculated as the following equation:

$$T_{(R,p)} = -Q_{(R,p)} + \beta, \tag{15}$$

where β is a positive number to ensure $T_{(R,p)} \geq 0$.

To select a node at the current position, the defender needs to select one from all legal strategies that satisfy the rules of the game and also satisfy the following equation:

$$S_{(R,p)} = \max_{p \in \text{strategy}} \left(T_{(R,p)} + U_{(R,p)} \right), \tag{16}$$

where $T_{(R,p)}$ presents the relative defense revenue, $U_{(R,p)}$ represents the upper limit of the confidence interval of $T_{(R,p)}$, and $U_{(R,p)}$ is calculated as the following equation:

$$U_{(R,p)} = c \times \sqrt{\frac{\ln\left(\sum_i n_i\right)}{n_j}}, \tag{17}$$

where c is the priority probability stored in the strategy branch, n_j is the number of times the action p_j has been executed, and $\sum_i n_i$ is the total number of the exploration policy so far. Parameter c can be selected by the expert knowledge in the actual process, the larger the c, the more attention will be paid to the nodes with relatively few visits [26].

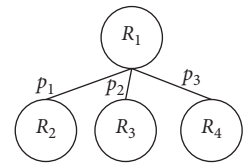

Figure 3: The state tree diagram of the MCTS algorithm.

It can be seen that $U_{(R,p)}$ can measure the degree of policy exploration as the degree of uncertainty of $T_{(R,p)}$. The addition of $U_{(R,p)}$ can improve the situation of the simple greedy policy which is easy to fall into the local minimum. [27].

Step 2. Expansion. In order to parallelize the algorithm, we modify the expansion stages as the way proposed in ref [28]. After selecting the attack strategy by $S_{(R,p)}$, we expand the node into m random children rather than a single child. In this paper, we assume that the defense resources of the defender can be quantified as a nonnegative integer. Since the defender has at most resource + 1 methods of defense force deployment, m is less than resource + 1.

Step 3. Simulation. Selection and expansion are the simulation process, and the simulation process must follow the rules of the game. There are two simulation end conditions: one is that the simulation reaches the leaf node and the other is that the simulation reaches the end state of the strategy and cannot be expanded. In order to parallelize the algorithm, we modify the simulation stages as the way proposed in ref [28] too. Rather than simulating out the child state only once, we simulate each child state k times. Here, k is a parameter that can be determined using m and N. N is the number of nodes in the cluster. k is at least N/m. This is to ensure that every node in the cluster is occupied during the simulation stage.

Step 4. Backpropagation. After simulation process is completed, the parameters of all edges in the simulation path must be updated. This process can reflect how the Monte Carlo tree search samples stronger strategic actions. The update method of a single simulation process is as the following equations:

$$N_{(R,p)} = N_{(R,p)} + 1, \tag{18}$$

$$\text{Revenue}_{(R,p)} = \text{Revenue}_{(R,p)} + \text{Revenue}^{*}_{(R,p)}, \tag{19}$$

$$T_{(R,p)} = -Q_{(R,p)} = -\frac{\text{Revenue}_{(R,p)}}{N_{(R,p)}} + \beta. \tag{20}$$

Among them, the formula updates two related variables, the number of visits of each edge is increased by 1, and the attack revenue is accumulated $\text{Revenue}^*_{(R,p)}$. $\text{Revenue}^*_{(R,p)}$ represents the increase in the cumulative attack revenue during the expansion process and finally calculates the new $T_{(R,p)}$.

Through the above steps, it can be concluded that as the number of sampling increases, the Monte Carlo tree grows and the coverage state becomes more. The shape of the final tree is usually unbalanced. Some states are searched very deep and some are very shallow. This also reflects the advantages of Monte Carlo tree search. The most potential branches will be fully searched to a very deep level.

At this point, the Monte Carlo tree search has been sampled, and the information of all edges has been updated. Based on the finally formed Monte Carlo tree, the defender can make a defense strategy for the actual attack action of the current situation.

5.3. Algorithm Analysis. The algorithm complexity can be simply expressed as $O(mkI/C)$ where m and k are the same as Section 5.2, I is the number of iterations, and C is the number of cores available [28].

6. Experiment and Analysis

According to the network game rules, a simulation experiment is designed to test and analyze the effectiveness and applicability of the model and algorithm proposed in this paper.

