Security and Communication Networks

Analytical Models and Decision Making in Adaptive Security

Lead Guest Editor: Akbar S. Namin
Guest Editors: Rakesh M. Verma and Rattikorn Hewett

Analytical Models and Decision Making in Adaptive Security

Security and Communication Networks

Analytical Models and Decision Making in Adaptive Security

Lead Guest Editor: Akbar S. Namin
Guest Editors: Rakesh M. Verma and Rattikorn Hewett

Chief Editor

Associate Editors

Academic Editors

Contents

Hindawi
Security and Communication Networks
Volume 2019, Article ID 3038586, 16 pages
https://doi.org/10.1155/2019/3038586

Research Article

Optimal Decision-Making Approach for Cyber Security Defense Using Game Theory and Intelligent Learning

Yuchen Zhang and Jing Liu

Zhengzhou Information Science and Technology Institute, Zhengzhou 450001, China

Correspondence should be addressed to Yuchen Zhang; 2744190810@qq.com

Received 28 June 2019; Revised 22 September 2019; Accepted 25 October 2019; Published 23 December 2019

Guest Editor: Akbar S. Namin

Existing approaches of cyber attack-defense analysis based on stochastic game adopts the assumption of complete rationality, but in the actual cyber attack-defense, it is difficult for both sides of attacker and defender to meet the high requirement of complete rationality. For this aim, the influence of bounded rationality on attack-defense stochastic game is analyzed. We construct a stochastic game model. Aiming at the problem of state explosion when the number of network nodes increases, we design the attack-defense graph to compress the state space and extract network states and defense strategies. On this basis, the intelligent learning algorithm WoLF-PHC is introduced to carry out strategy learning and improvement. Then, the defense decision-making algorithm with online learning ability is designed, which helps to select the optimal defense strategy with the maximum payoff from the candidate strategy set. The obtained strategy is superior to previous evolutionary equilibrium strategy because it does not rely on prior data. By introducing eligibility trace to improve WoLF-PHC, the learning speed is further improved and the defense timeliness is significantly promoted.

1. Introduction

With the continuous strengthening of social informatization, cyber attacks are becoming more frequent, causing tremendous losses to defenders [1]. Because of the complexity of the network itself and the limitation of the defender's ability, the network cannot achieve absolute security. It is urgent to have a technology which can analyze the attack-defense behavior and effectively compromise the network risk and security investment so that the defender can make reasonable decisions with limited resources. Game theory and cyber attack-defense have a high degree of opposition, non-cooperative relationship, and strategic dependence [2]. The research and application of game theory in cyber security are rising day by day [3]. The analysis of attack-defense confrontation based on stochastic game has become a hotspot. Stochastic game is a combination of game theory and Markov decision making. It not only extends the single state of traditional game to multistate but also characterizes the randomness of cyber attack-defense. At present, the cyber security analysis based on stochastic game has achieved some results, but there are still some shortcomings and challenges [4–7]. The existing stochastic game of attack-defense is based on the assumption of complete rationality, through Nash equilibrium for attack prediction and defense guidance. Complete rationality includes many aspects of perfection requirements, such as rational consciousness (pursuit of maximum benefits), analytical reasoning ability, identification and judgment ability, memory ability, and accurate behavior ability, among which any aspect of imperfection belongs to limited rationality [8]. The high requirement of complete rationality is too harsh for both sides of attack-defense, which makes it difficult for Nash equilibrium under the assumption of complete rationality to appear in practice, and reduces the accuracy and guiding value of existing research results.

To solve the above problems, this paper studies the defense decision-making approach based on stochastic game under the restriction of bounded rationality. Section 2 introduces the research status of defense decision making based on stochastic game. Section 3 analyses the difficulties of studying cyber attack-defense stochastic game under bounded rationality and the idea of solving the problem in this paper. Moreover, Section 3 constructs attack-defense stochastic game model under bounded rationality constraints and proposes a host-centered attack-defense graph model to extract network state and attack-defense action in game model. Bowling et al. [9] first proposed WoLF-PHC for multiagent learning, Section 4 further improves WoLF-PHC algorithm based on eligibility trace for promoting the learning speed of defenders as well as reducing the dependence of the algorithm on data. Using the improved intelligent learning algorithm WoLF-PHC (Wolf Mountain Climbing Strategy) to analyze the stochastic game model in the previous section, we design the defense decision-making algorithm. Section 5 verifies the effectiveness of the proposed approach through experiments. Section 6 summarizes the full text and discusses future research.

There are three main contributions of this paper:

(1) The extraction of network state and attack-defense actions is one of the keys to the construction of stochastic game model. The network state of the existing stochastic game model contains the security elements of all nodes in the network, and there is a "state explosion" problem. In order to solve this problem, a host-centered attack-defense graph model is proposed and an attack-defense graph generation algorithm is designed, which effectively compresses the game state space.

(2) Limited rationality means that both sides of attack-defense need to find the optimal strategy through trial and error and learning. It is a key point to determine the learning mechanism of players. In this paper, reinforcement learning is introduced into stochastic game, which expands stochastic game from complete rationality to limited rationality. Defenders use WoLF-PHC to learn the game in adversarial attack-defense so as to make the best choice for current attackers. Most of the existing bounded rationality games use biological evolutionary mechanism to learn and take the group as the research object. Compared with the existing bounded rationality games, the approach proposed in this paper reduces the exchange of information among game players and is more suitable for guiding individual defense decision making.

(3) WoLF-PHC algorithm is improved based on eligibility trace [10], which speeds up the learning speed of defenders, reduces the dependence of the algorithm on data, and proves the effectiveness of the approach through experiments.

2. Related Works

Some progress has been made in cyber security research based on game theory at home and abroad, but most of the current studies are based on the assumption of complete rationality [11]. Under complete rationality, according to the decision-making times of both sides in the game process, it can be divided into single-stage game and multistage game. The research of single-stage cyber attack-defense game started earlier. Liu et al. [12] used static game theory to analyze the effectiveness of worm virus attack-defense strategy. Li et al. [13] established a non-cooperative game model between attackers and sensor trust nodes and gave the optimal attack strategy based on Nash equilibrium. In cyber attack-defense, although part of the simple attack-defense confrontation belongs to single-stage game, in most scenarios, the process of attack-defense often lasts for many stages, so multistage cyber attack-defense game becomes a trend. Zhang et al. [14] regarded the defender as the source of signal transmission and the attacker as the receiver and constructed a multistage attack-defense process using differential game. Afrand and Das [15] established a repeated game model between the intrusion detection system and wireless sensor nodes and analyzed the forwarding strategy of node packets. Although the above results can be used to analyze multistage attack-defense confrontation, the state transition between stages is not only affected by attack-defense action, but also by the interference of system operating environment and other external factors, which has randomness. The above results ignore this randomness and weaken its guiding value.

Stochastic game is a combination of game theory and Markov theory. It is a multistage game model. It can accurately analyze the impact of randomness on attack-defense process by using the Markov process to describe the state transition. Wei et al. [16] abstracted the cyber attack-defense as a stochastic game problem and gave a more scientific and accurate quantitative approach of attack-defense benefits applicable to the stochastic game model of attack-defense. Wang et al. [17] used stochastic game theory to study the network confrontation problem. Convex analysis theory was used to prove the existence of equilibrium, and the equilibrium solution was transformed into a nonlinear programming problem. Based on incomplete information stochastic game, Liu et al. [3] proposed the decision-making approach for moving targets defense. All the aforementioned schemes are based on the assumption of complete rationality, which is too strict for both sides of attack-defense. In most cases, both sides of attack-defense are only limited rationality level, which leads to the deviation of the above research results in the analysis of attack-defense game. Therefore, it has important research value and practical significance to explore the bounded rationality of cyber attack-defense game law.

Limited rationality means that both sides of attack-defense will not find the optimal strategy at the beginning. They will learn the game of attack-defense in the game of attack-defense. The appropriate learning mechanism is the key to win in the game. At present, the research of limited rational attack-defense game is mainly centered on evolutionary game [18]. Hayel and Zhu [19] established an evolutionary Poisson game model between malicious software and antivirus programs and used the replication dynamic equation to analyze the antivirus program strategy. Huang and Zhang [20] improved the traditional replication dynamic equation by introducing incentive coefficient and improved the calculation approach of replication dynamic rate. Based on this, an evolutionary game model was constructed for defense. Evolutionary game takes the group as the research object, adopts the biological evolution mechanism, and completes the learning by imitating the advantage strategy of other members. In evolutionary game, there is too much information exchange among players, and it mainly studies the adjustment process, trend, and stability of attack-defense group strategy, which is not conducive to guiding the real-time strategy selection of individual members.

Reinforcement learning is a classic online intelligent learning approach. Its players learn independently through environmental feedback. Compared with evolutionary biology, reinforcement learning is more suitable for guiding individual decision making. This paper introduces reinforcement-learning mechanism into stochastic game, expands stochastic game from complete rationality to finite rationality, and uses bounded rationality stochastic game to analyze cyber attack-defense. On the one hand, compared with the existing attack-defense stochastic game, this approach uses bounded rationality hypothesis, which is more realistic. On the other hand, compared with evolutionary game, this approach uses reinforcement learning mechanism, which is more suitable for guiding real-time defense decision making.

3. Modeling of Attack-Defense Confrontation Using Stochastic Game Theory

3.1. Description and Analysis of Cyber Attack-Defense Confrontation. Cyber attack-defense confrontation is a complex problem, but from the level of strategy selection, it can be described as a stochastic game problem as depicted in Figure 1. We take the DDoS attack of using Sadmind vulnerability of Soloris platform as an example. The attack is implemented through multiple steps including *IP sweep*, *Sadmind ping*, *Sadmind exploit*, *Installing DDoS software*, and *Conducting DDoS attack*. Each attack step can lead to change of security state of network.

Taking the first step as an example, the initial network state is denoted as S_0 (H_1, none). It means that the attacker *Alice* does not have any privileges of host H_1. Then, attacker *Alice* implemented an *IP sweep* attack on H_1 through its open port 445 and gained the User privilege of H_1. This network state is denoted as S_1 (H_1, User). Afterwards, if the defender *Bob* selected and implemented a defense strategy from the candidate strategy set {*Reinstall Listener program*, *Install patches*, *Close unused port*}, then the network state is transferred back to S_0; otherwise, the network may continue to evolve to another more dangerous state S_3.

The continuous time axis is divided into time slices, and each time slice contains only one network state. The network state may be the same in different time slices. Each time slice is a game of attack-defense. Both sides detect the current network state, then select the attack-defense actions according to the strategy, and get immediate returns. Attack-defense strategies are related to network state. The network system transfers from one state to another under the candidate action of the attacking and defending sides. The transition between network states is not only affected by attack-defense actions but also by factors such as system operating environment and external environment, which is random. The goal of this paper is to enable defenders to obtain higher long-term benefits in attack-defense stochastic game.

Both sides of attack-defense can predict the existence of Nash equilibrium, so Nash equilibrium is the best strategy for both sides. From the description of complete rationality in the introduction, we can see that the requirement of complete rationality for both sides of attack-defense is too strict, and both sides of attacker and defender will be constrained by limited rationality in practice. Limited rationality means that at least one of the attacking and defending sides will not adopt Nash equilibrium strategy at the beginning of the game, which means that it is difficult for both sides to find the optimal strategy in the early stage of the game, and they need to constantly adjust and improve the strategy for their opponents. It means that the game equilibrium is not the result of one choice but that both sides of the attack-defense sides are constantly learning to achieve in the course of the attack-defense confrontation and because of the influence of learning mechanism may deviate again even if it reaches equilibrium.

From the above analysis, we can see that learning mechanism is the key to win the game of limited rationality. For defense decision making, the learning mechanism of attack-defense stochastic game under bounded rationality needs to satisfy the following two requirements: (1) Convergence of learning algorithm: attacker strategy under bounded rationality has dynamic change characteristics, and because of the interdependence of attack-defense strategy, the defender must learn the corresponding optimal strategy when facing different attack strategies to ensure that he is invincible. (2) The learning process does not need too much attacker information: both sides of the cyber attack-defense have opposition of objectives and non-cooperation, and both sides will deliberately hide their key information. If too much opponent information is needed in the learning process, the practicability of the learning algorithm will be reduced.

WoLF-PHC algorithm is a typical strategy gradient intelligent learning approach, which enables defenders to learn through network feedback without too much information exchange with attackers. The introduction of WoLF mechanism ensures the convergence of WoLF-PHC algorithm [9]. After the attacker learns to adopt Nash equilibrium strategy, WoLF mechanism enables the defender to converge to the corresponding Nash equilibrium strategy, while the attacker has not yet learned Nash equilibrium strategy, and WoLF mechanism enables the defender to converge to the corresponding optimal defense strategy. In conclusion, WoLF-PHC algorithm can meet the demand of attack-defense stochastic game under bounded rationality.

3.2. Stochastic Game Model for Attack-Defense. The mapping relationship between cyber attack-defense and stochastic

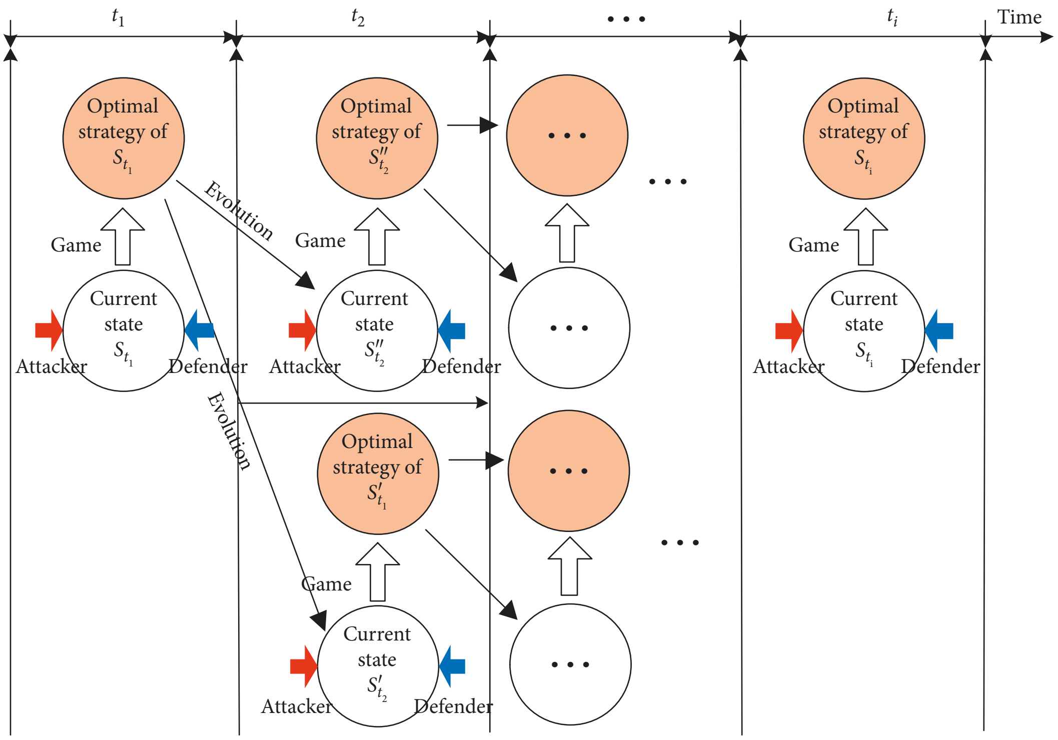


Figure 1: The game process and strategy selection of attack-defense confrontation.

game model is depicted in Figure 2. Stochastic game consists of attack-defense game in each state and transition model between states. The two key elements of "information" and "game order" are assumed. Constrained by bounded rationality, the attacker's historical actions and the attacker's payoff function are set as the attacker's private information.

Herein, we use the above example in Figure 1 to explain Figure 2; the security state corresponds to S_0 (H_1, none) and S_1 (H_1, User) in this case. The candidate strategy set against DDoS attack is {*Reinstall Listener program, Install patches, Close unused port*}. Network state is the common knowledge of both sides. Because of the non-cooperation between the attack and defense sides, the two sides can only observe each other's actions through the detection network, which will delay the execution time for at least one time slice, so the attack-defense sides are acting at the same time in each time slice. The "simultaneous" here is a concept of information rather than a concept of time; that is, the choice of attack-defense sides may not be based on the concept of time. At the same time, because the attack-defense sides do not know the other side's choice when choosing action, they are considered to be simultaneous action.

Construct the network state transition model. Use probability to express the randomness of network state transition. Because the current network state is mainly related to the previous network state, the first-order Markov is used to represent the state transition relationship, in which the network state is the attack-defense action. Because both sides of attacker and defender are constrained by bounded rationality, in order to increase the generality of the model, the transfer probability is set as the unknown information of both sides of attack-defense.

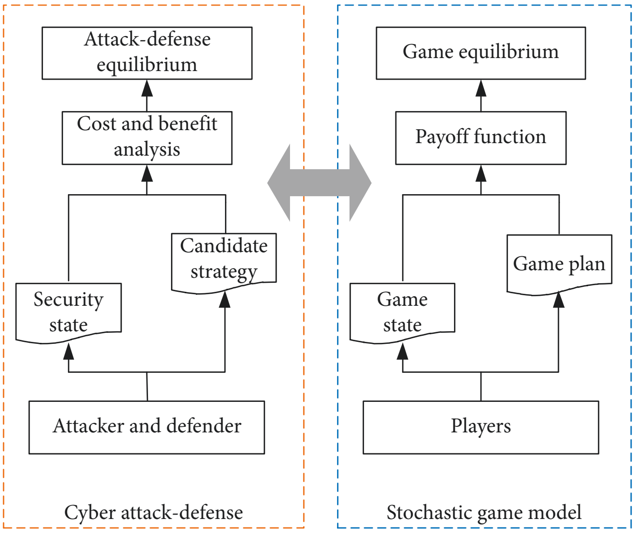


Figure 2: The mapping relationship between cyber attack-defense and stochastic game model.

On the basis of the above, a game model is constructed to solve the defense decision-making problem.

Definition 1. The attack-defense stochastic game model (AD-SGM) is a six-tuple $\mathrm{AD}-\mathrm{SGM}=(N,S,D,R,Q,\pi)$, in which

(1) $N=$ (attacker, defender) are the two players who participate in the game representing cyber attackers and defenders, respectively

(2) $S=(s_1,s_2,\ldots,s_n)$ is a set of stochastic game states, which is composed of network states (see Section 3.3 for the specific meaning and generation approach)

(3) $D = (D_1, D_2, \ldots, D_n)$ is the action set of the defender, in which $D_k = \{d_1, d_2, \ldots, d_m\}$ is the action set of the defender in the game state S_k

(4) $R_d(s_i, d, s_j)$is the immediate return from state s_i to s_j after the defender performs action d.

(5) $Q_d(s_i, d)$ is the state-action payoff function of the defender indicating the expected payoff of the defender after taking action d in the state s_i

(6) $\pi_d(s_k)$ is the defense strategy of the defender in the state s_k

Defense strategy and defense action are two different concepts. Defense strategy is the rule of defense action, not the action itself. For example, $\pi_d(s_k) = (\pi_d(s_k, d_1), \ldots, \pi_d(s_k, d_m))$ is the strategy of the defender in the network state s_k, where $\pi_d(s_k, d_m)$ is the probability of selecting action d_m, $\sum_{d \in D_k} \pi_d(s_k, d) = 1$.

3.3. Network State and Attack-Defense Action Extraction Approach Based on Attack-Defense Graph. Network state and attack-defense action are important components of stochastic game model. Extraction of network state and attack-defense action is a key point in constructing attack-defense stochastic game model [21]. In the current attack-defense stochastic game, when describing the network state, each network state contains the security elements of all nodes in the current network. The number of network states is the power set of security elements, which will produce a state explosion [22]. Therefore, a host-centered attack-defense graph model is proposed. Each state node only describes the host state. It can effectively reduce the size of state nodes [23]. Using this attack-defense graph to extract network state and attack-defense action is more conducive to cyber attack-defense confrontation analysis.