6.1. Network Attack-Defense Environment

6.1.1. Topological Structure Description. This paper assumes that there is a typical network topology as shown in Figure 4. In the beginning, the attacker penetrates the internal network from the host v_0, and the defender protects network information system $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}, v_{12}, v_{13}\}$, including 3 desktops, 4 laptops, 2 printers, and 4 servers. It is divided into 4 different subnets, and the target of the attacker is v_{13}.

6.1.2. Formal Description of the Optimal Defense Strategy. A formal description of the attack-defense strategy is carried out according to the method in Section 3. Initially, the attacker has the authority of node v_0 and launches an attack on the network system V. The final attack target is v_{13}.

The network topology matrix A is expressed as follows:

$$A_{[13\times13]} = \begin{bmatrix} 1&0&1&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&1&0&0&0&0&0&0&0&0&0 \\ 1&1&1&1&1&0&0&0&0&0&0&0&0 \\ 0&1&1&1&0&1&0&0&0&0&0&0&0 \\ 0&0&1&0&1&1&1&0&0&0&0&0&0 \\ 0&0&0&1&1&1&1&0&1&0&0&0&0 \\ 0&0&0&0&1&1&1&1&0&0&0&0&0 \\ 0&0&0&0&0&0&1&1&1&1&0&0&0 \\ 0&0&0&0&0&1&0&1&1&1&1&1&0 \\ 0&0&0&0&0&0&0&1&1&1&0&0&1 \\ 0&0&0&0&0&0&0&0&1&0&1&1&1 \\ 0&0&0&0&0&0&0&0&1&0&1&1&1 \\ 0&0&0&0&0&0&0&0&0&1&1&1&1 \end{bmatrix}. \tag{21}$$

The formal expression of the attacker's target is as follows: $\overrightarrow{T} = (0,0,0,0,0,0,0,0,0,0,0,0,-1)$, which means that the attacker obtains the permission of the target v_{13}. The initial network attack reachable node is formally expressed as follows: $\overrightarrow{R} = (1,1,0,0,0,0,0,0,0,0,0,0,0)$, which means that the attack can attack v_1 and v_2 nodes at the first.

In summary, based on the network shown in Figure 4, the optimal defense strategy selection problem can be described as follows: on the certain attack-defense resources, based on the rules of the wargame model, the given network structure is $A_{[13\times13]}$, an initial reachable node vector $\overrightarrow{R}$, and initial attack-defense situation vector $\overrightarrow{S} = 1$, the optimal defense strategy for each attack strategy is found.

6.2. Description of Attack-Defense Resources and Attack-Defense Strategies. For the convenience of calculation, $\text{resource}_A = \text{resource}_D = 10$ is set at the beginning of the game, which means that the resources of the attacker and the defender are both 10 at first.

In this simulation experiment, when the attacker obtains the authority of node v with a certain cost, it has the highest root privilege.

Give an example of the description of the attack-defense strategy in the game. When the attacker obtains the privilege of v_3 through v_1 in subnet A, $\text{resource}_A = 6$, the next attack can reach the node vector: $\overrightarrow{R} = (0,1,0,1,1,0,0,0,0,0,0,0,0)$. At this time, the attacker has 21 types of attack-defense strategies. Among them, there are a total of 3 attack nodes $\{v_2, v_4, v_5\}$ to choose from and then the attack power chooses on the determined node in the set as follows:$\text{cost}_A = \{0,1,2,3,4,5,6\}$.

In the above example, the attacker can adopt a set of action strategies as follows:

$$S_A^k = \left(S_A^k(v_2), S_A^k(v_4), S_A^k(v_5)\right), \tag{22}$$

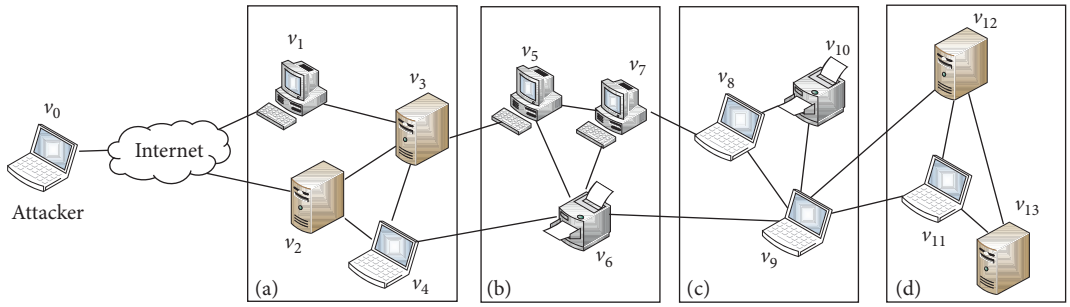

Figure 4: Network topology structure diagram.