Definition 2. attack-defense graph is a binary group $G = (S, E)$, in which $S = \{s_1, s_2, \ldots, s_n\}$ is a set of node security states and $s_i = \langle \text{host}, \text{privilege} \rangle$, host is the unique identity of the node, and privilege = {none, user, root} indicates that it does not have any privileges, has ordinary user privileges, and has administrator privileges. For directed edge $E = (E_a, E_d)$, it indicates that the occurrence of attack or defense action causes the transfer of node state and $e_k = (s_r, v/d, s_d)$, $k = a, d$, where s_r is the source node and s_d is the destination node.

The generation process of attack-defense map is shown in Figure 3. Firstly, target network scanning is used to acquire cyber security elements, then attack instantiation is combined with attack template, and defense instantiation is combined with defense template. Finally, attack-defense graph is generated. The state set of attack-defense stochastic game model is extracted by attack-defense graph nodes, and the defense action set is extracted by the edge of attack-defense graph.

3.3.1. Elements of Cyber Security. The elements of cyber security NSE are composed of network connection C, vulnerability information V, service information F, and access rights P. Matrix $C \subseteq \text{host} \times \text{host} \times \text{port}$ describes the connection relationship between nodes, the row of matrix represents the source node shost, the list of matrix represents the destination node dhost, and the port access relationship is represented by matrix elements. When port is empty, it indicates that there is no connection relationship between shost nodes and dhost nodes. $V = \langle \text{host}, \text{service}, \text{cveid} \rangle$ indicates the vulnerability of services on nodes' host, including security vulnerabilities and improper configuration or misconfiguration of system software and application software. $F = \langle \text{host}, \text{service} \rangle$ indicates that a service is opened on a node *host*. $P = \langle \text{host}, \text{privilege} \rangle$ indicates that an attacker has access rights privilege on a node host.

3.3.2. Attack Template. Attack template AM is the description of vulnerability utilization, where $\text{AM} = \langle \text{tid}, \text{prec}, \text{postc} \rangle$. Among them, tid is the identification of attack mode; $\text{prec} = \langle P, V, C, F \rangle$ describes the set of prerequisites for an attacker to use a vulnerability, including the initial access rights privilege of the attacker on the source node shost, the vulnerability information cveid of the target node, the network connection relationship C, and the running service of the node F. Only when the set of conditions is satisfied, the attacker can succeed. Use this vulnerability; $\text{postc} = \langle P, C, \text{sd} \rangle$ describes the consequences of an attacker's successful use of a vulnerability, including the increase of attacker's access to the target node, the change of network connection relationship, and service destruction.

3.3.3. Defense Template. Defense templates DM are the response measures taken by defenders after predicting or identifying attacks, where $\text{DM} = \langle \text{tid}, d\text{set} \rangle$. $d\text{set} = \{\langle d_1, \text{post}d_1 \rangle, \ldots, \langle d_m, \text{post}d_m \rangle\}$ is the defense strategy set for specific attacks. $\text{post}d_i = \langle F, V, P, \mathbf{C} \rangle$ describes the impact of defense strategy on cyber security elements, including the impact on node service information, vulnerability information, attacker privilege information, node connection relationship, and so on.

In the process of attack-defense graph generation, if there is a connection between two nodes and all the prerequisites for attack occurring are satisfied, the edges from source node to destination node are added. If the attack changes the security elements such as connectivity, the cyber security elements should be updated in time. If the defense strategy is implemented, the connection between nodes or the existing rights of attackers should be changed. As shown in Algorithm 1, the first step is to use cyber security elements

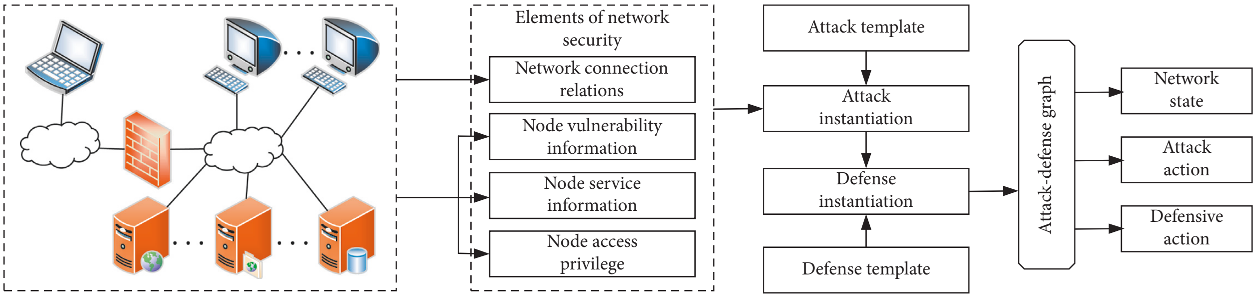


Figure 3: Attack-defense graph generation.

```
     Input: Elements of Cyber security NSE, Attack Template AM, Defense Template DM
     Output: Attack graph $G = (S, E)$
 (1) $S \longleftarrow$ NSE, $E \longleftarrow \varnothing$/* Generate all nodes */
 (2) for each $S$ do:/* Attack instantiation to generate attack edges */
 (3)   update NSE in $s$/* Updating Cyber security Elements */
 (4) if $C$.shost = $s$.host and $C$.dhost.$V$ ≥ AM.prec.$V$ and $C$.dhost.$F$ ≥ AM.prec.$F$ and $C$.dhost.$P$.privilege ≥ AM.prec.$P$.privilege:
 (5)     $s_r$.host $\longleftarrow C$.shost
 (6)     $s_d$.host $\longleftarrow C$.dhost
 (7)     $s_d$.privilege $\longleftarrow$ AM.postc.$P$.privilege
 (8)     $E_a \longleftarrow E_a \cup \{e_a(s_r, \text{AM.tid}, s_d)\}$
 (9)   end if
(10) end for
(11) for each $S$ do:/* Defense instantiation to generate defense edges */
(12)   if $E_a.s_d = s$ and DM.tid = $E_a$.tid:
(13)     $E_d \longleftarrow E_d \cup \{e_d(E_a.s_d, \text{DM.dset}.d, E_a.s_r)\}$
(14)   end if
(15) end for
(16) for each $S$ do:/* Remove isolated nodes S */
(17)   if $e_a(s, \text{tid}, s_d) = \varnothing$ and $e_d(s_r, d, s) = \varnothing$:
(18)     $S \longleftarrow S - s$
(19)   end if
(20) end for
(21) Return $G$
```

Algorithm 1: Attack-defense graph generation algorithm.

to generate all possible state nodes and initialize the edges. Steps 2–8 are to instantiate attacks and generate all attack edges. Steps 9–15 are to instantiate defenses and generate all defense edges. Steps 16–20 are used to remove all isolated nodes. And step 21 is to output attack-defense maps.

Assuming the number of nodes in the target network is n and the number of vulnerabilities of each node is m, the maximum number of nodes in the attack-defense graph is $3n$. In the attack instantiation stage, the computational complexity of analyzing the connection relationship between each two nodes is $o(n^2 - n)$. The computational complexity of matching the vulnerability of the nodes with the connection relationship is $o(m(n^2 - n))$. In the defense instantiation stage, we remove the isolated nodes, and the computational complexity of traversing the edges of all the nodes is $o(9n^2 - 3n)$. In summary, the order of computational complexity of the algorithm is $o(n^2)$. The node of attack-defense graph G can extract network state, and the edge of attack-defense graph G can extract attack-defense action.

4. Stochastic Game Analysis and Strategy Selection Based on WoLF-PHC Intelligent Learning

In the previous section, cyber attack-defense is described as a bounded rational stochastic game problem, and an attack-defense stochastic game model AD-SGM is constructed. In this section, reinforcement learning mechanism is introduced into finite rational stochastic game, and WoLF-PHC algorithm is used to select defense strategies based on AD-SGM.

4.1. Principle of WoLF-PHC

4.1.1. Q-Learning Algorithm. Q-learning [24] is the basis of WoLF-PHC algorithm and a typical model-free reinforcement learning algorithm. Its learning mechanism is shown in Figure 4. Agent in Q-learning obtains knowledge of return and environment state transfer through interaction

```
Input: AD – SGM; $\alpha, \delta, \lambda$, and $\gamma$
Output: Defense action $d$
initialize AD – SGM, $C(s) = 0$, $e(s,d) = 0$/* Network state and attack-defense actions are extracted by Algorithm 1 */
$s^* = \text{get}(E)$ /* Getting the current network state from Network $E$ */
repeat:
    $d^* = \pi_d(s^*)$ /* Select defense action */
    Output $d^*$; /* Feedback defense actions to defenders */
    $s' = \text{get}(E)$ /* Get the status after the action $d^*$ is executed */
    $\rho^* = R_d(s^*, d^*, s') + \gamma \max_{d'} Q_d(s', d') - Q_d(s^*, d^*)$
    $\rho_g = R_d(s^*, d^*, s') + \gamma \max_{d'} Q_d(s', d') - \max_d Q_d(s^*, d)$
    for each state-action pair $(s, d)$ except $(s^*, d^*)$ do:
        $e(s,d) = \gamma\lambda e(s,d)$
        $Q_d(s,d) = Q_d(s,d) + \alpha\rho_g e(s,d)$
    end for/* Update noncurrent eligibility trace $(s^*, d^*)$ and values $Q_d$ */
    $Q_d(s^*, d^*) = Q_d(s^*, d^*) + \alpha\rho^*$/* Update $Q_d$ of $(s^*, d^*)$ */
    $e(s^*, d^*) = \gamma\lambda e(s^*, d^*) + 1$/* Update track of $(s^*, d^*)$ */
    $C(s^*) = C(s^*) + 1$
    Updating average strategy based on formula (6)
    Selecting the learning rate of strategies based on formula (5)
    $\delta_{s^*d} = \min(\pi_d(s^*, d), \delta/(D(s^*) - 1)), \forall d \in D(s^*)$
    $\Delta_{s^*d} = \begin{cases} -\delta_{s^*d} & d \neq \text{argmax}_{d_i} Q_d(s^*, d_i) \\ \sum_{d_j \neq d^*} \delta_{s^*d_j} & \text{Others} \end{cases}$
    $\pi_d(s^*, d) = \pi_d(s^*, d) + \Delta_{s^*d}, \forall d \in D(s^*)$ /* Update defense strategy */
    $s^* = s'$
end repeat
```

Algorithm 2: Defense decision-making algorithm.

with environment. Knowledge is expressed by payoff Q_d and learned by updating Q_d. Q_d is

$$Q_d(s,d) = Q_d(s,d) + \alpha\left[R_d(s,d,s') + \gamma \max_{d'} Q_d(s',d') - Q_d(s,d)\right], \tag{1}$$

where α is payoff learning rate and γ is the discount factor. The strategy of Q-learning is $\pi_d(s) = \text{argmax}_d Q_d(s,d)$.

4.1.2. PHC Algorithm.

The Policy Hill-Climbing algorithm [25] is a simple and practical gradient descent learning algorithm suitable for hybrid strategies, which is an improvement of Q-learning. The state-action gain function Q_d of PHC is the same as Q-learning, but the policy update approach of Q-learning is no longer followed, but the hybrid strategy $\pi_d(s_k)$ is updated by executing the hill-climbing algorithm, as shown in equations (2)–(4). In the formula, the strategy learning rate is

$$\pi_d(s_k, d_i) = \pi_d(s_k, d_i) + \Delta_{s_k d_i}, \tag{2}$$

where

$$\Delta_{s_k d_i} = \begin{cases} -\delta_{s_k d_i}, & d_i \neq \arg\max_d Q_d(s_k, d), \\ \sum_{d_j \neq d_i} \delta_{s_k d_j}, & \text{others}, \end{cases} \tag{3}$$

$$\delta_{s_k d_i} = \min\left(\pi_d(s_k, d_i), \frac{\delta}{|D_k - 1|}\right). \tag{4}$$

4.1.3. WoLF-PHC Algorithm.

WoLF-PHC algorithm is an improvement of PHC algorithm. By introducing WoLF mechanism, the defender has two different strategy learning rates: low strategy learning rate δ_w when winning and high strategy learning rate δ_l when losing, as shown in formula (5). The two learning rates enable defenders to adapt quickly to attackers' strategies when they perform worse than expected and to learn cautiously when they perform better than expected. The most important thing is the introduction of WoLF mechanism, which guarantees the convergence of the algorithm [9]. WoLF-PHC algorithm uses average strategy as the criterion of success and failure, as shown in formulae (6) and (7).

$$\delta = \begin{cases} \delta_w, & \sum_{d \in D_k} \pi_d(s_k, d) Q_d(s_k, d) > \sum_{d \in D_k} \overline{\pi}_d(s_k, d) Q_d(s_k, d), \\ \delta_l, & \text{others}, \end{cases} \tag{5}$$

$$\overline{\pi}_d(s,d) = \overline{\pi}_d(s,d) + \frac{1}{C(s)}\left(\pi_d(s,d) - \overline{\pi}_d(s,d)\right), \tag{6}$$

$$C(s) = C(s) + 1. \tag{7}$$

4.2. Defense Decision-Making Algorithm Based on Improved WoLF-PHC.

The decision-making process of our approach is shown in Figure 5, which consists of five steps. It receives two types of input data: attack evidence and abnormal

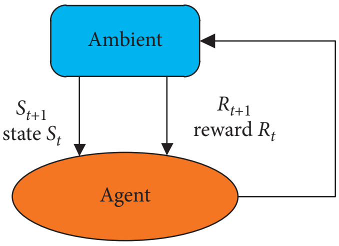


FIGURE 4: Q-learning learning mechanism.

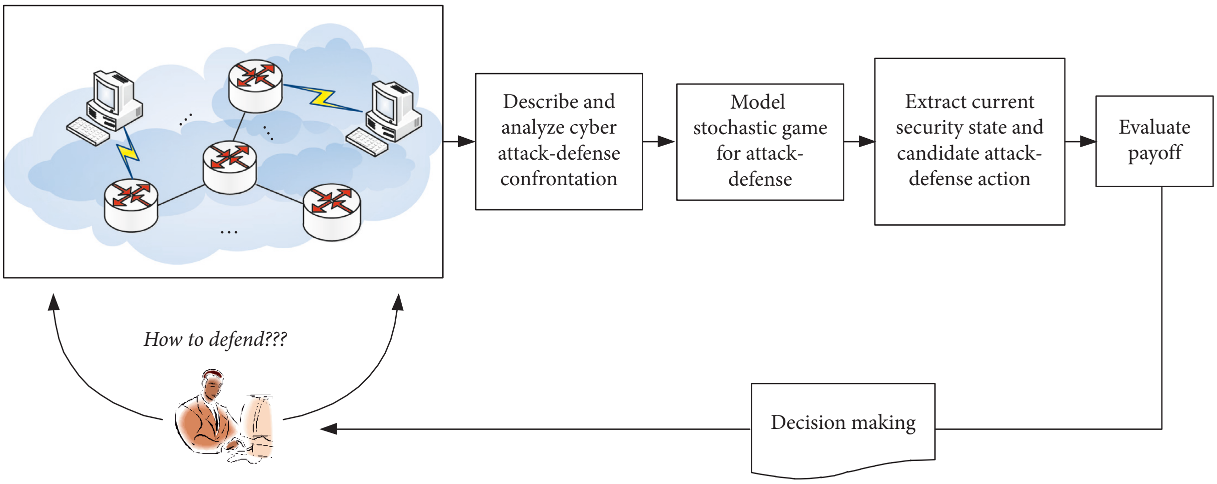

Describe and analyze cyber attack-defense confrontation
Model stochastic game for attack-defense
Extract current security state and candidate attack-defense action
Evaluate payoff
Decision making
How to defend???

FIGURE 5: The process of our decision-making approach.

evidence. All these pieces of evidence come from real-time intrusion detection systems. After decision making, the optimal security strategy is determined against detected intrusions.

In order to improve the learning speed of WoLF-PHC algorithm and reduce the dependence of the algorithm on the amount of data, the eligibility trace is introduced to improve WoLF-PHC. The eligibility trace can track specific state-action trajectories of recent visits and then assign current returns to the state-action of recent visits. WoLF-PHC algorithm is an extension of Q-learning algorithm. At present, there are many algorithms combining Q-learning with eligibility trace. This paper improves WoLF-PHC by using the typical algorithm [10]. The qualification trace of each state-action is defined as $e(s,a)$. Suppose s^* is the current network state s^*, and the eligibility trace is updated in the way shown in formula (8). Among them, the trace attenuation factor is λ.

$$e(s,d) = \begin{cases} \gamma\lambda e(s,d), & s \neq s^*, \\ \gamma\lambda e(s,d) + 1, & s = s^*. \end{cases} \quad (8)$$

WoLF-PHC algorithm is an extension of Q-learning algorithm, which belongs to off-policy algorithm. It uses greedy policy when evaluating defense actions for each network state and occasionally introduces nongreedy policy when choosing to perform defense actions in order to learn. In order to maintain the off-policy characteristics of WoLF-PHC algorithm, the state-action values are updated by formulae (9)–(12), in which the defense actions d^* are selected for execution s^* because only the recently visited status-action pairs will have significantly more eligibility trace than 0, while most other status-action pairs will have almost none eligibility trace. In order to reduce the memory and running time consumption caused by eligibility trace, only the latest status-action pair eligibility trace can be saved and updated in practical application.

$$Q_d(s^*,d^*) = Q_d(s^*,d^*) + \alpha\rho^*, \quad (9)$$

$$Q_d(s,d) = Q_d(s,d) + \alpha\rho_g e(s,d), \quad (10)$$

$$\rho^* = R_d(s^*,d,s') + \gamma \max_{d'} Q_d(s',d') - Q_d(s^*,d^*), \quad (11)$$

$$\rho_g = R_d(s^*,d^*,s') + \gamma \max_{d'} Q_d(s',d') - \max_{d} Q_d(s^*,d). \quad (12)$$

In order to achieve better results, the defense decision-making approach based on WoLF-PHC needs to set four parameters α, δ, λ, and γ reasonably. (1) The range of the payoff learning rate is $0 < \alpha < 1$. The bigger the representative α is, the more important the cumulative reward is. The faster the learning speed is, the smaller α is and the better the stability of the algorithm is. (2) The range of strategy learning rate $0 < \delta < 1$ is obtained. According to the experiment, we can get a better result when adopting $\delta_l/\delta_w = 4$. (3) The attenuation factor λ of eligibility trace is in the range of

$0 < \lambda < 1$, which is responsible for the credit allocation of status-action. It can be regarded as a time scale. The greater the credit allocated λ to historical status-action, the greater the credit allocated to historical status-action. (4) The range of the discount factor $0 < \gamma < 1$ represents the defender's preference for immediate return and future return. When γ approaches 0, it means that future returns are irrelevant and immediate returns are more important. When γ approaches 1, it means immediate returns are irrelevant and future returns are more important.

Agent in WoLF-PHC is the defender in the stochastic game model of attack-defense $AD-SGM$, the game state in agent's state corresponds to $AD-SGM$, the defense action in agent's behavior corresponds to $AD-SGM$, the immediate return in agent's immediate return corresponds to $AD-SGM$, and the defense strategy in agent's strategy corresponds to $AD-SGM$.

On the basis of the above, a specific defense decision-making approach as shown in Algorithm 2 is given. The first step of the algorithm is to initialize the stochastic game model $AD-SGM$ of attack-defense and the related parameters. The network state and attack-defense actions are extracted by Algorithm 1. The second step is to detect the current network state by the defender. Steps 3–22 are to make defense decisions and learn online. Steps 4-5 are to select defense actions according to the current strategy, steps 6–14 are to update the benefits by using eligibility traces, and steps 15–21 are the new payoffs using mountain climbing algorithm to update defense strategy.