where $S_A^k(v_2) = (0, \text{cost}_A, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$,

$$\begin{aligned} S_A^k(v_4) &= (0, 0, 0, \text{cost}_A, 0, 0, 0, 0, 0, 0, 0, 0, 0), \\ S_A^k(v_5) &= (0, 0, 0, 0, \text{cost}_A, 0, 0, 0, 0, 0, 0, 0, 0). \end{aligned} \tag{23}$$

At the same time, the defender adopts a set of actions as follows:

$$S_D^k = S_D^k(c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9, c_{10}, c_{11}, c_{12}, c_{13}), \tag{24}$$

which satisfies $\sum_{i=0}^{10} c_i = 10$ and $c_i \in \{0, 1, 2, \ldots, 10\}$.

6.3. Defense Strategy Selection. In this example, we set two typical attack-defense states that defenders must pay more attention to during the attack-defense game, that is, the beginning of the game and the end of the game. The selection of the defense strategy is discussed based on the abovementioned state using the method proposed in this paper.

6.3.1. At the Beginning of the Game. The following discusses the defensive force distribution of the defender at the beginning of the attack.

The defender firstly allocates the defense resource for each node in the network. There are tens of thousands of defense resource deployment plans for the abovementioned network. Analyzing the topological diagram of the network structure shown in Figure 4, it can be seen that node v_9 is a necessary node from subnet C to subnet D whose authority attribution plays a key role in the network attack-defense. Combined with the network topology, this paper selects the following four typical game initial states for analysis.

Strategy 1: $S_D^k(1) = S_D^k(5, 5, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$

Strategy 2: $S_D^k(2) = S_D^k(0, 0, 0, 0, 0, 0, 0, 0, 0, 10, 0, 0, 0)$

Strategy 3: $S_D^k(3) = S_D^k(3, 3, 0, 0, 0, 0, 0, 0, 4, 0, 0, 0, 0)$

Strategy 4: $S_D^k(4) = S_D^k(1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1)$

Considering both the path revenue and the target node revenue of the attacker, the strategy is evaluated according to the relative defense revenue. The larger the relative defense revenue, the better the defense effect of the strategy.

The above four typical strategies are analyzed by relative defense revenue under different simulation times. As shown in Table 2, rows indicate the number of iterations and columns indicate different strategies.

As shown in Table 2, we can see that when the number of simulations increases, the score of relative defense revenue of the defense strategy gradually stabilizes. We can distinguish optimal strategies, among them $T_{S_D^k(3)} > T_{S_D^k(1)} > T_{S_D^k(2)} > T_{S_D^k(4)}$. According to the simulation results, we analyze the four strategies in detail.

Strategy 1. This strategy is to pay attention only to the network entry node.

$$S_D^k(1) = S_D^k(5, 5, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0). \tag{25}$$

In this strategy, we concentrate the defense resource on the two entrance nodes, and each node deploys 5 resources. During the first attack, the attacker has an initial attack resource of 10, so the probability of the attacker entering the intranet is 50%. Suppose the attacker first attacks node v_1, if the attack is successful, the attacker can drive straight into the network and obtain the permission of the node v_{13} directly. At the same time, the gains on the attack path are relatively large, $\text{revenue}_A \geq 8$ making the relative defense revenue at a medium level.

Strategy 2. This strategy is to pay attention only to the key nodes of the network.

$$S_D^k(2) = S_D^k(0, 0, 0, 0, 0, 0, 0, 0, 0, 10, 0, 0, 0, 0). \tag{26}$$

In this strategy, we concentrate all the defense resources on the key node v_9. During the attack, the attacker can successfully obtain the following 9 nodes $\{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_{10}\}$ due to the attack resource $\text{resource}_A \leq \text{cost}_D$, yet the attacker cannot get the permission of v_9 and thus cannot enter the subnet D. This ensures the security of the target node v_{13} but let the attacker process gain more as $\text{revenue}_A = 8$. It is not acceptable that this strategy protects the target but loses a lot of node permission along the path. If the security of the important data server is ensured yet network devices such as computers and printers are all implanted with Trojan horses or file encryption by the attacker, the company still cannot operate normally. So, in our opinion, this strategy is not a good strategy. This result is in line with our simulation result, which shows the effectiveness of the wargame model.