The spatial complexity of the Algorithm 2 mainly concentrates on the storage of pairs $R_d(s, d, s')$ such as $e(s, d)$, $\pi_d(s, d)$, $\overline{\pi}_d(s, d)$, and $Q_d(s, d)$. The number of states is $|S|$. $|D|$ and $|A|$ are the numbers of measures taken by the defender and attacker in each state, respectively. The computational complexity of the proposed algorithm is $O(4 \cdot |S| \cdot |D| + |S|^2 \cdot |D|)$. Compared with the recent method using evolutionary game model with complexity $O(|S|^2 \cdot (|A| + |D|)^3)$ [14], we greatly reduce the computational complexity and increase the practicability of the algorithm since the proposed algorithm does not need to solve the game equilibrium.

5. Experimental Analysis

5.1. Experiment Setup. In order to verify the effectiveness of this approach, a typical enterprise network as shown in Figure 6 is built for experiment. Attacks and defenses occur on the intranet, with attackers coming from the extranet. As a defender, network administrator is responsible for the security of intranet. Due to the setting of Firewall 1 and Firewall 2, legal users of the external network can only access the web server, which can access the database server, FTP server, and e-mail server.

The simulation experiment was carried out on a PC with Intel Core i7-6300HQ @3.40 GHz, 32 GB RAM memory, and Windows 10 64 bit operating system. The Python 3.6.5 emulator was installed, and the vulnerability information in the experimental network was scanned by Nessus toolkit as shown in Table 1. The network topology information was collected by ArcGis toolkit. We used the Python language to write the project code. During the experiment, we set up about 25,000 times of attack-defense strategy studies. The experimental results were analyzed and displayed using Matlab2018a as described in Section 5.3.

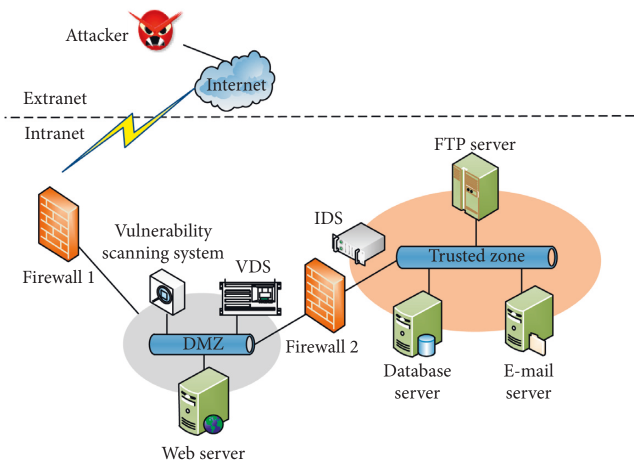


FIGURE 6: Experimental network topology.

TABLE 1: Network vulnerability information.

Attack identifier	Host	CVE	Target privilege
tid_1	Web server	CVE-2015-1635	user
tid_2	Web server	CVE-2017-7269	root
tid_3	Web server	CVE-2014-8517	root
tid_4	FTP server	CVE-2014-3556	root
tid_5	E-mail server	CVE-2014-4877	root
tid_6	Database server	CVE-2013-4730	user
tid_7	Database server	CVE-2016-6662	root

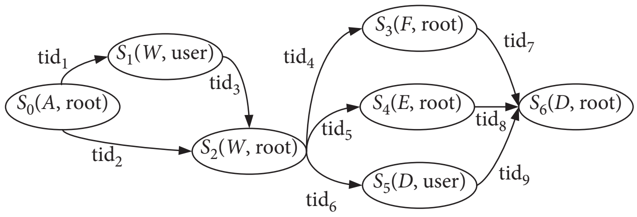


FIGURE 7: Attack graph.

Referring to MIT Lincoln Lab attack-defense behavior database, attack-defense templates are constructed. Attack-defense maps are divided into attack maps and defense maps by using attacker host A, web server W, database server D, FTP server F, and e-mail server E. In order to facilitate display and description, attack-defense maps are divided into attack maps and defense maps, as shown in Figures 7 and 8, respectively. The meaning of defense action in the defense diagram is shown in Table 2.

5.2. Construction of the Experiment Scenario AD-SGM

(1) $N =$ (attacker, defender) are players participating in the game representing cyber attackers and defenders, respectively

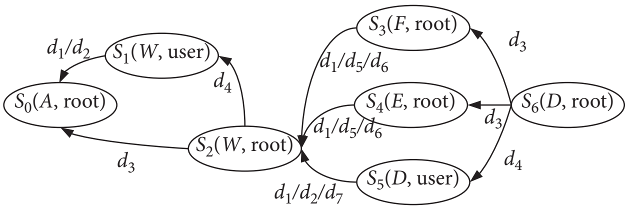


Figure 8: Defense graph.

Table 2: Defense action description.

Atomic defense action	d_1	d_2	d_3	d_4	d_5	d_6	d_7
Renew root data	✓		✓		✓	✓	
Limit SYN/ICMP packets		✓					
Install Oracle patches	✓						✓
Reinstall listener program	✓				✓		
Uninstall delete Trojan		✓				✓	
Limit access to MDSYS		✓		✓			
Restart database server			✓	✓	✓		
Delete suspicious account		✓					✓
Add physical resource	✓			✓	✓	✓	
Repair database			✓	✓			✓
Limit packets from ports	✓	✓	✓			✓	

(2) The state set of stochastic game is $S = (s_0, s_1, s_2, s_3, s_4, s_5, s_6)$, which consists of network state and is extracted from the nodes shown in Figures 7 and 8

(3) The action set of the defender is $D = (D_0, D_1, D_2, D_3, D_4, D_5, D_6)$, $D_0 = \{\text{NULL}\}D_1 = \{d_1, d_2\}D_2 = \{d_3, d_4\}D_3 = \{d_1, d_5, d_6\}D_4 = \{d_1, d_5, d_6\}D_5 = \{d_1, d_2, d_7\}D_6 = \{d_3, d_4\}$, and the edges are extracted from Figure 8

(4) Quantitative results of immediate returns $R_d(s_i, d, s_j)$ of defenders [16, 26] are

$$
\begin{aligned}
&\left(R_d(s_0, \text{NULL}, s_0), R_d(s_0, \text{NULL}, s_1), R_d(s_0, \text{NULL}, s_2)\right) = (0, -40, -59),\\
&\quad\left(R_d(s_1, d_1, s_0), R_d(s_1, d_1, s_1), R_d(s_1, d_1, s_2); R_d(s_1, d_2, s_0), R_d(s_1, d_2, s_1), R_d(s_1, d_2, s_2)\right)\\
&= (40, 0, -29; 5, -15, -32),\\
&\left(R_d(s_2, d_3, s_0), R_d(s_2, d_3, s_1), R_d(s_2, d_3, s_2), R_d(s_2, d_3, s_3), R_d(s_2, d_3, s_4), R_d(s_2, d_3, s_5);\right.\\
&\quad\left.R_d(s_2, d_4, s_0), R_d(s_2 d_4, s_1), R_d(s_2, d_4, s_2), R_d(s_2, d_4, s_3), R_d(s_2, d_4, s_4), R_d(s_2, d_4, s_5)\right)\\
&= (24, 9, -15, -55, -49, -65; 19, 5, -21, -61, -72, -68),\\
&\left(R_d(s_3, d_1, s_2), R_d(s_3, d_1, s_3), R_d(s_3, d_1, s_6); R_d(s_3, d_5, s_2), R_d(s_3, d_5, s_3), R_d(s_3, d_5, s_6);\right.\\
&\quad\left.R_d(s_3, d_6, s_2), R_d(s_3, d_6, s_3), R_d(s_3, d_6, s_6)\right)\\
&= (21, -16, -72; 15, -23, -81; -21, -36, -81),\\
&\left(R_d(s_4, d_1, s_2), R_d(s_4, d_1, s_4), R_d(s_4, d_1, s_6); R_d(s_4, d_5, s_2), R_d(s_4, d_5, s_4), R_d(s_4, d_5, s_6);\right.\\
&\quad\left.R_d(s_4, d_6, s_2), R_d(s_4, d_6, s_4), R_d(s_4, d_6, s_6)\right)\\
&= (26, 0, -62; 11, -23, -75; 9, -25, -87),\\
&\left(R_d(s_5, d_1, s_2), R_d(s_5, d_1, s_5), R_d(s_5, d_1, s_6); R_d(s_5, d_2, s_2), R_d(s_5, d_2, s_5), R_d(s_5, d_2, s_6);\right.\\
&\quad\left.R_d(s_5, d_7, s_2), R_d(s_5, d_7, s_5), R_d(s_5, d_7, s_6)\right)\\
&= (29, 0, -63; 11, -21, -76; 2, -27, -88),\\
&\left(R_d(s_6, d_3, s_3), R_d(s_6, d_3, s_3), R_d(s_6, d_3, s_5), R_d(s_6, d_3, s_6); R_d(s_6, d_4, s_3),\right.\\
&\quad\left.R_d(s_6, d_4, s_4), R_d(s_6, d_4, s_5), R_d(s_6, d_4, s_6)\right)\\
&= (-23, -21, -19, -42; -28, -31, -24, -49).
\end{aligned}
\tag{13}
$$

(5) In order to detect the learning performance of the Algorithm 2 more fully, the defender's state-action payoff $Q_d(s_i, d)$ is initialized with a unified 0, without introducing additional prior knowledge

(6) Defender's defense strategy adopts average strategy to initialize, that is, $\pi_d(s_k, d_1) = \pi_d(s_k, d_2) = \cdots = \pi_d(s_k, d_m)$, $\sum_{d\in D(s_k)}\pi_d(s_k, d) = 1, \forall s_k \in S$, where no additional prior knowledge is introduced

5.3. Testing and Analysis. The experiment in this section has three purposes. The first is to test the influence of different parameter settings on the proposed Algorithm 2 so as to find out the experimental parameters suitable for this scenario. The second is to compare this approach with the existing typical approaches to verify the advancement of this approach. The third is to test the effectiveness of WoLF-PHC algorithm improvement based on eligibility trace.

From Figures 7 and 8, we can see that the state of attack-defense strategy selection is the most complex and representative. Therefore, the performance of the algorithm is analyzed by the experimental state selection, and the other network state analysis approaches are the same.

5.3.1. Parameter Test and Analysis. Different parameters will affect the speed and effect of learning. At present, there is no relevant theory to determine the specific parameters. In Section 4, the relevant parameters are preliminarily analyzed. On this basis, the different parameter settings are further tested to find the parameter settings suitable for this attack-defense scenario. Six different parameter settings were tested. Specific parameter settings are shown in Table 3. In the experiment, the attacker's initial strategy is random strategy, and the learning mechanism is the same as the approach in this paper.

The probability of the defender's choice of defense actions and sums in state is shown in Figure 9. The learning speed and convergence of the algorithm under different parameter settings can be observed from Figure 9, which shows that the learning speed of settings 1, 3, and 6 is faster, and the best strategy can be obtained after learning less than 1500 times under the three settings, but convergence of 3 and 6 is poor. Although the best strategy can be learned by settings 3 and 6, there will be oscillation afterwards, and the stability of setting 1 is not suitable.

Defense payoff can represent the degree of optimization of the strategy. In order to ensure that the payoff value does not reflect only one defense result, the average of 1000 defense gains is taken, and the change of the average payoff per 1000 defense gains is shown in Figure 10. As can be seen from Figure 10, the benefits of Set 3 are significantly lower than those of other settings, but the advantages and disadvantages of other settings are difficult to distinguish. In order to display more intuitively, the average value of 25,000 defense gains calculated under different settings in Figure 10 is shown in Figure 11. From Figure 11, we can see that the average value of settings 1 and 5 is higher. For further comparison, the standard deviation of settings 1 and 5 is calculated one step on the basis of the average value to reflect the discreteness of the gains. As shown in Figure 12, the standard deviations of setting 1 and setting 6 are small. Moreover, the result of setting 1 is smaller than setting 6.

In conclusion, setting 1 of the six sets of parameters is the most suitable for this scenario. Since setting 1 has achieved an ideal effect and can meet the experimental requirements, it is no longer necessary to further optimize the parameters.

5.3.2. Comparisons. In this section, stochastic game [16] and evolutionary game [20] are selected to conduct comparative experiments with this approach. According to the difference of attacker's learning ability, this section designs two groups of comparative experiments. In the first group, the attacker's learning ability is weak and will not make adjustments to the attack-defense results. In the second group, the attacker's learning ability is strong and adopts the same learning mechanism as the approach in this paper. In both groups, the initial strategies of attackers were random strategies.

In the first group of experiments, the defense strategy of this approach is as shown in Figure 9(a). The defense strategies calculated by the approach [16] are $\pi_d(s_2, d_3) = 0.7$, $\pi_d(s_2, d_4) = 0.3$. The defense strategies of [20] are evolutionarily stable and balanced. And the defense strategies of [20] are $\pi_d(s_2, d_3) = 0.8$, $\pi_d(s_2, d_4) = 0.2$. Its average earnings per 1000 times change as shown in Figure 13.

From the results of the strategies and benefits of the three approaches, we can see that the approach in this paper can learn from the attacker's strategies and adjust to the optimal strategy, so the approach in this paper can obtain the highest benefits. Wei et al. [16] adopted a fixed strategy when confronting any attacker. When the attacker is constrained by bounded rationality and does not adopt Nash equilibrium strategy, the benefit of this approach is low. Although the learning factors of both attackers and defenders are taken into account in [20], the parameters required in the model are difficult to quantify accurately, which results in the deviation between the final results and the actual results, so the benefit of the approach is still lower than that of the approach in this paper.

In the second group of experiments, the results of [16, 20] are still as $\pi_d(s_2, d_3) = 0.7$, $\pi_d(s_2, d_4) = 0.3$ and $\pi_d(s_2, d_3) = 0.8$, $\pi_d(s_2, d_4) = 0.2$. The decision making of this approach is shown in Figure 14. After about 1800 times of learning, the approach achieves stability and converges to the same defense strategy as that of [16]. As can be seen from Figure 15, the payoff of [20] is lower than that of other two approaches. The average payoff of this approach in the first 2000 defenses is higher than that of [26], and then it is almost the same as that of [26]. Combining Figures 14 and 15, we can see that the learning attacker cannot get Nash equilibrium strategy at the initial stage, and the approach in this paper is better than that in [26]. When the attacker learns to get Nash equilibrium strategy, the approach in this paper can converge to Nash equilibrium strategy. At this time, the performance of this approach is the same as that in [26].

In conclusion, when facing the attackers with weak learning ability, the approach in this paper is superior to that

Table 3: Different parameter settings.

Set	α	δ_l	δ_w	λ	γ
1	0.01	0.004	0.001	0.01	0.01
2	0.1	0.004	0.001	0.01	0.01
3	0.01	0.004	0.001	0.01	0.1
4	0.01	0.004	0.001	0.1	0.01
5	0.01	0.04	0.01	0.01	0.01
6	0.01	0.008	0.001	0.01	0.01

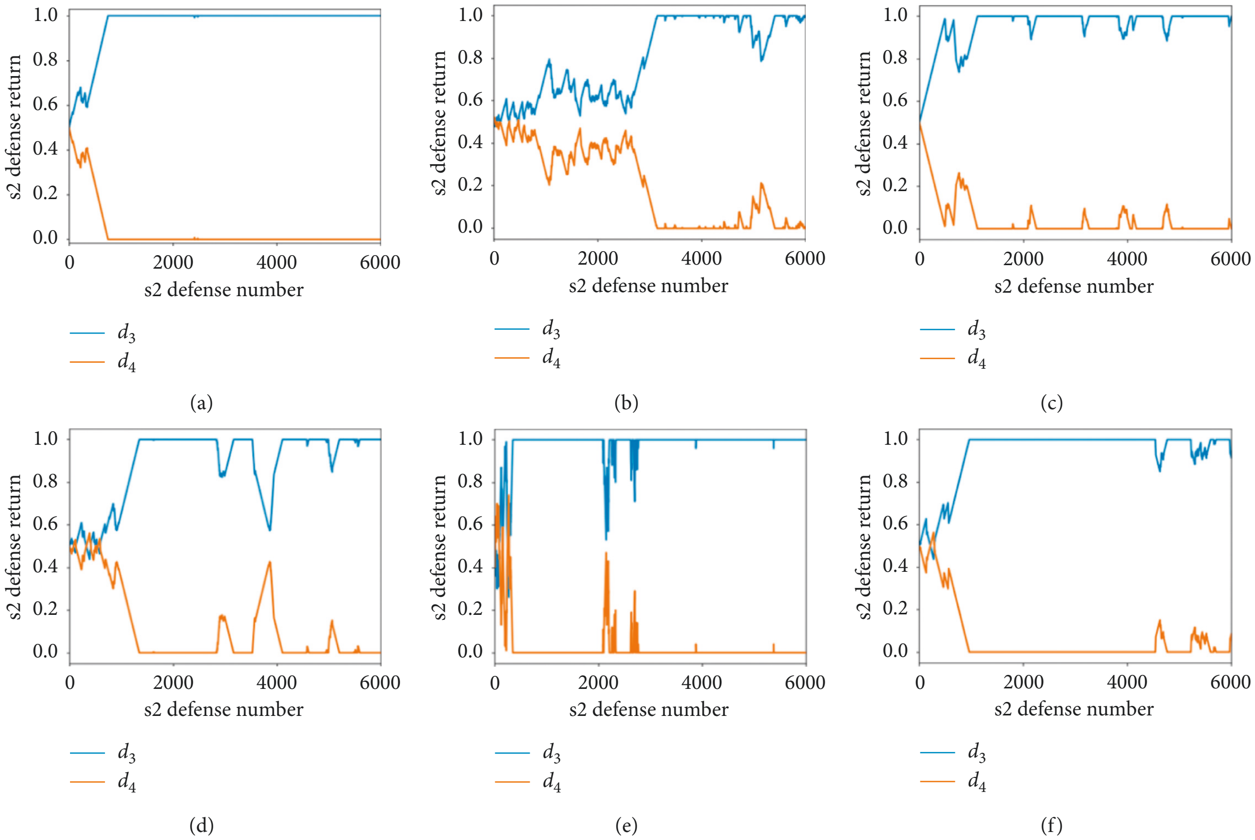


Figure 9: Defense decision making under different parameter settings. (a) Defense decision under setting 1. (b) Defense decision under setting 2. (c) Defense decision under setting 3. (d) Defense decision under setting 4. (e) Defense decision under setting 5. (f) Defense decision under setting 6.

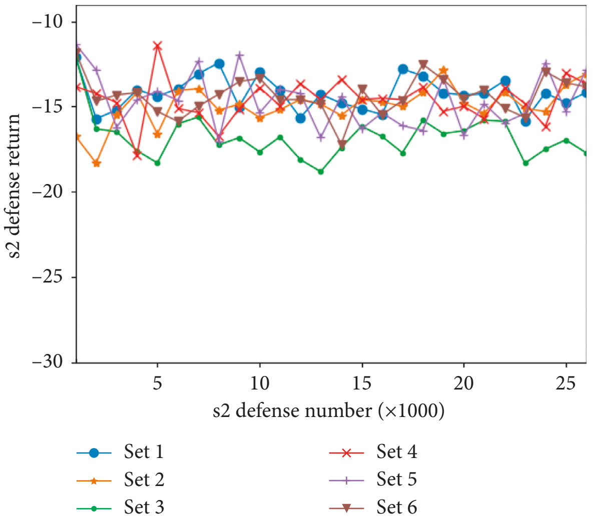


Figure 10: Defense payoff under different parameter settings.

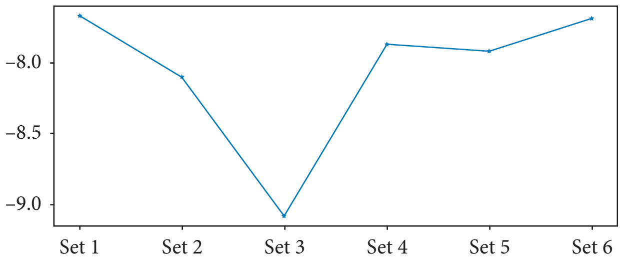


Figure 11: Average return under different parameter settings.

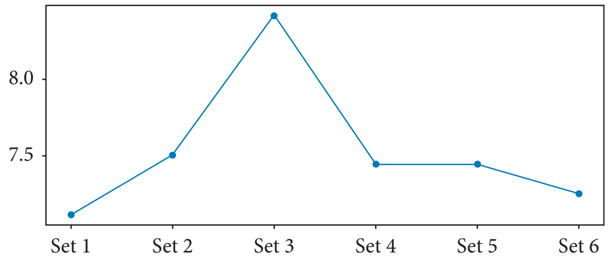


Figure 12: Standard deviation of defense payoff under different parameter settings.

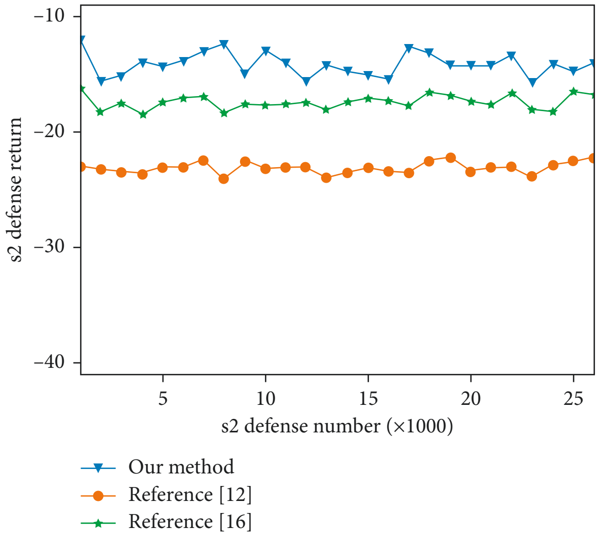


Figure 13: Comparison of defense payoff change.

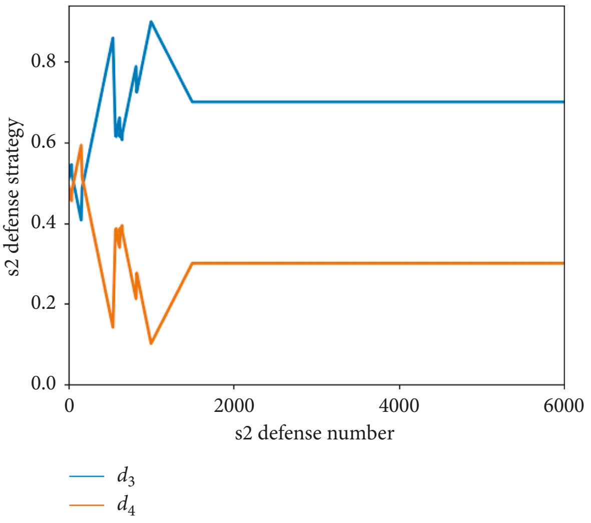


Figure 14: Defense decision making of our approach.

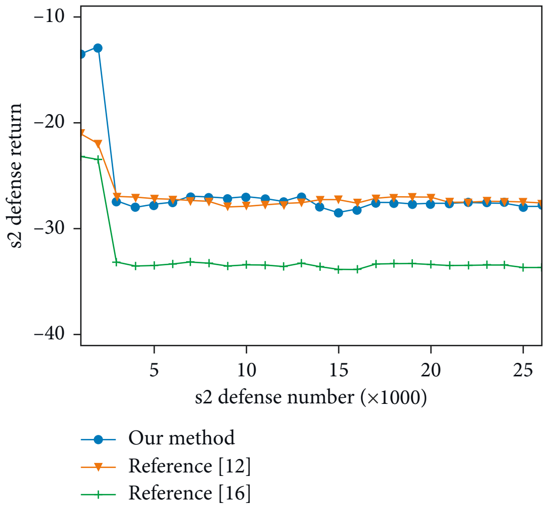


Figure 15: Comparison of defense payoff change.

in [16, 20]. When facing an attacker with strong learning ability, if the attacker has not obtained Nash equilibrium through learning, this approach is still better than [16, 20]. If the attacker obtains Nash equilibrium through learning, this paper can also obtain the same Nash equilibrium strategy as [26] and obtain its phase. The same effect is superior to that in [20].

5.3.3. Test Comparison with and without Eligibility Trace. This section tests the actual effect of the eligibility traces on the Algorithm 2. The effect of eligibility traces on strategy selection is shown in Figure 16, from which we can see that the learning speed of the algorithm is faster when the qualified trace is available. After 1000 times of learning, the algorithm can converge to the optimal strategy. When the qualified trace is not available, the algorithm needs about 2500 times of learning to converge.

Average earnings per 1,000 times change as shown in Figure 17, from which we can see that the benefits of the algorithm are almost the same when there is or not any qualified trace after convergence. From Figure 17, we can see that 3000 defenses before convergence have higher returns from qualified trails than those from unqualified trails. In order to further verify this, the average of the first 3000 defense gains under qualified trails and unqualified trails is counted 10 times each, respectively. The results are shown in Figure 18, which further proves that in the preconvergence defense phase, qualified traces are better than unqualified traces.

The addition of eligibility trace accelerates the learning speed but also brings additional memory and computing overhead. In the experiment, only 10 state-action pairs that were recently accessed were saved, which effectively reduced

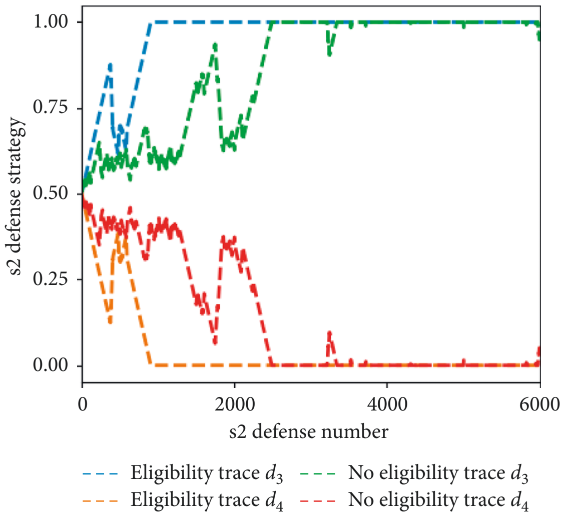


FIGURE 16: Comparison of defense decision making.

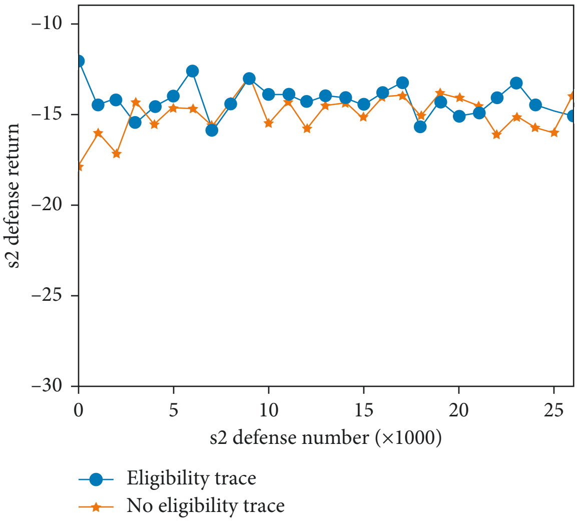


FIGURE 17: Comparison of defense payoff change.

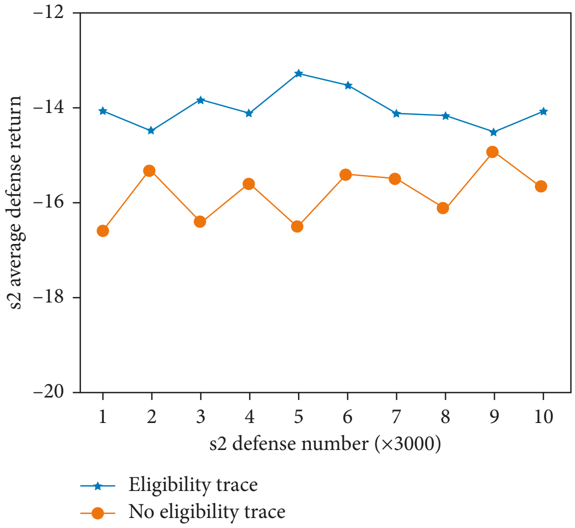


FIGURE 18: Average payoff comparison of the first 3000 defenses.

Table 4: Comprehensive comparison of existing approaches.

Ref.	Game type	Model assumption	Learning mechanism	Game process	Applicable object	Practicability
[3]	Stochastic game	Rationality	—	Multistage	Personal	Bad
[12]	Static game	Rationality	—	Single-stage	Personal	Bad
[14]	Dynamic game	Rationality	—	Multistage	Personal	Bad
[16]	Stochastic game	Rationality	—	Multistage	Personal	Bad
[20]	Evolutionary game	Bounded rationality	Biological evolution	—	Group	Good
Our approach	Stochastic game	Bounded rationality	Reinforcement learning	Multistage	Personal	Good

the increase of memory consumption. In order to test the computational cost of qualified trail, the time of 100,000 defense decisions made by the algorithm was counted for 20 times with and without qualified trail. The average of 20 times was 9.51s for qualified trail and 3.74 s for unqualified trail. Although the introduction of eligibility traces will increase the decision-making time by nearly 2.5 times, the time required for 100,000 decisions after the introduction of eligibility traces is still only 9.51 s, which can still meet the real-time requirements.

In summary, the introduction of eligibility trace at the expense of a small amount of memory and computing overhead can effectively increase the learning speed of the algorithm and improve the defense gains.

5.4. Comprehensive Comparisons. This approach is compared with some typical research results, as shown in Table 4. [3, 12, 14, 16] is based on the assumption of complete rationality. The equilibrium strategy obtained by it is difficult to appear in practice and has a low guiding effect on actual defense decision making. [20] and this paper have more practicability on the premise of bounded rationality hypothesis, but [20] is based on the theory of biological evolution and mainly studies population evolution. The core of game analysis is not the optimal strategy choice of players, but the strategy adjustment process, trend, and stability of group members composed of bounded rational players, and the stability here refers to group members. This approach is not suitable for guiding individual real-time decision making because the proportion of specific strategies is unchanged and not the strategy of a player. On the contrary, the defender of the proposed approach adopts reinforcement learning mechanism, which is based on systematic feedback to learn in the confrontation with the attacker and is more suitable for the study of individual strategies.

6. Conclusions and Future Works

In this paper, cyber attack-defense confrontation is abstracted as a stochastic game problem under the restriction of limited rationality. A host-centered attack-defense graph model is proposed to extract network state and attack-defense action, and an algorithm to generate attack-defense graph is designed to effectively compress the game state space. The WoLF-PHC-based defense decision approach is proposed to overcome the problem, which enables defenders under bounded rationality to make optimal choices when facing different attackers. The lattice improves the WoLF-PHC algorithm, speeds up the defender's learning speed, and reduces the algorithm's dependence on data. This approach not only satisfies the constraints of bounded rationality but also does not require the defender to know too much information about the attacker. It is a more practical defense decision-making approach.

The future work is to further optimize the winning and losing criteria of WoLF-PHC algorithm for specific attack-defense scenarios, in order to speed up defense learning and increase defense gains.

Data Availability

The data that support the findings of this study are not publicly available due to restrictions as the data contain sensitive information about a real-world enterprise network. Access to the dataset is restricted by the original owner. People who want to access the data should send a request to the corresponding author, who will apply for permission of sharing the data from the original owner.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National High Technology Research and Development Program of China (863 Program) (2014AA7116082 and 2015AA7116040) and National Natural Science Foundation of China (61902427). In particular, we thank Junnan Yang and Hongqi Zhang of our research group for their previous work on stochastic game. We sincerely thank them for their help in publishing this paper.

References

[1] Z. Tian, W. Shi, Y. Wang et al., "Real time lateral movement detection based on evidence reasoning network for edge computing environment," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4285–4294, 2019.

[2] Y. Wang, J. Yu, W. Qu, H. Shen, and X. Cheng, "Evolutionary game model and analysis approaches for network group behavior," *Chinese Journal of Computers*, vol. 38, no. 2, pp. 282–300, 2015.

[3] J. Liu, H. Zhang, and Y. Liu, "Research on optimal selection of moving target defense policy based on dynamic game with incomplete information," *Acta Electronica Sinica*, vol. 46, no. 1, pp. 82–29, 2018.

[4] Z. Tian, X. Gao, S. Su, J. Qiu, X. Du, and M. Guizani, "Evaluating reputation management schemes of internet of vehicles based on evolutionary game theory," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 5971–5980, 2019.

[5] Q. Zhu and S. Rass, "Game theory meets network security: a tutorial," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pp. 2163–2165, Toronto, Canada, October 2018.

[6] X. Li, C. Zhou, Y. C. Tian, and Y. Qin, "A dynamic decision-making approach for intrusion response in industrial control systems," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 5, pp. 2544–2554, 2019.

[7] H. Hu, Y. Liu, H. Zhang, and R. Pan, "Optimal network defense strategy selection based on incomplete information evolutionary game," *IEEE Access*, vol. 6, pp. 29806–29821, 2018.

[8] J. Li, G. Kendall, and R. John, "Computing Nash equilibria and evolutionarily stable states of evolutionary games," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 3, pp. 460–469, 2016.

[9] M. Bowling and M. Veloso, *Multiagent Learning Using a Variable Learning Rate*, Elsevier Science Publishers Ltd, Amsterdam, Netherlands, 2002.

[10] J. Harb and D. Precup, "Investigating recurrence and eligibility traces in deep Q-networks," https://arxiv.org/abs/1704.05495, 2017.

[11] C. T. Do, N. H. Tran, and C. Hong, "Game theory for cyber security and privacy," *ACM Computing Surveys*, vol. 50, no. 2, p. 30, 2017.

[12] Y.-L. Liu, D.-G. Feng, L.-H. Wu, and Y.-F. Lian, "Performance evaluation of worm attack and defense strategies based on static Bayesian game," *Chinese Journal of Software*, vol. 23, no. 3, pp. 712–723, 2012.

[13] 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.

[14] H. Zhang, L. Jiang, S. Huang, J. Wang, and Y. Zhang, "Attack-defense differential game model for network defense strategy selection," *IEEE Access*, vol. 7, pp. 50618–50629, 2019.

[15] A. Afrand and S. K. Das, "Preventing DoS attacks in wireless sensor networks: a repeated game theory approach," *Internet Journal of Cyber Security*, vol. 5, no. 2, pp. 145–153, 2007.

[16] J. Wei, B. Fang, and Z. Tian, "Research on defense strategies selection based on attack-defense stochastic game model," *Chinese Journal of Computer Research and Development*, vol. 47, no. 10, pp. 1714–1723, 2010.

[17] C. Wang, X. Cheng, and Y. Zhu, "A Markov game model of computer network operation," *Chinese Systems Engineering-Theory and Practice*, vol. 34, no. 9, pp. 2402–2410, 2014.

[18] J. Hofbauer and K. Sigmund, "Evolutionary game dynamics," *Bulletin of the American Mathematical Society*, vol. 40, no. 4, pp. 479–519, 2011.

[19] 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.

[20] J. Huang and H. Zhang, "Improving replicator dynamic evolutionary game model for selecting optimal defense strategies," *Chinese Journal on Communications*, vol. 38, no. 1, pp. 170–182, 2018.

[21] H. Hu, Y. Liu, H. Zhang, and Y. Zhang, "Security metric methods for network multistep attacks using AMC and big data correlation analysis," *Security and Communication Networks*, vol. 2018, Article ID 5787102, 14 pages, 2018.

[22] H. Hu, Y. Liu, Y. Yang, H. Zhang, and Y. Zhang, "New insights into approaches to evaluating intention and path for network multistep attacks," *Mathematical Problems in Engineering*, vol. 2018, Article ID 4278632, 13 pages, 2018.

[23] H. Hu, H. Zhang, Y. Liu, and Y. Wang, "Quantitative method for network security situation based on attack prediction," *Security and Communication Networks*, vol. 2017, Article ID 3407642, 19 pages, 2017.

[24] M. Zimmer and S. Doncieux, "Bootstrapping Q-learning for robotics from neuro-evolution results," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 10, no. 1, pp. 102–119, 2018.

[25] R. S. Sutton and A. G. Barto, "Reinforcement learning: an introduction, Bradford book," *Machine Learning*, vol. 16, no. 1, pp. 285-286, 2005.

[26] H. Zhang, J. Wang, D. Yu, J. Han, and T. Li, "Active defense strategy selection based on static Bayesian game," in *Proceedings of IET International Conference on Cyberspace Technology*, pp. 1–7, London, UK, October 2016.

Hindawi
Security and Communication Networks
Volume 2019, Article ID 1362964, 14 pages
https://doi.org/10.1155/2019/1362964

Research Article

Active Defense Strategy Selection Method Based on Two-Way Signaling Game

Xiaohu Liu,[1,2] Hengwei Zhang,[1,2] Yuchen Zhang,[1,2] Lulu Shao,[1] and Jihong Han[1,2]

[1] *Zhengzhou Information Science and Technology Institute, Zhengzhou 450001, China*
[2] *State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou 450001, China*

Correspondence should be addressed to Hengwei Zhang; wlby_zzmy_henan@163.com

Received 23 June 2019; Revised 25 September 2019; Accepted 31 October 2019; Published 29 November 2019

Guest Editor: Akbar S. Namin

Most network security research studies based on signaling games assume that either the attacker or the defender is the sender of the signal and the other party is the receiver of the signal. The attack and defense process is commonly modeled and analyzed from the perspective of one-way signal transmission. Aiming at the reality of two-way signal transmission in network attack and defense confrontation, we propose a method of active defense strategy selection based on a two-way signaling game. In this paper, a two-way signaling game model is constructed to analyze the network attack and defense processes. Based on the solution of a perfect Bayesian equilibrium, a defense strategy selection algorithm is presented. The feasibility and effectiveness of the method are verified using examples from real-world applications. In addition, the mechanism of the deception signal is analyzed, and conclusions for guiding the selection of active defense strategies are provided.

1. Introduction

Network information technology is developing rapidly, and interconnected systems are on the rise [1]. However, network security incidents pose a major and perpetual problem [2]. Defense technologies represented by firewalls, intrusion detection, and antivirus software provide passive response defense based on a priori knowledge and attack characteristics, but they cannot respond to new types of complex network attacks in an effective and timely manner [3]. If the defending party can actively select a targeted defense strategy by predicting the attacker's actions and disrupt or block the attack process, while simultaneously maximizing its own benefits, then the defense may be called an active defense [4]. The essence of cybersecurity is a battle between the offense and defense. The effectiveness of the defense depends not only on its own strategic action, but also influenced and constrained by the attacker's action [5]. The key issue is how to select the optimal active defense strategy in an information-constrained confrontation environment.

The characteristics of opposite goals, strategic dependence, and noncooperative relationships in network attack and defense are in line with the core philosophy of game theory, namely, optimal decision in an environment of conflict. Some scholars, such as the authors of Refs. [6–11], have established network security models based on game theory, analyzed the offensive and defensive confrontation process, and solved the game equilibrium to determine the defense strategy and guide defense actions. We classified and analyzed the existing research results by combining the two factors of game information and action timing and came to the following conclusions:

(1) In a static game with complete information, there are many premise assumptions and the model is easy to establish, as demonstrated in Ref. [12].

(2) In a dynamic game with complete information, given the sustained nature of the offensive and defensive confrontation process, previous actions could be studied to affect the subsequent game process, as shown in Ref. [13].

(3) In a static game with incomplete information, the players may use the static Bayes' rule to infer the opponent's private information and break through

the complete information assumption, such as in Ref. [14].

(4) In a dynamic game with incomplete information, the late player observes the partial action of the early player, even without fully understanding the behavior type. However, since the behavior is type dependent, one can modify the a priori judgment of the behavior type of the early player by using the dynamic Bayes' rule, as depicted in Ref. [15]. Since neither the offense player nor the defense player can fully understand the opponent's information, influenced by the dynamic and persistent nature of the confrontation process, the dynamic game with incomplete information is more in line with the actual network attack and defense. Hence, this type of game is the focus of current network security game research.