Table 2: The relative defense revenue of different strategies.

	50	500	5 k	10 k
Strategy 1	7.83	7.60	7.23	7.12
Strategy 2	4.02	5.32	5.53	5.80
Strategy 3	6.53	7.83	8.14	8.45
Strategy 4	4.30	2.91	2.88	2.12

Strategy 3. This strategy is to pay attention to network entry nodes and key nodes at the same time.

$$S_D^k(3) = S_D^k(3, 3, 0, 0, 0, 0, 0, 0, 4, 0, 0, 0, 0). \quad (27)$$

This strategy is a compromise between the above two strategies. During the attack-defense process, the defender pays attention to the entry node and key nodes at the same time. That is to say, on the way to the target node, the defender balances the defense resource and network structure characteristics, setting two safety protection measures. Data analysis from Table 2 shows that it is the most suitable defense deployment plan. The above-mentioned plan also conforms to our common methods of the network protection, which proves the practicability of the wargame model.

Strategy 4. This strategy is to randomly select and defend 10 nodes on the network and divide the defense resources equally.

$$S_D^k(4) = S_D^k(1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1). \quad (28)$$

This strategy is random. When cost_A is greater than 1, the attacker can attack the node where the defender deploys the defense without being discovered. However, since the defensive resource of the defender is scattered, when the cost of attack is high, it cannot defend effectively.

6.3.2. Near the End of the Game. The following discusses the attack-defense game state and the attack-defense force distribution at the end of the game. At this time, $\text{resource}_A = 5$ and $\text{resource}_D = 10$.

The $\text{state}_{\text{finish}}$ means that the attacker has obtained the node of v_8 in subnet C and will launch an attack on v_9. Since the attacker has used 5 attack resources when reaching the node v_8, he can use not more than 5 attack resources in v_9. It is very important that node v_9 is the only entry node for subnet C to subnet D. To ensure the safety of the target node v_{13}, the defense deployment of the defender in v_9 is shown in Figure 5.

In this state, when the attacker uses $\text{cost}_A = 2$ to attack the defender, the target node is effectively protected if the defense cost $\text{cost}_D \in [2, 10]$, and when the attacker uses $\text{cost}_A = 3$ to attack the defender, the target node is effectively protected if the defense cost $\text{cost}_D \in [3, 10]$. As a result, when the defender discovers that the attacker is approaching the target node, the defense should be concentrated on the key node entering the subnet entrance.

Through the abovementioned analysis, it can be concluded that the defender should pay more attention to the entry nodes and key nodes at the beginning of the game,

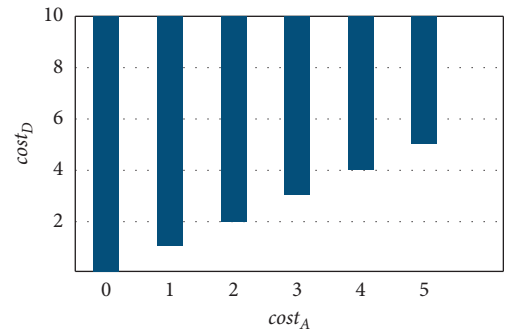

Figure 5: The cost of defense deployment of defenders in v_9.

while focusing on the key node entering the last subnet entrance when the attacker approaches the large node. The abovementioned strategy of the typical states also conforms to our common defense strategy in network protection, which proves the practicability of the model and algorithm in selecting the optimal defense strategy.

7. Related Work

This section will compare our work with related work, as game models, defense strategy, the rules of the game, and so on.

Liu et al. [29] used the state attack-defense map in combination with the security vulnerability assessment system to calculate and modeled the utility matrix. The optimal attack-defense decision was made by calculating the mixed strategy Nash equilibrium. However, since this defense strategy corresponded to the attack strategy one by one, it did not consider all possible defense strategies. Tosh et al. [30] proposed an evolutionary game model framework for cyber threat intelligence sharing. However, there existed excessive subjectivity in the quantitative calculation of the cost of attack-defense strategies. Lin et al. [31] proposed a full-information dynamic active defense game model by converting the "virtual node" into a game tree. Despite giving an algorithm for the game of attack-defense that suits two scenarios of complete and incomplete information, the algorithm did not fully consider the attacker's intentions. Zhang et al. [32] proposed the heterogeneous population evolutionary game model and then the decision method of network security defense and improved the accuracy of network security defense decision.

The comparison of the characteristics of this paper and related research is shown in Table 3.