A signaling game is a typical dynamic game with incomplete information, which provides a formal mathematical way to analyze how identity and deception are coupled in cyber-social systems. [16] It describes the strategic interplay of the game process through signal transmission [17], which is well-suited for studying the selection of active defense strategy. In Ref. [18], from the perspective of dynamic confrontation and limited information, a two-stage signaling game model is constructed to derive an optimal defense strategy. As demonstrated in Ref. [19], the signaling game model can be used to analyze the moving target defense. The defense side can alter the information asymmetry of the two sides by releasing the dynamically transformed signal and thereby expand its own benefits. In Ref. [20], the DDoS attack and defense process is modeled as a multistage signaling game, and an equilibrium solution is found. Moreover, the server port hopping defense strategy has been demonstrated to be effective. In Ref. [21], a multistage offensive and defensive signaling game model is constructed for modeling the multistage dynamic attack and defense process under incomplete information constraints. Also, the signal attenuation factor is used to quantify the influence of the defensive signal of the defending party. In Ref. [22], to address the spear-phishing attack of industrial control systems, a multistage offensive and defensive game model is established. Defense strategies are selected based on the comprehensive consideration of the benefits and costs. Finally, Ref. [23] analyzes the security issues of the Internet of Things through a multistage game model and provides specific defense strategies.

Despite their strengths, all the studies above assume that the network attack and defense process involve only one-way signal transmission, so the attack and defense process is modeled and analyzed by designating either the attacker or defender as the signal sender and the other party as the signal receiver. However, in an actual network attack and defense process, the attacker and the defender will have a series of strategic interactions. The attack and defense parties are generally both senders and receivers of signals. If the sender's transmitted signal is viewed as a stimulus, then the response chosen by the recipient is a reaction. In a two-way sustained stimulus-response process, the defender and the attacker are constantly adjusting and optimizing their respective strategies, thus dynamically propelling the attack and defense evolution [24]. Therefore, the game signal in network attack and defense should be a two-way send-and-receive mechanism.

To address the problem described above, we construct a two-way signaling game model to analyze the network attack and defense processes based on a two-way transmission mechanism of actual attack and defense signals. Based on the solution of the perfect Bayesian equilibrium, a defense strategy selection algorithm is presented. The main contributions of this work are as follows:

(1) Two-way signal transmission mechanism: both the offense and defense parties play a dual role of the sender and receiver. While affecting the other party's strategy selection by releasing the signal, they are also affected by the signal released by the other party.

(2) Game signal set containing both true and fake signals: in order to disrupt the cognitive decision-making process of the other party, both the offense and defense sides in the process of network confrontation use information countermeasures that release a mixture of true and false signals. Since the signal recipient has a certain discriminating ability against false signals, the deceptive effect of the false signal diminishes as the attack and defense game progresses.

(3) Dynamic multistage game process: the offensive and defensive confrontation continues in multiple stages as both sides continue to learn and evolve based on the interaction of signals, dynamically adjust the action strategy, and maximize their gains. Through a two-way signal transmission mechanism, the method proposed in this paper can more accurately characterize the offensive and defensive strategy confrontation process. Hence, this method more closely models an actual network attack and defense process. It also serves as a better theoretical reference, providing practical guidance in the selection of active defense strategies under dynamic conditions of incomplete information.

2. Construction of a Two-Way Attack and Defense Game Signal Model

2.1. Analysis of Attack and Defense Game Process

2.1.1. Basic Signaling Game Process. The basic signaling game consists of two players: the signal sender and the signal receiver. First, according to the Harsanyi conversion [25], the virtual player "Nature" selects the type of signal sender as θ and transforms the selection problem under the condition of incomplete information into a selection problem under the condition of uncertainty type. The signal sender knows that its type is θ, but the signal receiver only knows the a priori probability $P(\theta)$ that the sender belongs to type θ. The signal sender releases a signal H, and the signal receiver, having observed signal H, uses Bayes' rule to deduce the posteriori probability $P(\theta | H)$ from the a priori probability $P(\theta)$ and subsequently selects an action strategy. The signal sender determines its own action strategy by predicting the signal receiver's action strategy, and both parties strive to maximize their respective gains. The process of the basic signaling game is shown in Figure 1.

2.1.2. Two-Way Attack and Defense Signaling Game Process. Network confrontations are dynamic and sustained. The attacker and the defender take sequential actions, and each party selects its own action strategy after observing the signal released by the other party. The two-way signaling game process is shown in Figure 2.

(1) Initial Configuration (ICN). The defender acts as the signal sender, and the attacker acts as the signal receiver. The defender deploys the network information system and configures the network topography, IP address, and network segmentation. Since the network must provide services to the outside world, it is characterized by open sharing, interconnection, and interoperability. The network must also have homologous, isomorphic, and homogenous characteristics of information network products. The attacker can gather information on the initial configuration of the defender through a variety of avenues, including infiltration by social engineering means, continuous scanning and detection, and public information acquisition [26]. Such information serves as the basis for the attacker to launch a network attack. In this work, the information is treated as a signal H_D released by the defender. The attacker observes the signal H_D, corrects the a priori judgment regarding the type of defender, and identifies its attack strategy. The game process is shown in the S_1 stage of Figure 2.

(2) Dynamic Confrontation (DCN). Both the offense and defense sides are constantly switching between the role of the signal sender and the signal receiver. Each stage of the game consists of a basic signaling game, as shown in the S2, S3, and S*i* stages in Figure 2. In the S_2 phase, the attacker selects the attack strategy and releases the signal H_A. The defender receives the signal H_A, corrects the a priori judgment about the type of the attacker, and selects the defense strategy accordingly. In the S_3 stage, the defender releases the signal H_D and the attacker receives the signal H_D and again corrects the a priori assessment regarding the type of the defender to determine the attack strategy. In the process of dynamic confrontation, the signal is transmitted in both directions, and both the offense and defense sides use Bayes' rule to incrementally correct their estimate of the true type of the other party. From the perspective of the defender, the termination condition of the game is when the attacker stops the attack and no longer releases signals. The game process is shown in the S_n phase of Figure 2.

2.2. Definition of Two-Way Attack-Defense Signaling Game Model. The signal plays a role in the strategic interaction between the sender and receiver. The sender of the signal determines the content of the signal and influences the recipient's action strategy through the signal. According to the Cyber Kill Chain model [27], the first stage of network reconnaissance is an intelligence gathering activity, such as detection and scanning, which is conducted by the attacker on the defender. This may be regarded as receiving the signal released by the defender. In the course of the confrontation, the sender of the signal may adopt the idea of deception by releasing signals that do not match its own type for the purpose of misleading the other party's judgment and expanding its own gain [28]. Therefore, the signals transmitted by both the offense and defense parties can be divided into two types: real signals and deception signals.

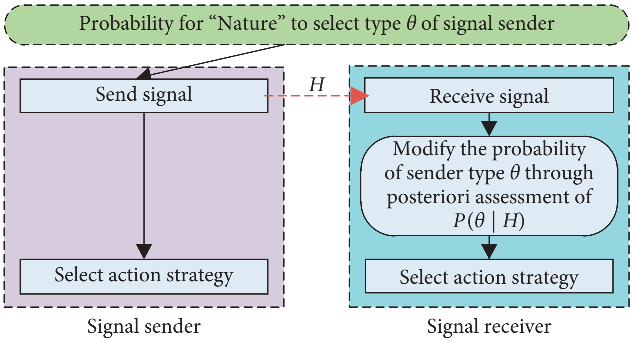

Probability for "Nature" to select type θ of signal sender
Send signal
H
Receive signal
Modify the probability of sender type θ through posteriori assessment of $P(\theta | H)$
Select action strategy
Select action strategy
Signal sender
Signal receiver

FIGURE 1: The basic signaling game process.

Definition 1 (real signal (RS)). A real signal is a signal that reflects the true type of the player. The player chooses the action strategy according to its own type. In the process of implementing its strategy, some private information is inevitably exposed; this information is transmitted to the receiver as a real signal. A real signal is accompanied by an action strategy, and the release of a real signal does not require additional cost.

Definition 2 (deception signal (DS)). A deception signal is a signal that does not match the true type of the player. In order to conceal its real type, the player induces the signal receiver to establish a wrong correction to the a priori probability by sending a signal that does not match its type, thereby rendering the receiver into a passive state. Since a signal will not be generated for no reason, the deceptive player must pay an extra cost to release the deceptive signal [29]. For example, if a low-defense user wishes to spoof as a high-defense user, it must deploy some camouflage facility and pay a certain defense cost to release the spoofing signal. The release of defensive signals by the defense player is a concrete manifestation of the active defense philosophy [30], in line with the deceptive concept that "when we are able to attack, we must seem unable; when using our forces, we must seem inactive" in Sun Tzu's *The Art of War*.

Based on the above analysis, a two-way signaling game (TWSG) model is constructed for the two-way transmission mechanism in the actual network attack and defense confrontation process.

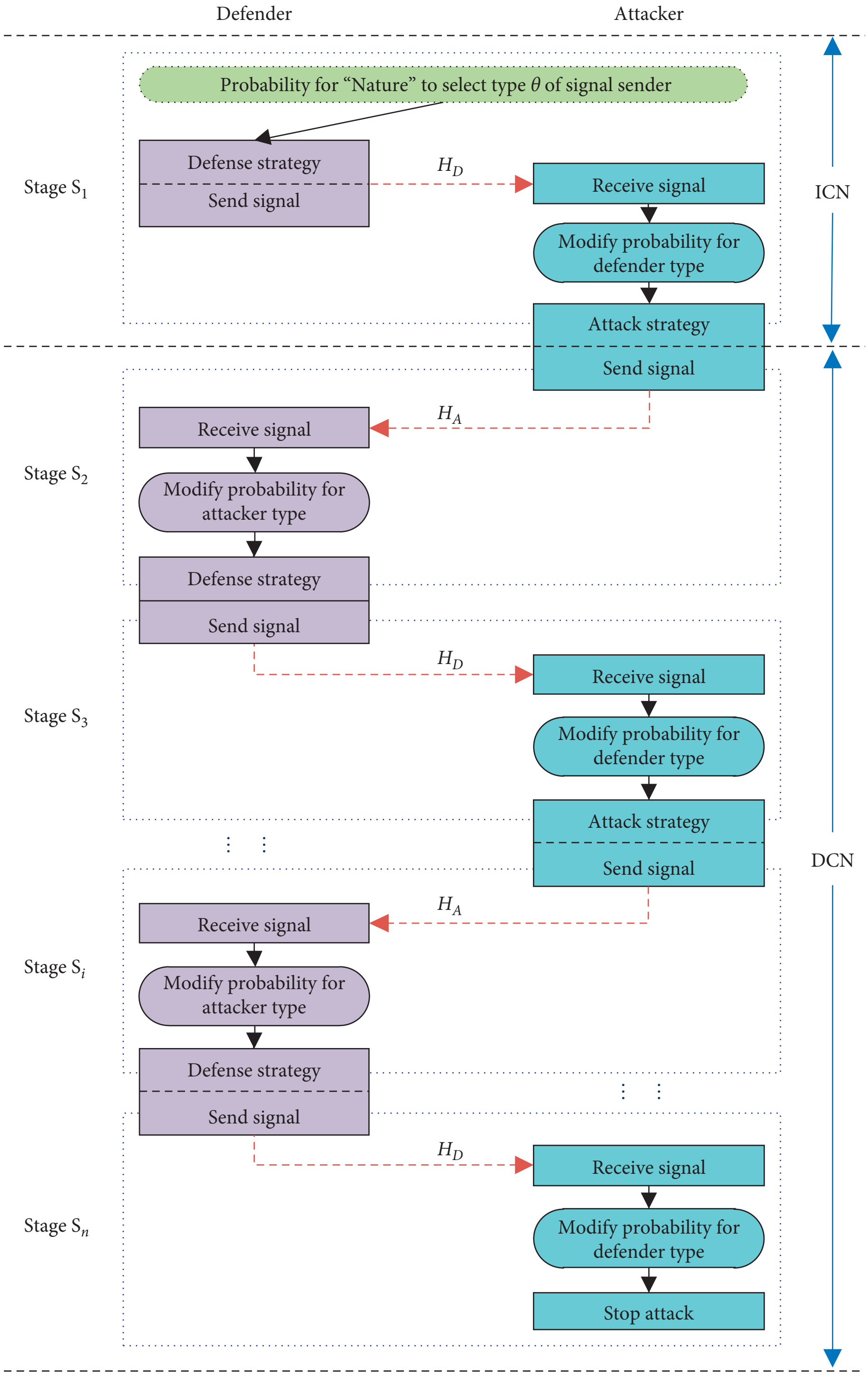


Figure 2: Two-way signaling game process.

Definition 3. The *TWSG* model has ten elements, where TWSG = $(N, \Theta, H, T, \sigma, \xi, S, P, \widetilde{P}, U)$.

① $N = (N_D, N_A)$ is the player space of the game. It includes two players: the defender N_D and the attacker N_A.

② $\Theta = (\theta_D, \theta_A)$ is the type space. θ_D is the type of the defender, $\theta_D = (\phi_i \mid i = 1, 2, \ldots n)$, $n \geq 2$, and θ_A is the type of the attacker, $\theta_A = (\varphi_j \mid j = 1, 2, \ldots m)$, $m \geq 2$. The type of the player is private information, determined by the action strategy, and the player type can affect the game return of both parties.

③ $H = (H_D, H_A)$ is the signal space. H_D is the defense signal, $H_D = (h_{Dk} \mid k = 1, 2, \ldots v)$, $v \geq 2$, and H_A is the attack signal, $H_A = (h_{Al} \mid l = 1, 2, \ldots w)$, $w \geq 2$. The signal receiver can estimate the type of sender according to the signal received, and the signal space logically corresponds to the type space. However, due to the existence of the spoofing signal, a specific signal does not have a strict correspondence relationship with the specific type of the attacker or defender.

④ T is the number of game stages, and $T = (1, 2, 3, \ldots, t)$, $t \geq 3$. The two-way signaling game continues in multiple stages, and the tth stage of the game is represented as TWSG(t).

⑤ σ is the spoofing signal attenuation factor. After multiple strategic interactions between the attacker and defender, the two sides become more familiar with each other, and the influence of deception signals is gradually attenuated. The posteriori probability generated in the tth stage of the game is modified by the factor σ_t to make it more realistic, where $0 \leq \sigma_t \leq 1$. The initial stage deception signal is not attenuated. The degree of attenuation of the deception signal at the TWSG(t) stage is expressed as $\sigma_t = \sigma^{t-1}$. For a sufficiently large T, $\sigma_T = \sigma^{T-1} \approx 0$, and the influence of the spoofing signal disappears completely. The signal and type constitute a corresponding relationship, and the two-way signaling game degenerates into a static game of incomplete information.

⑥ ξ is the gain discount factor and ξ represents the discount ratio of the gain in the $t+1$ stage as well as the gain in the t-stage. The discount ratio is used to convert the gain of a future stage into the present value.

⑦ $S = (S_D, S_A)$ is the strategy space. S_D is a defensive party strategy, $S_D = \{d_g \mid g = 1, 2, \ldots\}$ and S_A is an attacker party strategy, $S_A = \{a_h \mid h = 1, 2, \ldots\}$.

⑧ $P = (P_D, P_A)$ is the a priori probability space. P_D is the set of a priori probability of the defender, and it represents the a priori probability of the attacker's type known to the defender, where $P_D \neq \varnothing$, $P_D = [p_{D1}, p_{D2}, \ldots, p_{DT}]$. P_A is the a priori probability of the attacker, and it represents the a priori probability of the defender's type known to the attacker, where $P_A \neq \varnothing, P_A = [p_{A1}, p_{A2}, \ldots, p_{AT}]$.

⑨ $\widetilde{P} = (\widetilde{P}_D, \widetilde{P}_A)$ is the posteriori probability space. $\widetilde{P}_D$ is a set of posteriori probability of the defender, meaning the defender's posteriori assessment of the attacker's type, where $\widetilde{P}_D(\varphi_j \mid h_l) = (\varepsilon_{D1}, \varepsilon_{D2}, \ldots, \varepsilon_{\mathrm{DT}})$. $\widetilde{P}_A$ is the attacker's posteriori probability set, meaning the attacker's posteriori assessment of the defender's type, where $\widetilde{P}_A(\phi_i \mid h_k) = (\varepsilon_{A1}, \varepsilon_{A2}, \ldots, \varepsilon_{\mathrm{AT}})$.

⑩ $U = (U_D, U_A)$ is the gain space. U_D and U_A represent the defender's gain and the attacker's gain, respectively.

2.3. Gain Calculation. Based on the characteristics of the two-way signaling game model, we provide the following definition and calculation method for the game return.

Definition 4. The system damage cost (SDC), attack cost (AC), defense cost (DC), and related definitions and calculation methods can be found in Refs. [23, 31, 32]. Among them, SDC is affected by the combination of attack and defense strategies and is often recorded as $\mathrm{SDC}(d_g, a_h)$, which represents the value that the system suffers when the defense strategy is d_g and the attack strategy is a_h.

Definition 5 (deception cost). The deception defense cost (DDC) is the cost incurred to the defense party for actively releasing a spoofing signal to confuse the attacker. The deception attack cost (DAC) is the cost incurred to the attacking party for actively releasing a spoofing signal to confuse the defender.

According to the cost/reward calculation method, the returns of the attacker are the SDC and the total cost is the sum of the AC and DAC. The defender's cost is the sum of the SDC, DC, and DDC.

The discount factor ξ is used to convert future earnings into current gain. The gain target functions of the offensive and defensive parties can be expressed, respectively, as follows:

$$
\begin{aligned}
U_A\left(d_g, a_h, t\right) &= \sum_{g,h,t} \xi^{t-1}\left[\mathrm{SDC}\left(d_g, a_h\right) - \mathrm{AC} - \mathrm{DAC}\right],\\
U_D\left(d_g, a_h, t\right) &= -\sum_{g,h,t} \xi^{t-1}\left[\mathrm{SDC}\left(d_g, a_h\right) + \mathrm{DC} + \mathrm{DDC}\right].
\end{aligned}
\tag{1}
$$

According to the attack-defense types of θ_A and θ_D, the attack-defense strategies can be divided into different levels, such as enhanced type and regular type. The costs and returns of the strategies at the same level are basically the same. For example, if an attack level contains a total of h attack policies, then the probability that the attacker selects the strategy a_h is *1/h*. The gain from this attack level can be expressed as an average of $\overline{U}_A(d_g, a_h, t) = \sum_h U_A(d_g, a_h, t)/h$. Similarly, if a defense level has a total of g defensive strategies, the gain of the defense level is $\overline{U}_D(d_g, a_h, t) = \sum_g U_D((d_g, a_h, t))/g$.

3. Two-Way Signaling Game Equilibrium Solution and Defense Strategy Selection

A two-way signaling game is a finite game consisting of several basic signaling games. In the game, the attacker and defender alternately act as signal senders and receivers and the single role equilibrium solution is no longer applicable. In this paper, we first present the solution process for a one-stage game equilibrium and then apply it to a multistage equilibrium solution.

We carry out the calculation and analysis for the single-stage game equilibrium solution by referring to the signal sender as the Leader and the signal receiver as the Follower. The relevant parameters are set as follows:

① Signal sender action strategy $\{l_1, l_2, \ldots, l_n\}$

② Signal receiver action strategy $\{f_1, f_2, \ldots, f_m\}$

③ Defender type space $\theta_D = (\phi_{\mathrm{DH}}, \phi_{\mathrm{DM}})$ = (enhanced type defense, regular type defense)

④ Defender's signal space $H_D = (h_{\mathrm{DH}}, h_{\mathrm{DM}})$ = (enhanced defense signal, regular defense signal)

⑤ Attacker type space $\theta_A = (\varphi_{\mathrm{AH}}, \varphi_{\mathrm{AM}})$ = (enhanced attack, regular attack)

⑥ Attacker signal space $H_A = (h_{\mathrm{AH}}, h_{\mathrm{AM}})$ = (enhanced attack signal, regular attack signal)

3.1. Single-Stage Game Equilibrium Solution

Definition 6. The TWSG(t) game equilibrium solution is $\mathrm{EQ}_t = (h^*(l^*, \Theta), f^*(h), \widetilde{P}_F(\Theta \mid h))$, where $h^*(l^*, \Theta)$ is the Leader's signal strategy, abbreviated as $h^*(\Theta)$, $f^*(h)$ is the Follower's strategy, abbreviated as $f^*(h)$, and $\widetilde{P}_F(\Theta \mid h)$ is the Follower's posteriori probability of the Leader type, where the parameter $F \in \{A, D\}$ indicates that the Follower can be an attacker or defender in different game stages, abbreviated as $\widetilde{P}_F(\Theta)$. According to game theory, the equilibrium should satisfy two conditions:

(i) $f^*(h) \in \arg\max_{f\in F} \sum \widetilde{P}_F(\Theta \mid h) U_F(h^*(\Theta), f, \Theta)$, indicating that under the condition of posteriori probability $\widetilde{P}_F(\Theta \mid h)$, the Follower is the optimal strategy for the Leader

(ii) $h^*(\Theta) \in \arg\max_{h\in H} U_L(h, f^*(h), \Theta)$, indicating that the Leader is the optimal strategy for the Follower

Here, $\widetilde{P}_F(\Theta \mid h)$ represents the posteriori probability of the Leader type calculated for the Follower based on a priori probability P, observed signal h, and its own strategy $f^*(h)$.

The steps for solving the perfect Bayesian equilibrium is more complex, and the entire process may be divided into the following three steps:

(1) *Step 1.* Calculate optimal strategy $f^*(h)$ based on the signal received by the Follower

(2) *Step 2.* Leader reduces the optimal strategy $h^*(\Theta)$

(3) *Step 3.* Select the perfect equilibrium solution $\mathrm{EQ}_t = (h^*(\Theta), f^*(h), \widetilde{P}_F(\Theta))$

The detailed process is shown in the Appendix.

Based on game theory, the perfect Bayesian equilibrium solution is the optimal strategy for the player [33]. Therefore, the defender should determine the active defense strategy based on its role and game equilibrium EQ_t.

3.2. Multistage Game Equilibrium Solution.

In the multistage continuous confrontation process, the defense party may incrementally modify the attacker's motivation and behavioral preference using the stimulus-response learning mechanism, reduce the impact of the attacker's deception signal, and implement a targeted active defense strategy to maximize the expected return.

(1) In the first stage of the game TWSG(1), the Leader is the defender and the Follower is the attacker.

Based on the Harsanyi conversion, the viral player "Nature" selects the type of the defender. Type ϕ_{DH} is selected with a priori probability p_1, and type ϕ_{DM} is selected with probability $1 - p_1$. The defender releases the signals h_{DH} and h_{DM}. Based on the observed signals, the attacker selects strategy types φ_{AH} and φ_{AM} and corrects its a priori assessment of the defender type. According to the single-stage game equilibrium solution process in Section 3.1, the game equilibrium $\mathrm{EQ}_1 = (h^*(\Theta), f^*(h), \widetilde{P}_A(\Theta))$ can be obtained for TWSG(1). The TWSG(1) game tree is shown in Figure 3.

(2) In the second stage of the game TWSG(2), the Leader is the attacker and the Follower is the defender.

The attacker selects the attack strategy according to EQ_1 and sends a signal to the defender. The offense and defense sides have interchanged their role as the sender and receiver of the signal. Through the TWSG(1) game, both the offensive and defensive sides have gained some mutual understanding and the decay phenomenon of the deception signal begins to emerge. At this point, the attacker no longer relies on "Nature" to select the type. Instead, the selection is determined by the signal attenuation factor σ of the deception signal and the posteriori probability $\mathrm{EQ}_1(\widetilde{P}_A(\Theta))$ in EQ_1, as expressed by $\sigma\mathrm{EQ}_1(\widetilde{P}_A(\Theta))$. The attacker chooses φ_{AH} with probability $\sigma\mathrm{EQ}_1(\widetilde{P}_A(\Theta))$ and chooses φ_{AM} with probability $1 - \sigma\mathrm{EQ}_1(\widetilde{P}_A(\Theta))$. The TWSG(2) game tree is shown in Figure 4.

(3) In the third stage game TWSG(3), the Leader is the defender and the Follower is the attacker. The TWSG(3) game tree is shown in Figure 5.

The defender selects the defense strategy according to EQ_2 and sends a signal to the attacker. The attack and defense roles are interchanged again. After the first two stages of the game, the attenuation effect of the deception signal is more pronounced, as represented by the expression $\sigma^2\mathrm{EQ}_2(\widetilde{P}_D(\Theta))$. The defender chooses ϕ_{DH} with probability $\sigma^2\mathrm{EQ}_2(\widetilde{P}_D(\Theta))$ and selects ϕ_{DM} with probability $1 - \sigma^2\mathrm{EQ}_2(\widetilde{P}_D(\Theta))$.

(4) In the T-stage of the game TWSG(T), the Leader is the defender and the Follower is the attacker.

As described in Section 2.1.2, both the attacker and the defender continuously interchange their roles as the sender and receiver of the signal during the ongoing confrontation, which dynamically adjusts the strategy and moves the game process forward. When the game stage T is large enough, the spoofing signal will be screened by the other party and its influence will completely disappear. The two-way signaling game will degenerate into a static game of incomplete information. The defender will continue to use defensive measures as the Leader releases signals to the outside world. The attacker will terminate the confrontational behavior and act only as the Follower to receive the signals sent by the defender. The TWSG(T) game tree is shown in Figure 6.

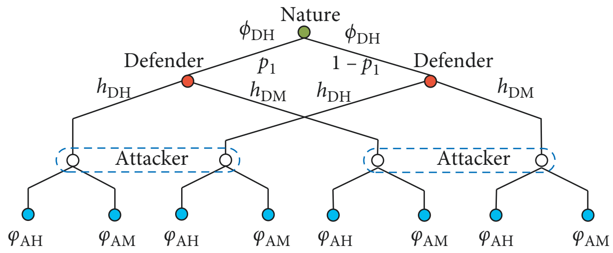


Figure 3: TWSG(1) game tree.

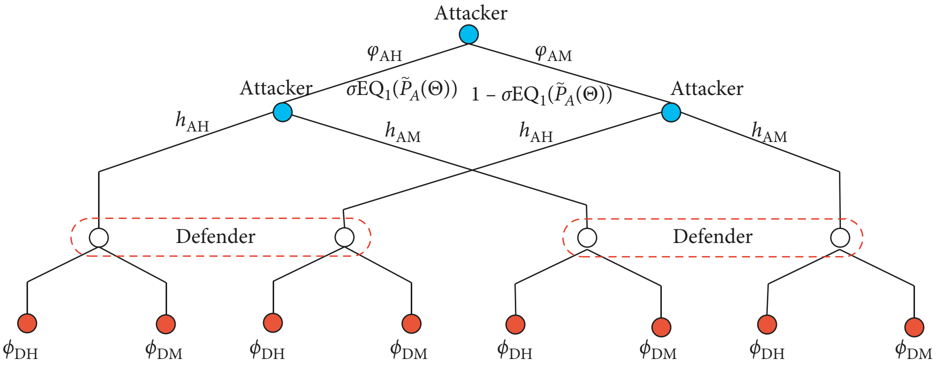


Figure 4: TWSG(2) game tree.

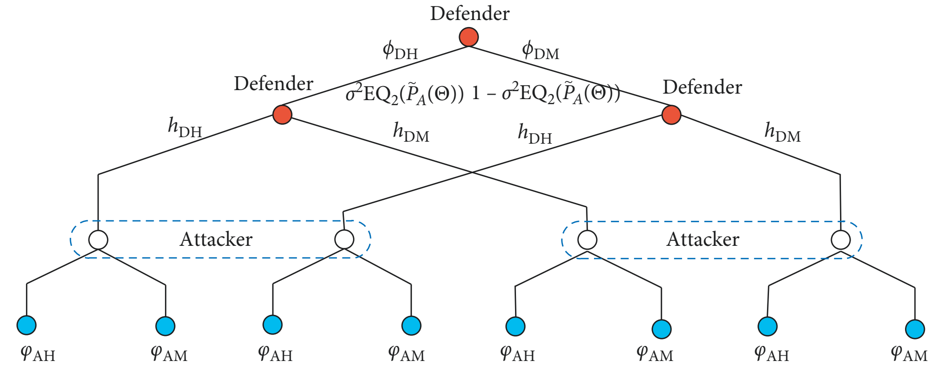


Figure 5: TWSG(3) game tree.

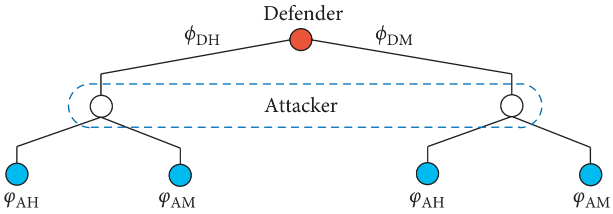


Figure 6: TWSG(T) game tree.

3.3. Defense Strategy Selection Algorithm and Comparison with Results. The algorithm for designing the active defense strategy is shown in Algorithm 1.

If the number of types on the defense side is n, the number of types on the attacker side is m, the number of game stages is t, the number of defense strategies is g, and the number of attack strategies is h, then according to Refs. [17, 21], the time complexity of the active defense strategy selection algorithm is $O(2t(mn + \max(g,h)^3))$ and the space complexity is $O(mn \max(g,h))$.

The results of our method are compared with available research on signaling games in Table 1.

The signal transmission mechanism refers to whether the signal transmission direction is one-way or two-way in the model. The attenuation of the deception signal indicates whether the model characterizes the deception signal attenuation phenomenon. The game process is used to distinguish whether the model has single-stage analysis capability or multistage analysis capability. The model expansion indicates whether the type and strategy of attack and defense in the model can be expanded. The better the

```
Input: Two-way signaling game model
Output: Active defense strategy
Initialize TWSG = $(N, \Theta, H, T, \sigma, \xi, S, P, \widetilde{P}, U)$
Calculate attack gain $U_A(d_g, a_h, t)$;
Calculate defense gain $U_D(d_g, a_h, t)$;
for $(t = 1, t \leq T, t++)$
    {
Initialize $P(\Theta \mid h)$;
Leader releases signal $H$;
Calculate {Inferred optimal dependence strategy $f^*(h)$ for Follower};
Calculate {Inferred optimal dependence strategy $h^*(\Theta)$ for Leader};
Generate posteriori inference of $\widetilde{P}_F(\Theta)$ for Follower based on Bayes' rule;
If $\widetilde{P}_F(\Theta)$ and $P(\Theta \mid h)$ not in conflict;
Then, Create $\mathrm{EQ}_t = (h^*(\Theta), f^*(h), \widetilde{P}_F(\Theta))$;
Return $S_D^*$;
$\widetilde{P}_F(\Theta) = \sigma^{t-1}\mathrm{EQ}_t(\widetilde{P}_F(\Theta))$;
    }
End
```

Algorithm 1: Active defense strategy selection algorithm.

Table 1: Comparison of research methods.

Reference	Signal transmission mechanism	Deception signal attenuation	Game process	Model expandability	Equilibrium solution	Operating costs	Performances
Ref. [16]	One-way	No	Single stage	Average	Detailed	Low	Poor
Ref. [18]	One-way	No	Single stage	Better	Simple	Low	Poor
Ref. [19]	One-way	No	Multistage	Average	Simple	High	Medium
Ref. [20]	One-way	No	Multistage	Average	Simple	High	Medium
Ref. [21]	One-way	Yes	Multistage	Good	Detailed	High	Medium
Ref. [22]	One-way	Yes	Multistage	Good	Detailed	High	Medium
This study	Two-way	Yes	Multistage	Good	Detailed	High	Good

expansion ability, the wider the scope of application of the model. The equilibrium solution of the model represents the degree of detail of the game equilibrium solution process. The more detailed the solution process is, the more practical it is. In terms of operating costs, it means time complexity and space complexity of the defense strategy selection algorithm. The lower the operation cost, the better; the better the performance, the better. Most previous studies use the one-way signal transmission mechanism to model the attack and defense process, and less consideration is given to the phenomenon of deception signal attenuation in the confrontation. Additionally, some studies are limited to single-stage game analysis. In this paper, we conduct an in-depth analysis of the two-way signal transmission mechanism, establish a two-way signaling game model, provide a detailed game equilibrium solution process, and design a defense strategy selection algorithm. In terms of signal transmission mechanisms, deception signal attenuation, and game process, this work comes closer to actual network attack and defense, and the model has better scalability and practicability. By sending deception signals from both the offense and defense sides, the parties seek to control the other party's strategy selection as well as maximize their own expected returns. This process embodies the confrontational philosophy under the condition of limited information.

Zhu et al. [34] propose two iterative reinforcement learning algorithms which allow the defender to identify optimal defenses. Reinforcement learning and signaling game model have their own advantages and disadvantages, and they should be adapted to different application scenarios. The purpose of this paper is to analyze process of network attack and defense. Reinforcement learning is a black box. Although the optimal defenses can be obtained, the analysis process and principles cannot be visualized. Using the two-way signaling game model to conduct the network attack-defense confrontation analysis, the analysis process and principles can be visulized more cleraly.

4. Real Case Application and Results Analysis

4.1. Experimental Environment and Parameter Configuration. In order to verify the feasibility and effectiveness of the proposed method, an experimental network environment was set up to carry out a simulation experiment. The experimental network was a typical business network, which was divided into three areas: external network, internal network, and DMZ. The attack and defense scenario are set as follows: the attacker located in the external network area and attempted to remotely attack the internal network zone of the enterprise intranet. The defender was the network security administrator of the enterprise and selected the active defense strategy according to the method in the paper. The topography of the experimental network is shown in Figure 7.

To ensure the availability and security of the enterprise network, a set of access control rules were set up between the network partitions as shown in Table 2. Among them, ⊕ indicates that access was allowed; × indicates that access was not allowed; and ∅ indicates that access requires certain permissions.

In general, the database server (databaseserver) stores a large amount of confidential data of the enterprise, so it was set as the target of attack in the experiment. According to the access control rules in Table 2, the attacker cannot directly access the databaseserver; however, through multiple steps, the vulnerability of the bastion server in the DMZ area can be used to obtain access to the internal network area, thereby achieving the goal of the attack.

Combined with the description of Common Vulnerabilities and Exposures (CVE) information in the information security vulnerability library [35], the vulnerability scanning tool Nessus was used to detect and discover the security vulnerabilities that existed in the experimental network. The security vulnerability of the experimental network is given in Table 3.

The attacker used the security vulnerabilities and defects that existed in the enterprise network to select an attack strategy consisting of several atomic attack actions. The defender selected a defense strategy containing different atomic defense actions in a targeted manner [36]. According to the attack and defense classification of the Lincoln Laboratory [37], we obtained the attack and defense strategies and their operating costs, as shown in Table 4.

In Refs. [17, 28], historical statistical data and expert experience were combined to provide the SDC values for different combinations of attack and defense strategies, as shown in Table 5, and to set $\xi = 0.5$ and $\sigma = 0.6$. In the ninth stage, $\xi^{t-1} = 0.5^8 \approx 0.0039$, which shows that after this stage, the gain has very less influence on the total return calculation; thus, the number of game stages was set to $T = 9$.

4.2. Equilibrium Solution and Strategy Selection

4.2.1. TWSG(1) Game Equilibrium and Defense Strategy. "Nature" selects the type of defense strategy with a probability of (0.4, 0.6). When the strategy type of the defender is φ_{DH}, the signal h_{DH} is sent out. When the type of the attack strategy is φ_{AH}, there are a total of four strategy combinations: (d_1, a_1), (d_1, a_2), (d_2, a_1), and (d_2, a_2). The SDC values for different combinations of attack-defense strategies are given in Table 5.

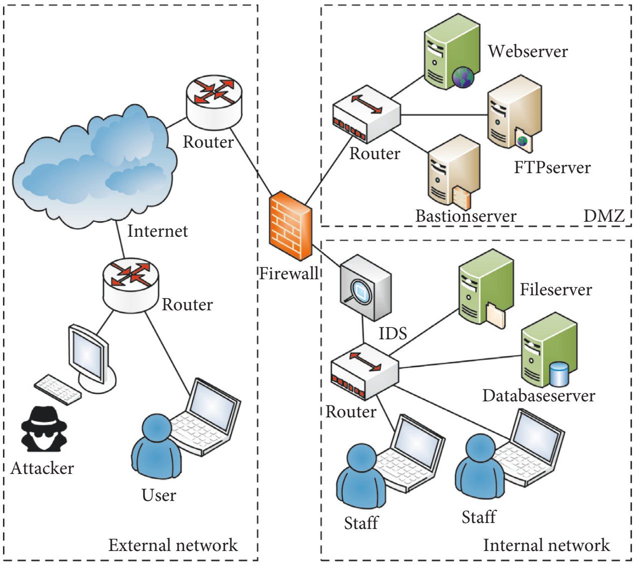


Figure 7: Topography of the experimental network.

Under the first strategy combination (d_1, a_1), the spoof signal of the attacker is DAC = 0. Thus,

$$\begin{aligned} U_A(d_1, a_1, 1) &= \mathrm{SDC}(d_1, a_1) - \mathrm{AC} - \mathrm{DAC} \\ &= 2320 - 480 - 0 = 1840. \end{aligned} \tag{2}$$

The gains for the other three strategy combinations can be calculated in the same way: $U_A(d_1, a_2, 1) = 1810$, $U_A(d_2, a_1, 1) = 1900$, and $U_A(d_2, a_2, 1) = 1770$.

Since the probability for selecting different strategies at the same attack and defense level is the same, the probability for each strategy combination is 0.25, and therefore the average gain u_{12} of the attacker under strategy type φ_{AH} is

$$\begin{aligned} u_{12} &= \overline{U}_A(\phi_{\mathrm{DH}}, \varphi_{\mathrm{AH}}, 1) \\ &= 0.25U_A(d_1, a_1, 1) + 0.25U_A(d_1, a_2, 1) \\ &\quad + 0.25U_A(d_2, a_1, 1) + 0.25U_A(d_2, a_2, 1) \\ &= 1830. \end{aligned} \tag{3}$$

Similarly, we have.

$U_D(d_1, a_1, 1) = -[\mathrm{SDC}(d_1, a_1) + \mathrm{DC} + \mathrm{DDC}] = -3000$, $U_D(d_1, a_2, 1) = -2950$, $U_D(d_2, a_1, 1) = -3020$, and $U_D(d_2, a_2, 1) = -2870$.

$$\begin{aligned} u_{11} &= \overline{U}_D(\phi_{\mathrm{DH}}, \varphi_{\mathrm{AH}}, 1) \\ &= 0.25U_D(d_1, a_1, 1) + 0.25U_D(d_1, a_2, 1) \\ &\quad + 0.25U_D(d_2, a_1, 1) + 0.25U_D(d_2, a_2, 1) \\ &= -2960. \end{aligned} \tag{4}$$

Similarly, the above method can be used to obtain the offensive and defensive gains under different combinations of strategy types.

Using the equilibrium solution algorithm of Section 3.3, a pooling equilibrium solution is obtained for TWSG(1). There are two possible combinations of strategy types:

Table 2: Access control rules.

Network region	External network	Internal network	DMZ
External network	⊕	×	⊕
Internal network	×	⊕	∅
DMZ	⊕	∅	⊕

Table 3: Security vulnerability of the experimental network.