Compared with the related work, the defense strategy selection method based on the network attack-defense wargame model has the following characteristics:

Table 3: Comparison of features among our method and others.

Features	Ref [29]	Ref [30]	Ref [31]	Ref [32]	Our method
New attack-defense model	✓	✓	✓	✓	✓
Support multiturn system	×	×	✓	×	✓
Support redistribution of attack-defense resources	×	×	×	×	✓
Use artificial intelligence methods	×	✓	×	✓	✓

(i) We model high-level network attack-defense confrontation as a turn-based wargame in which both attackers and defenders can continuously adjust their strategies in response to attack-defense posture.

(ii) We add a higher and stronger defense strategy, by which the defender can redistribute defense resources when discovered that the target node is attacked.

(iii) We propose the wargame model as a multistate dynamic attack-defense game.

(iv) We use Monte Carlo tree search method to solve the optimal defense strategy.

8. Conclusions and Future Work

This paper proposed a defense strategy selection method based on the network attack-defense wargame model. We modeled the high-level network attack-defense confrontation process as a turn-based wargame in which both attackers and defenders could continuously adjust their strategies in response to the attack-defense posture. Based on the idea of artificial intelligence, we used the Monte Carlo tree search method to solve the optimal defense strategy in the game confrontation environment. Finally, a simulation model is designed to analyze the attack-defense process of the target network with rapid modeling and quantitative calculation.

Our future research will focus on the following 3 points. Firstly, stronger network attack-defense strategies will be added to the attack-defense wargame model. Secondly, the impact of defense deployment will be analyzed in key sections game. Thirdly, the applicability of the attack-defense game model will be improved with distributed coordinated attack-defense considered.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Key R&D Program of China Nos. 2019QY1302 and 2019QY1305.

References

[1] Y. Wu, G. Feng, N. Wang, and H. Liang, "Game of information security investment: impact of attack types and network vulnerability," *Expert Systems with Applications*, vol. 42, no. 15-16, pp. 6132–6146, 2015.

[2] E. M. Hutchins, M. J. Cloppert, and R. M. Amin, "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains," *Leading Issues in Information Warfare &Security Research*, vol. 1, no. 1, pp. 112–125, 2011.

[3] D. Caltagirone, A. Pendergast, and C. Betz, *The Diamond Model of Intrusion Analysis*, Center for Cyber Intelligence Analysis and Treat Research, Hanover, MD, USA, 2013.

[4] S. Mishra, A. K. Sangaiah, M. Narayan Sahoo, and S. Bakshi, "Pareto-optimal cost optimization for large scale cloud systems using a joint allocation of resources," *Journal of Ambient Intelligence and Humanized Computing*, pp. 352–370, 2019.

[5] J. Yang, H. Zhang, and C. Zhang, "Network defense decision-making method based on stochastic game and improved WoLF-PHC," *Journal of Computer Research and Development*, vol. 56, no. 5, pp. 942–954, 2019.

[6] X. Liu, H. Zhang, Y. Zhang, L. Shao, and J. Han, "Active defense strategy selection method based on two-way signaling game," *Security and Communication Networks*, vol. 2019, pp. 1–14, Article ID 1362964, 2019.

[7] K. X. Huang, C. J. Zhou, Y. Q. Qin, and W. X. Tu, "A game-theoretic approach to cross-layer security decision-making in industrial cyber-physical systems," *IEEE Transactions on Industrial Electronics*, vol. 3, no. 67, pp. 2371–2379, 2020.

[8] H. Hu, *Research on Network Security Situational Awareness Method Based on Attack Graph*, State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou, China, 2018.

[9] W. Lee, W. Fan, M. Miller, S. J. Stolfo, and E. Zadok, "Toward cost-sensitive modeling for intrusion detection and response," *Journal of Computer Security*, vol. 10, no. 1-2, pp. 5–22, 2002.

[10] Y. Li, D. E. Quevedo, S. Dey, and L. Shi, "A game-theoretic approach to fake-acknowledgment attack on cyber-physical systems," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 3, no. 1, pp. 1–11, 2017.

[11] E. Serra, S. Jajodia, A. Pugliese, A. Rullo, and V. S. Subrahmaniyan, "Pareto-optimal adversarial defense of enterprise systems," *ACM Transactions on Information and System Security*, vol. 17, no. 6, 2015.

[12] M. Esmalifalak, G. Shi, Z. Han, and L. Song, "Bad data injection attack and defense in electricity market using game theory study," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 160–169, 2013.