No.	Object of action	CVE code	Threat type	Threat level
1	Webserver	CVE-2015-1635	Code injection	Extreme risk
2	Webserver	CVE-2017-7269	Buffer zone overflow	Extreme risk
3	FTPserver	CVE-2014-8517	Operating system command injection	High risk
4	Bastionserver	CVE-2014-3556	Operating system command injection	High risk
5	Fileserver	CVE-2013-4730	Buffer zone overflow	Extreme risk
6	Databaseserver	CVE-2016-6662	Authorization and access control	Extreme risk

Table 4: Attack-defense strategy and operating cost.

Atomic attack action	φ_{AH}		φ_{AM}		Atomic defense action	ϕ_{DH}		ϕ_{DM}	
	a_1	a_2	a_3	a_4		d_1	d_2	d_3	d_4
Install listener program	√	√	√	√	Uninstall listener program	√	√	√	
Remote buffer overflow	√	√	√		Buffer overflow protection	√	√		
Install delete Trojan	√				Uninstall delete Trojan	√	√		
Attack SSH on FTPServer	√	√			Restart FTPserver	√		√	√
Steal account and password	√		√	√	Change account and password		√	√	√
Raise authority	√	√			Delete suspicious account	√	√	√	
Remote code injection	√	√			Identify code injection	√			
Violent crack password			√	√	Increase password complexity	√	√	√	√
AC	480	460	240	220	DC	680	640	440	410
DAC	80	70	30	20	DDC	100	80	40	30

Table 5: SDC values for different combinations of attack-defense strategies.

d a	d_1	d_2	d_3	d_4
a_1	$\mathrm{SDC}(d_1, a_1) = 2320$	$\mathrm{SDC}(d_2, a_1) = 2380$	$\mathrm{SDC}(d_3, a_1) = 2640$	$\mathrm{SDC}(d_4, a_1) = 2680$
a_2	$\mathrm{SDC}(d_1, a_2) = 2270$	$\mathrm{SDC}(d_2, a_2) = 2230$	$\mathrm{SDC}(d_3, a_2) = 2520$	$\mathrm{SDC}(d_4, a_2) = 2570$
a_3	$\mathrm{SDC}(d_1, a_3) = 2180$	$\mathrm{SDC}(d_2, a_3) = 2120$	$\mathrm{SDC}(d_3, a_3) = 2280$	$\mathrm{SDC}(d_4, a_3) = 2320$
a_4	$\mathrm{SDC}(d_1, a_4) = 2120$	$\mathrm{SDC}(d_2, a_4) = 2080$	$\mathrm{SDC}(d_3, a_4) = 2210$	$\mathrm{SDC}(d_4, a_4) = 2260$

Option 1: the defender selects strategy type ϕ_{DH} and releases signal h_{DH}, and the attacker selects strategy type φ_{AM}. This time, $U_{11} = -2960$ and $U_{12} = 1830$.

Option 2: the defender selects strategy type ϕ_{DM} and releases signal h_{DH}, and the attacker selects strategy type φ_{AM}. At this time, $U_{11} = -2727.5$ and $U_{12} = 2037.5$.

Therefore, the defender selects option 2 as the defense strategy, designated as $(\phi_{\mathrm{DM}}, h_{\mathrm{DH}})$. The game tree of attack and defense is shown in Figure 8.

4.2.2. TWSG(2) Game Equilibrium and Defense Strategy. In the TWSG(1) equilibrium solution process, the attacker may choose either the strategy type φ_{AH} or φ_{AM}, and therefore the defender's posteriori probability of the attacker is modified to (0.5, 0.5). Using the equalization solution algorithm described in Section 3.3, the solution of TWSG(2) remains a pooling equilibrium. There are two possible combinations of strategies:

(i) The attacker selects the strategy type φ_{AH} and releases signal h_{AM}, and the defender chooses strategy type ϕ_{DM}

(ii) The attacker selects strategy type φ_{AM} and releases signal h_{AM}, and the defender selects strategy type ϕ_{DM}

Therefore, the defender selects the regular type strategy, designated as ϕ_{DM}.

4.2.3. Game Equilibrium and Defense Strategy for Stages Three through Nine. Using the above method, the game equilibrium for each stage is solved sequentially.

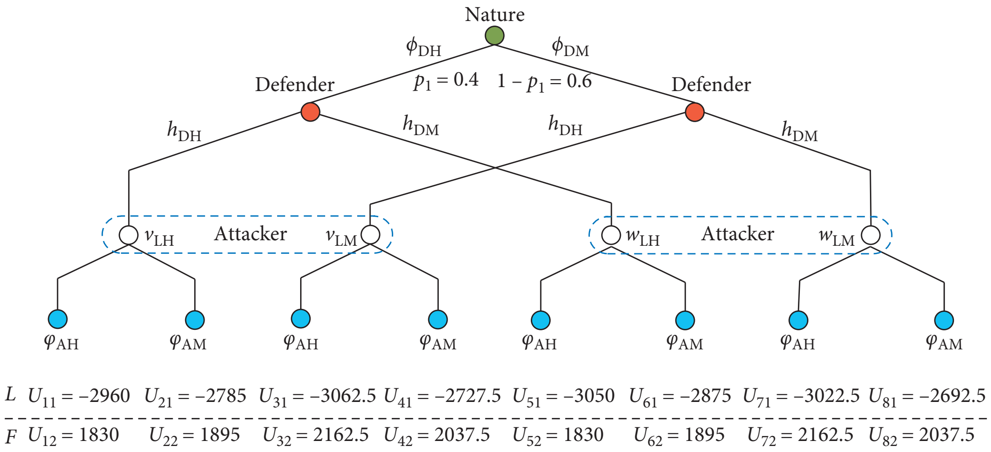

Nature
ϕDH
ϕDM
Defender
p1 = 0.4
1 − p1 = 0.6
Defender
hDH
hDM
hDH
hDM
vLH
Attacker
vLM
wLH
Attacker
wLM
φAH
φAM
φAH
φAM
φAH
φAM
φAH
φAM
L U11 = −2960 U21 = −2785 U31 = −3062.5 U41 = −2727.5 U51 = −3050 U61 = −2875 U71 = −3022.5 U81 = −2692.5
F U12 = 1830 U22 = 1895 U32 = 2162.5 U42 = 2037.5 U52 = 1830 U62 = 1895 U72 = 2162.5 U82 = 2037.5

Figure 8: Game tree of attack and defense.

For stages three through six, as shown in Table 6, the game equilibrium solution remains a pooling equilibrium, but the deceptive signal is gradually attenuated. In stages seven through nine, the deception signal is completely attenuated, the game evolves into an incomplete information static game, and the pooling equilibrium solution becomes a separating equilibrium solution. At this point, the defender selects the enhanced ϕ_{DH} as the strategy type and releases an enhanced signal h_{DH}, designated as $(\phi_{\mathrm{DH}}, h_{\mathrm{DH}})$.

4.3. Experimental Analysis. Based on the above experiments and data analysis, the following conclusions can be drawn from the general analysis of the offensive and defensive game equilibrium and the gain without considering specific parameter values.

(1) Deception signals can improve attack and defense performance.

The game equilibrium solutions for stages one through six are pooling equilibrium solutions, indicating that, in the initial stage of the offensive and defensive game, the defender may adopt the regular type of defense strategy ϕ_{DM} and confuse and mislead the attacker by releasing the spoofing signal h_{DH}. By disrupting the cognition of the attacker, the defender's own gain can be maximized at a small cost. The effectiveness of the spoofing signal should therefore be fully utilized to actively release the spoofing signal. At the same time, the ability to identify the attacking party's spoofing signals should be enhanced so that the motivation and preference of the attacker can be recognized as early as possible and a targeted active defense strategy can be implemented.

(2) The role of the spoofing signal is limited and attenuated.

As the game progresses, the spoofing signal becomes gradually attenuated. In the seventh through ninth stages of the game, the game equilibrium solution becomes a separating equilibrium solution, indicating that the function of the deception signal has completely disappeared. The defender no longer releases spoofing signals but instead increases the defensive input and adopts an enhanced defense strategy ϕ_{DH} to fight against network attacks. Therefore, when selecting the strategy, one should avoid the limitations of the spoofing signal and the attenuation process should be delayed by improving the quality of the spoofing signal. At the same time, attention should be given to collecting threat information and amplifying the limitations of the attacker's spoofing signal.

(3) Spoofing signals can delay the attack speed and reduce the suddenness of the attack.

An analysis of the first through ninth stages of the game shows that the deception signal released by the defender can delay the formation of the network kill chain and gain some reaction time for the defender. The deception signal can partially offset the time asymmetry advantage and the first-move advantage possessed by the attacker. However, due to the limitations of the spoofing signal, relying solely on the spoofing signal itself cannot completely resist network attacks. Therefore, the defending party should evolve according to the game process and use other means of defense to dynamically adjust the defense strategy to maximize its own return.

(4) Reduce security losses by enhancing defense capabilities.

We analyze the gamer's return when different strategy types are adopted. In the first through sixth stages, the defender adopts the regular type of defense strategy and the average return is −2853. In the seventh through ninth stages, the defender chooses the enhanced defense strategy type and the defender's average return is −2496. This shows that when faced with continuous high-intensity network attacks, the defending party should increase its security investment, enhance its defense capabilities, and reduce its security losses.

Table 6: Defense strategies of different stages and attack-defense returns.

Game stage	Defense role	Equilibrium type	Defense strategy	Attacker return	Defender return
TWSG (1)	Leader	Pooling equilibrium	$(\phi_{\mathrm{DM}}, h_{\mathrm{DH}})$	2037.5	−2727.5
TWSG (2)	Follower	Pooling equilibrium	ϕ_{DM}	2053.5	−2785.5
TWSG (3)	Leader	Pooling equilibrium	$(\phi_{\mathrm{DM}}, h_{\mathrm{DH}})$	2079.5	−2833.5
TWSG (4)	Follower	Pooling equilibrium	ϕ_{DM}	2112.5	−2894.5
TWSG (5)	Leader	Pooling equilibrium	$(\phi_{\mathrm{DM}}, h_{\mathrm{DH}})$	2145.5	−2920.5
TWSG (6)	Follower	Pooling equilibrium	ϕ_{DM}	2069.5	−2956.5
TWSG (7)	Leader	Separating equilibrium	$(\phi_{\mathrm{DH}}, h_{\mathrm{DH}})$	2011	−2460
TWSG (8)	Follower	Separating equilibrium	ϕ_{DH}	2038	−2492
TWSG (9)	Leader	Separating equilibrium	$(\phi_{\mathrm{DH}}, h_{\mathrm{DH}})$	2089	−2536

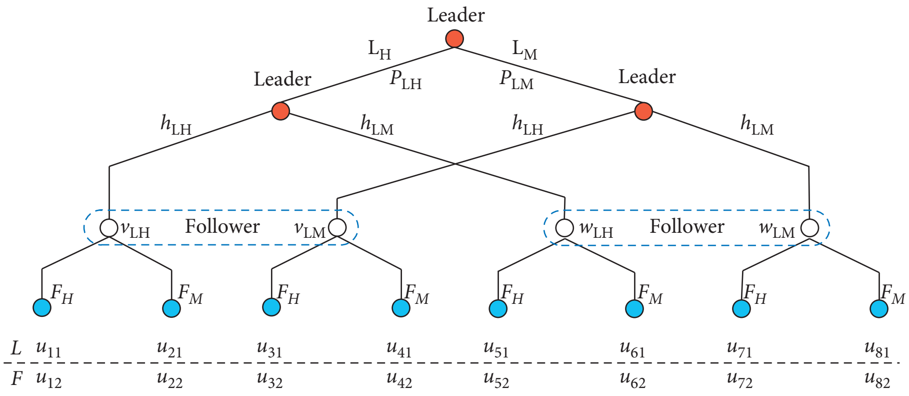


Figure 9: Single-stage signaling game tree.

5. Conclusion

Active defense is a topic at the forefront of research in the field of network security. Strategy selection is the key to defense effectiveness. Under the conditions of attack-defense confrontation and limited information, the defense party's optimal strategy is difficult to determine; however, a signaling game model is an effective way to solve this problem. To address the problem that one-way signal transmission does not conform to the actual problem of network attack and defense, we analyzed the two-way signal transmission process, constructed a two-way signaling game model, provided a multistage perfect Bayesian equilibrium solution process, and designed an active defense strategy selection algorithm in this paper. The feasibility and effectiveness of the method was verified through example applications and analysis. By analyzing the experimental results, we identified the mechanism driving the effectiveness and limitations of the deceptive signal and summarized four conclusions that guide the selection of active defense strategies. Compared with existing research, the two-way signaling game model proposed in this paper more accurately represents the offensive and defensive strategy confrontation process and more closely resembles an actual network attack and defense process. Thus, our work serves as the basis of, and provides reference to, the active defense strategy selection process under dynamic incomplete information conditions.

Appendix

Example Solution of Perfect Bayesian Equilibrium

Based on the parameter settings in this paper, the attacking party and defending party each have two strategy types and release two types of signals. The Leader type is represented by the symbols L_H and L_M, the signal space is represented by H_{LH} and H_{LM}, the Follower type is represented by the symbols F_H and F_M, $\{u_{11}, u_{21}, u_{31}, \ldots, u_{81}\}$ is the gain of the Leader, and $\{u_{12}, u_{22}, u_{32}, \ldots, u_{82}\}$ is the gain of the Follower. The single-stage signaling game tree is shown in Figure 9.

Step 1. Follower strategy calculation.

First, we assume that the posteriori inference of different signal sets on the single-stage game tree to be $P_F(\Theta \mid h)$. We then calculate the maximum return $\max_{f\in F} \sum P_F(\Theta \mid h) U_F(h^*(\Theta), f, \Theta)$.

When $H = h_{\mathrm{LH}}$,

$$\begin{aligned}
&\max_{f\in F} \sum P_F(\Theta \mid h) U_F(h^*(\Theta), f, \Theta) \\
&\quad = \max\Big\{U_F(h_{\mathrm{LH}}, F_H, L_H) \times \widetilde{P}\big(L_H \mid h_{\mathrm{LH}}\big) \\
&\qquad + U_F(h_{\mathrm{LH}}, F_H, L_M) \times \widetilde{P}\big(L_M \mid h_{\mathrm{LH}}\big), U_F(h_{\mathrm{LH}}, F_M, L_H) \\
&\qquad \times \widetilde{P}\big(L_H \mid h_{\mathrm{LH}}\big) + U_F(h_{\mathrm{LH}}, F_M, L_M) \times \widetilde{P}\big(L_M \mid h_{\mathrm{LH}}\big)\Big\} \\
&\quad = \max\{u_{12} \cdot v_{\mathrm{LH}} + u_{32} \cdot v_{\mathrm{LM}}, u_{22} \cdot v_{\mathrm{LH}} + u_{42} \cdot v_{\mathrm{LM}}\},
\end{aligned} \tag{A.1}$$

and the condition $v_{\mathrm{LH}} + v_{\mathrm{LM}} = 1$ is satisfied.

Assuming that $u_{12} \cdot v_{\mathrm{LH}} + u_{32} \cdot v_{\mathrm{LM}} = u_{22} \cdot v_{\mathrm{LH}} + u_{42} \cdot v_{\mathrm{LM}}$,

we solve and obtain $v^{*}_{\mathrm{LH}} = (u_{42} - u_{32}/u_{12} - u_{22} - u_{32} + u_{42})$, and $v^{*}_{\mathrm{LH}} \in [0, 1]$.

For $0 \leq v_{\mathrm{LH}} \leq v^{*}_{\mathrm{LH}}$, $(3) = u_{12} \cdot v_{\mathrm{LH}} + u_{32} \cdot v_{\mathrm{LM}}$ and $f^{*}(h) = F_{H}$.

For $v^{*}_{\mathrm{LH}} \leq v_{\mathrm{LH}} \leq 1$, $(3) = u_{22} \cdot v_{\mathrm{LH}} + u_{42} \cdot v_{\mathrm{LM}}$ and $f^{*}(h) = F_{L}$.

Similarly, we obtain $w^{*}_{\mathrm{LH}} = u_{82} - u_{72}/u_{52} - u_{62} - u_{72} + u_{82}$.

For $0 \leq w_{\mathrm{LH}} \leq w^{*}_{\mathrm{LH}}$, $f^{*}(h) = F_{H}$.

For $w^{*}_{\mathrm{LH}} \leq w_{\mathrm{LH}} \leq 1$, $f^{*}(h) = F_{L}$.

By repeating the above process, we calculate $f^{*}(h)$ for $H = h_{\mathrm{LM}}$.

Step 2. Leader strategy calculation.

$$\max_{h \in H} U_{L}(h, f^{*}(h), \Theta). \tag{A.2}$$

For $\Theta = L_{H}$, when $0 \leq v_{\mathrm{LH}} \leq v^{*}_{\mathrm{LH}}$ and $0 \leq w_{\mathrm{LH}} \leq w^{*}_{\mathrm{LH}}$,

$$\begin{aligned} &\max_{h \in H} U_{L}(h, f^{*}(h), \Theta) \\ &\quad = \max\{U_{L}(h_{\mathrm{LH}}, F_{H}, L_{H}), U_{L}(h_{\mathrm{LM}}, F_{H}, L_{H})\} \\ &\quad = \max\{u_{11}, u_{51}\}, \end{aligned} \tag{A.3}$$

and we obtain $h^{*}(L_{H})$.

Similarly, we obtain $h^{*}(L_{H})$ for different sections of v_{LH} and w_{LH}.

By repeating the above process, we calculate $h^{*}(L_{H})$ for $\Theta = L_{M}$.

Step 3. Calculate equilibrium solution.

We obtain $f^{*}(h)$ and $h^{*}(\Theta)$ in Step 1 and Step 2, respectively, by combining this with a priori probability PL and obtain the posteriori probability $\widetilde{P}_{F}(\Theta)$. If the calculated value of $\widetilde{P}_{F}(\Theta)$ is not in conflict with the premise hypothesis $P(\Theta \mid h)$, then the equilibrium solution is $\mathrm{EQ} = (h^{*}(\Theta), f^{*}(h), \widetilde{P}_{F}(\Theta))$.

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 they have no conflicts of interest regarding the publication of this paper.

Authors' Contributions

Xiaohu Liu and Hengwei Zhang contributed equally to this work.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant nos. 61521003 and 61572517 and in part by the Henan Science and Technology Research Project under Grant no. 182102210144.

References

[1] Z. Tian, W. Shi, Y. Wang et al., "Real-time lateral movement detection based on evidence reasoning network for edge computing environment," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4285–4294, 2019.

[2] Z. Tian, M. Li, M. Qiu, Y. Sun, and S. Su, "Block-DEF: a secure digital evidence framework using blockchain," *Information Sciences*, vol. 491, pp. 151–165, 2019.

[3] R. K. Sharma, B. Issac, and H. K. Kalita, "Intrusion detection and response system inspired by the defense mechanism of plants," *IEEE Access*, vol. 7, pp. 52427–52439, 2019.

[4] D. E. Denning and E. Dorothy, "Framework and principles for active cyber defense," *Computers & Security*, vol. 40, pp. 108–113, 2014.

[5] S. Huang, H. Zhang, J. Wang, and J. Huang, "Markov differential game for network defense decision making method," *IEEE Access*, vol. 6, pp. 39621–39634, 2018.

[6] C. T. Do, N. H. Tran, C. Hong et al., "Game theory for cyber security and privacy," *ACM Computing Surveys*, vol. 50, no. 2, pp. 1–37, 2017.

[7] A. Farraj, E. Hammad, A. A. Daoud, and D. Kundur, "A game theoretic analysis of cyber switching attacks and mitigation in smart grid systems," *IEEE Transactions on Smart Grid*, vol. 7, no. 4, pp. 1846–1855, 2016.

[8] H. Hu, Y. Liu, H. Zhang, and R. Pan, "Optimal network defense strategy selection based on incomplete information evolutionary game," *IEEE Access*, vol. 6, pp. 29806–29821, 2018.

[9] J. Huang, H. Zhang, and J. Wang, "Markov evolutionary games for network defense strategy selection," *IEEE Access*, vol. 5, pp. 19505–19516, 2017.