[13] J. Wu, K. Ota, M. Dong, J. Li, and H. Wang, "Big data analysis-based security situational awareness for smart grid," *IEEE Transactions on Big Data*, vol. 4, pp. 408–417, 2016.

[14] X. Liu, J. Huang, Y. Yao, C. Qi, and G. Zong, "Defending pollution attacks in network coding enabled wireless ad hoc

networks: a game-theoretic framework," *IET Communications*, vol. 19, no. 14, 2020.

[15] D. Wang, Q. Cao, H. Xu, and S. Shen, "Stochastic evolutionary trust strategy of WSNs nodes based on wright-Fisher process," *Computer Applications and Software*, vol. 34, no. 1, pp. 110–115, 2017.

[16] R. Na, G. Lei, Z. Haojin, J. Weijia, L. Xiang, and H. Qi, "Toward optimal DoS-resistant authentication in crowd-sensing networks via evolutionary game," in *Proceedings of the 2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS)*, pp. 364–373, IEEE, Nara, Japan, August 2016.

[17] A. A. A. Abass, L. Xiao, N. Mandayam, and Z. Gajic, "Evolutionary game theoretic analysis of advanced persistent threats against cloud storage," *IEEE Access*, vol. 1, no. 1, 2017.

[18] Y. Hayel and Q. Zhu, "Epidemic protection over heterogeneous networks using evolutionary Poisson games," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 8, pp. 1786–1800, 2017.

[19] M. Xian, D. Sun, S. Du, C. Wang, F. Zeng, and C. Zang, "Practical research on games of incomplete information," *Academic Working Committee of China Institute of Communications*, vol. 4, pp. 299–302, 2013.

[20] Exploit Price Current Top 5, https://vuldb.com/.1997-2021byvuldb.com.AlldataissharedunderthelicenseCCBY-NC-SA4.0.

[21] C. Wu, X. Li, W. Pan, J. Liu, and J. Wu, "Zero-sum game based optimal secure control under actuator attacks," *IEEE Transactions on Automatic Control*, vol. 66, no. 8, pp. 3773–3780, 2020.

[22] J. F. Nash, "Equilibrium points in N-person games," *Proceedings of the National Academy of Sciences*, vol. 1, no. 36, pp. 48-49.

[23] J. B. Chaslot, M. Winands, and H. Herik, "Parallel monte-carlo tree search," in *Proceedings of the International Conference on Computers & Games Springer-Verlag*, Beijing, China, October 2008.

[24] H. Kato and I. Takeuchi, "Parallel monte-carlo tree search with simulation servers," in *Proceedings of the 2010 International Conference on Technologies and Applications of Artificial Intelligence*, pp. 491–498, IEEE, Kaohsiung, Taiwa, November 2010.

[25] Q. Wang, *Research on Gobang Algorithm Based on Deep Reinforcement Learning*, Master's Thesis, Chongqing University, Chongqing, China, 2019.

[26] R. Coulom, "Efficient selectivity and backup operators in Monte-Carlo tree search," in *International Conference on Computers and Games*, vol. 4630, pp. 345–355, Springer, Berlin, Germany, 2006.

[27] R. Coulom, *Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search," International Conference on Computers and Games*, pp. 72–83, Springer, Berlin, Germany, 2006.

[28] Y. Jin and S. Benjamin, *Monte Carlo Search Tree Report*, Stanford University, Standford, CA, USA.

[29] G. Liu, H. Zhang, and Q. Li, "Network security of optimal attack and defense decision making method based on a game model," *Journal of Nanjing University of Science and Technology*, vol. 38, no. 1, pp. 12–21, 2014.

[30] D. Tosh, S. Sengupta, C. Kamhoua, K. Kwait, and A. Martin, "An evolutionary game-theoretic framework for cyber-threat information sharing," in *Proceedings of the International Conference on Communications*, pp. 7341–7346, IEEE, 2015.

[31] W. Q. Lin, H. Wang, J. H. Liu et al., "Research on active defense technology in network security based on non-cooperative dynamic game theory," *Journal of Computer Research and Development*, vol. 2, no. 48, pp. 306–316, 2011.

[32] E. N. Zhang, G. Wang, R. N. Ma, and L. N. Yan, "Network security defense decision-making method using double heterogeneous group evolutionary game," *Journal of Xi'an Jiaotong University*, vol. 8, pp. 1–13, 2021.