[10] C. Lei, D.-H. Ma, and H.-Q. Zhang, "Optimal strategy selection for moving target defense based on Markov game," *IEEE Access*, vol. 5, pp. 156–169, 2017.

[11] Z. Tian, X. Gao, S. Su, J. Qiu, X. Du, and M. Guizani, "Evaluating reputation management schemes of internet of vehicles based on evolutionary game theory," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 5971–5980, 2019.

[12] Y.-L. Liu, D.-G. Feng, L.-H. Wu, and Y.-F. Lian, "Performance evaluation of worm attack and defense strategies based on static Bayesian game," *Journal of Software*, vol. 23, no. 3, pp. 712–723, 2012.

[13] L. Wangqun, H. Wang, and L. Jiahong, "Research on active defense technology in network security based on non-cooperative dynamic game theory," *Journal of Computer Research and Development*, vol. 48, no. 2, pp. 306–316, 2011.

[14] W. Jin-Dong, "Active defense strategy selection based on the static Bayesian game," *Journal of Xidian University*, vol. 43, no. 1, pp. 144–150, 2016.

[15] C. Lei, H.-Q. Zhang, L.-M. Wan, L. Liu, and D.-H. Ma, "Incomplete information Markov game theoretic approach to strategy generation for moving target defense," *Computer Communications*, vol. 116, pp. 184–199, 2018.

[16] W. Casey, A. Kellner, P. Memarmoshrefi, J. A. Morales, and B. Mishra, "Deception, identity, and security," *Communications of the ACM*, vol. 62, no. 1, pp. 85–93, 2018.

[17] H. Xu, R. Freeman, V. Conitzer, S. Dughmi, and M. Tambe, "Signaling in Bayesian stackelberg games," in *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, pp. 150–158, Singapore, January 2016.

[18] H. Jihong and Y. Dingkun, "Defense policies selection method based on attack-defense signaling game model," *Journal on Communications*, vol. 36, no. 4, pp. 121–132, 2016.

[19] X. Feng, Z. Zheng, D. Cansever, A. Swami, and P. Mohapatra, "A signaling game model for moving target defense," in *Proceedings of the 2017 IEEE Conference on Computer Communications, INFOCOM 2017*, pp. 1–9, Atlanta, GA, USA, May 2017.

[20] X. Gao and Y. Zhu, "DDoS defense mechanism analysis based on signaling game model," in *Proceedings of the 2013 5th International Conference on Intelligent Human-Machine Systems and Cybernetics*, pp. 414–417, Hangzhou, China, August 2013.

[21] Z. Hengwei, L. Tao, W. Jindong, and H. Jihong, "Optimal active defense using dynamic multi-stage signaling game," *China Communications*, vol. 12, no. 2, pp. 114–122, 2015.

[22] X. Chen, X. Liu, L. Zhang, and C. Tang, "Optimal defense strategy selection for spear-phishing attack based on a multistage signaling game," *IEEE Access*, vol. 7, pp. 19907–19921, 2019.

[23] Y. Yang, B. Che, Y. Zeng, Y. Cheng, and C. Li, "MAIAD: a multistage asymmetric information attack and defense model based on evolutionary game theory," *Symmetry*, vol. 11, no. 2, pp. 215–229, 2019.

[24] M. O. Sayin and T. basar, "Deception as defense framework for cyber-physical systems," 2019, https://arxiv.org/abs/1902.01364.

[25] J. C. Harsanyi, "Games with incomplete information played by "Bayesian" players," in *Game Theory*, pp. 154–170, Springer, Dordrecht, Netherlands, 1982.

[26] Q. Zhu, "Game theory for cyber deception: a tutorial," 2019, https://arxiv.org/abs/1903.01442.

[27] S. Clio, "Cyber kill chain based on threat taxonomy and its application on cyber common operational picture," in *Proceedings of the 2018 International Conference on Cyber Situational Awareness*, pp. 1–8, Data Analytics and Assessment (Cyber SA), Glasgow, Scotland, June 2018.

[28] J. Pawlick, E. Colbert, and Q. Zhu, "Modeling and analysis of leaky deception using signaling games with evidence," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 7, pp. 1871–1886, 2019.

[29] E. Al-Shaer, J. Wei, W. Kevin, Hamlen, and C. Wang, *Autonomous Cyber Deception*, Springer International Publishing, Berlin, Germany, 2019.

[30] T. Zhang, L. Huang, J. Pawlick, and Q. Zhu, "Game-theoretic analysis of cyber deception: evidence-based strategies and dynamic risk mitigation," 2019, https://arxiv.org/abs/1902.03925.

[31] W. Yuzhuo, Y. Jianye, and Q. Wen, "Evolutionary game model and analysis methods for network group behavior," *Chinese Journal of Computers*, vol. 38, no. 2, pp. 282–300, 2015.

[32] J. Wei and F. Bing-Xing, "Defense strategies selection based on attack-defense game model," *Journal of Computer Research and Development*, vol. 47, no. 12, pp. 714–723, 2014.

[33] D. Fudenberg and J. Tirole, "Perfect Bayesian equilibrium and sequential equilibrium," *Journal of Economic Theory*, vol. 53, no. 2, pp. 236–260, 1991.

[34] M. Zhu, Z. Hu, and P. Liu, "Reinforcement learning algorithms for adaptive cyber defense against heartbleed," in *Proceedings of the 2014 in Proceedings of the First ACM Workshop on Moving Target Defense—MTD'14*, Scottsdale, AR, USA, November 2014.

[35] National Vulnerability Database of Information Security, https://nvd.nist.gov/.

[36] Q. Tan, Y. Gao, J. Shi, X. Wang, B. Fang, and Z. Tian, "Toward a comprehensive insight into the eclipse attacks of tor hidden services," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1584–1593, 2019.

[37] L. Gordon, M. Loeb, W. Lucyshyn, and R. Richardson, "CSI/FBI computer crime and security survey," in *Proceedings of the Computer Security Institute*, pp. 48–64, San Francisco, CA, USA, 2015.

Hindawi
Security and Communication Networks
Volume 2019, Article ID 2315627, 12 pages
https://doi.org/10.1155/2019/2315627

Research Article

Seeking Best-Balanced Patch-Injecting Strategies through Optimal Control Approach

Kaifan Huang,[1] **Pengdeng Li**,[1] **Lu-Xing Yang**,[2] **Xiaofan Yang**,[1] **and Yuan Yan Tang**[3]

[1] *School of Big Data & Software Engineering, Chongqing University, Chongqing 400044, China*
[2] *School of Information Technology, Deakin University, Melbourne, VIC 3125, Australia*
[3] *Department of Computer and Information Science, The University of Macau, Macau*

Correspondence should be addressed to Xiaofan Yang; xfyang1964@gmail.com

Received 28 February 2019; Revised 4 May 2019; Accepted 22 May 2019; Published 10 June 2019

Guest Editor: Akbar S. Namin

To restrain escalating computer viruses, new virus patches must be constantly injected into networks. In this scenario, the patch-developing cost should be balanced against the negative impact of virus. This article focuses on seeking best-balanced patch-injecting strategies. First, based on a novel virus-patch interactive model, the original problem is reduced to an optimal control problem, in which (a) each admissible control stands for a feasible patch-injecting strategy and (b) the objective functional measures the balance of a feasible patch-injecting strategy. Second, the solvability of the optimal control problem is proved, and the optimality system for solving the problem is derived. Next, a few best-balanced patch-injecting strategies are presented by solving the corresponding optimality systems. Finally, the effects of some factors on the best balance of a patch-injecting strategy are examined. Our results will be helpful in defending against virus attacks in a cost-effective way.

1. Introduction

Computer networks bring huge convenience to our work and life [1, 2]. Meanwhile, digital viruses can propagate rapidly through computer networks, posing a severe threat to human society. For example, Wanna Decryptor, the notorious ransomware, has recently swept across the globe, leading to massive computer paralysis [3]. Consequently, the problem of how to mitigate the negative impact of computer virus in a cost-effective way has long been a hotspot of research in the field of cyber security [4].

To restrain evolving computer viruses, new virus patches must be constantly injected into networks. In this scenario, there is an obvious conflict between the patch-developing cost and the impact of virus; reducing the former would increase the latter, whereas mitigating the latter would enhance the former. Therefore, the patch-developing cost should be balanced against the impact of virus. We refer to a dynamic patch-injecting strategy that achieves the best balance between the two aspects as a best-balanced patch-injecting strategy, and we refer to the problem of seeking best-balanced patch-injecting strategies as the *virus-patch tradeoff (VPT) problem*. Solving the VPT problem would be helpful in defending against virus attacks in a cost-effective way.

This article addresses the VPT problem. First, based on a novel virus-patch interactive model, the original problem is reduced to an optimal control problem which we refer to as the VPT control problem, in which (a) each admissible control stands for a feasible patch-injecting strategy and (b) the objective functional measures the balance of a feasible patch-injecting strategy. Second, the solvability of the VPT control problem is shown, and the optimality system for solving the VPT control problem is derived. Next, a few best-balanced patch-injecting strategies are given by solving the corresponding optimality systems. Finally, the effects of some factors on the best balance of a patch-injecting strategy are examined.

The remaining materials are organized in this fashion: Section 2 reviews the related work. Sections 3 and 4 establish and solve the VPT control problem, respectively. Section 5 illustrates how to solve the VPT control problem, and Section 6 examines the effects of some factors on the best balance. This work is summarized by Section 7.

2. Related Work

In order to solve the VPT problem, the expected total loss of all network users resulting from a patch-injecting strategy must be estimated [5, 6]. As this quantity relies on the expected network states at all times, we need to characterize the evolutionary process of the expected network state. The resulting evolutionary model is essentially a propagation model that captures the interactive propagation of viruses and patches [7, 8]. In the available literature, propagation models of this kind are referred to as Susceptible-Infected-Patched-Susceptible (SIPS) models.

Compartmental propagation models are propagation models in which all nodes of the same state are classified as a class, with the goal of understanding the evolutionary trend of the size or fraction of each class [9]. Compartmental models are suited to capturing propagation phenomena occurring on homogeneously mixed networks but fail to characterize propagation phenomena occurring on highly heterogeneous networks. The compartmental SIPS models proposed in [10–13] take patch forwarding into account but leave patch injection out of consideration. Very recently, [14] proposed a compartmental SIPS model with static patch-injecting mechanism and thereby assessed the effectiveness of patch injection.

Node-level propagation models are propagation models in which each node is regarded as a separate class, with the goal of gaining insight into the evolutionary trend of the expected network state [15, 16]. One striking advantage of node-level propagation models is that they can accurately characterize propagation phenomena occurring on arbitrary networks. With the progress of wireless and mobile communication technologies, most existing computer networks admit an irregular topology [17–19]. As a result, a number of node-level computer virus propagation models have been advised [20–25]. In particular, [26] introduced a node-level SIPS model with no patch injection. Recently, [27] proposed a node-level SIPS model with dynamic patch-injecting mechanism and thereby addressed a problem that is something like the VPT problem through differential game approach. In our opinion, this work has two weaknesses: (i) It is assumed that the network defender is aware of the total attack budget of all relevant cyber malefactors. However, in practice the budget is usually unknown to the defender. (ii) It is assumed that new patches can be injected into any network node. Due to the limited network bandwidth, in practice new patches are typically injected into a small subset of nodes and then forwarded to the unpatched nodes [28].

Optimal control theory [29, 30] provides a powerful tool for studying the problem of how to contain the prevalence of computer virus in a cost-effective way [31–35]. In view of the defects of the research approach used in [27], in this paper we deal with the VPT problem through optimal control approach. For this purpose, we propose a novel node-level SIPS model with dynamic patch-injecting mechanism, where new patches can be injected into only a small subset of nodes. Thereby, we accurately estimate the expected total loss of all network users. On this basis, we reduce the VPT problem to an optimal control problem and then solve the problem by means of optimal control theory. Our optimal control model is promising, because, by collecting and analyzing the relevant actual data, the model parameters can be estimated quite accurately.

3. The Modeling of the VPT Problem

This section is devoted to the modeling of the VPT problem. First, we introduce basic terms and notations. Second, we establish a node-level SIPS model. Finally, we model the VPT problem as an optimal control problem.

3.1. Terms and Notations. Consider a computer network with N nodes labeled v_1 through v_N. Let $G = (V, E)$ denote the topology of the network, i.e., $V = \{v_1, v_2, \ldots, v_N\}$, and each edge stands for a communication link between the two endpoints. Let $\mathbf{A} = (a_{ij})_{N\times N}$ denote the adjacency matrix of G, i.e, $a_{ij} = 1$ or 0 according as $\{v_i, v_j\} \in E$ or not. Suppose new computer viruses can be injected into any node of the network and can propagate over the network, and suppose new virus patches can be injected into only the node subset $U = \{v_1, v_2, \ldots, v_M\}$ of the network and can be forwarded to other nodes through the network.

Consider the finite time horizon $[0, T]$. Assume each and every node of the network is in one of three possible states: *susceptible*, *infected*, and *patched*. Susceptible nodes are nodes that are not infected with any virus but have not received the newest patch. This implies these nodes are vulnerable to new viruses. Infected nodes are nodes that are infected with some virus. Patched nodes are nodes that are not infected with any virus and have received the newest patch. This implies that these nodes possess temporary immunity to new viruses. Let $X_i(t) = 0$, 1, and 2 denote that the node v_i is susceptible, infected, and patched at time t, respectively. Then the state of the network at time t can be characterized by the vector

$$\mathbf{X}(t) = (X_1(t), \ldots, X_N(t)). \tag{1}$$

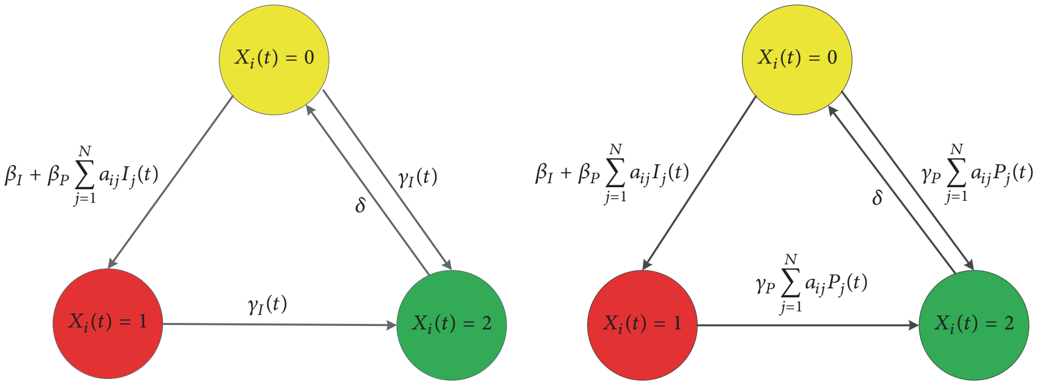
Let $S_i(t)$, $I_i(t)$, and $P_i(t)$ denote the probabilities of the node v_i being susceptible, infected, and patched at time t, respectively.

$$\begin{aligned} S_i(t) &= \Pr\{X_i(t) = 0\}, \\ I_i(t) &= \Pr\{X_i(t) = 1\}, \\ P_i(t) &= \Pr\{X_i(t) = 2\}. \end{aligned} \tag{2}$$

Since $S_i(t) = 1 - I_i(t) - P_i(t)$, the expected state of the network at time t can be characterized by the vector

$$\mathbf{E}(t) = (I_1(t), \ldots, I_N(t), P_1(t), \ldots, P_N(t)). \tag{3}$$

3.2. A Virus-Patch Interactive Model. In order to establish a virus-patch interactive model, we introduce a set of assumptions as follows.


$X_i(t) = 0$
$\beta_I + \beta_P \sum_{j=1}^{N} a_{ij} I_j(t)$
$\gamma_I(t)$
δ
$X_i(t) = 1$
$\gamma_I(t)$
$X_i(t) = 2$
$X_i(t) = 0$
$\beta_I + \beta_P \sum_{j=1}^{N} a_{ij} I_j(t)$
δ
$\gamma_P \sum_{j=1}^{N} a_{ij} P_j(t)$
$\gamma_P \sum_{j=1}^{N} a_{ij} P_j(t)$
$X_i(t) = 1$
$X_i(t) = 2$

(a) $v_i \in U$

(b) $v_i \in V - U$

FIGURE 1: Diagram of the assumptions (A_1)-(A_5).

(A_1) Due to virus injection, each susceptible node gets infected at the average rate β_I which we refer to as the *virus injection rate*.

(A_2) Due to virus propagation, the susceptible node v_i gets infected at time t at the average rate $\beta_P \sum_{j=1}^{N} a_{ij} I_j(t)$, where β_P is a constant which we refer to as the *virus propagation rate*.

(A_3) Due to patch injection, each unpatched node in U gets patched at time t at the average rate $\gamma_I(t)$ which we refer to as the *patch injection rate* at time t.

(A_4) Due to patch forwarding, the unpatched node $v_i \in V - U$ gets patched at time t at the average rate $\gamma_P \sum_{j=1}^{N} a_{ij} P_j(t)$, where γ_P is a constant which we refer to as the *patch forwarding rate*.

(A_5) Due to appearance of new viruses, each patched node becomes susceptible at the average rate δ which we refer to as the *patch failure rate*.

Remark 1. The virus injection rate, the virus propagation rate, the patch forwarding rate, and the patch failure rate can be estimated accurately by collecting and analyzing the relevant historical data. All patch injection rates are under control of the network defender.

Figure 1 shows the above assumptions schematically.

Based on the above assumptions, the expected network state evolves over time according to the following differential dynamical system:

$$\begin{aligned}
\frac{dI_i(t)}{dt} &= \left[\beta_I + \beta_P \sum_{j=1}^{N} a_{ij} I_j(t)\right] \left[1 - I_i(t) - P_i(t)\right] \\
&\quad - \gamma_I(t) I_i(t), \quad 0 \le t \le T, \ v_i \in U, \\
\frac{dP_i(t)}{dt} &= \gamma_I(t) \left[1 - P_i(t)\right] - \delta P_i(t), \\
&\qquad 0 \le t \le T, \ v_i \in U, \\
\frac{dI_i(t)}{dt} &= \left[\beta_I + \beta_P \sum_{j=1}^{N} a_{ij} I_j(t)\right] \left[1 - I_i(t) - P_i(t)\right] \\
&\quad - \gamma_P I_i(t) \sum_{j=1}^{N} a_{ij} P_j(t), \\
&\qquad 0 \le t \le T, \ v_i \in V - U, \\
\frac{dP_i(t)}{dt} &= \gamma_P \left[1 - P_i(t)\right] \sum_{j=1}^{N} a_{ij} P_j(t) - \delta P_i(t), \\
&\qquad 0 \le t \le T, \ v_i \in V - U, \\
\mathbf{E}(0) &= \mathbf{E}_0.
\end{aligned} \tag{4}$$

This is a novel SIPS model, which can be written in matrix-vector notation as

$$\begin{aligned}
\frac{d\mathbf{E}(t)}{dt} &= \mathbf{F}\left(\mathbf{E}(t), \gamma_I(t)\right), \quad 0 \le t \le T, \\
\mathbf{E}(0) &= \mathbf{E}_0.
\end{aligned} \tag{5}$$

3.3. The Modeling of the VPT Problem. The function γ_I defined by $\gamma_I(t)$, $t \in [0, T]$, is under control of the network defender. We refer to the function as a *patch-injecting strategy*. Let $L[0, T]$ denote the set of all Lebesgue integrable functions defined on the interval $[0, T]$ [36]. Henceforth, we assume the set of all allowable patch-injecting strategies is

$$\Gamma = \left\{\gamma_I \in L[0, T] \mid \underline{\gamma_I} \le \gamma_I(t) \le \overline{\gamma_I}, \ 0 \le t \le T\right\}. \tag{6}$$

We refer to $\underline{\gamma_I}$ as the *minimum allowable patch injection rate*, $\overline{\gamma_I}$ as the *maximum allowable patch injection rate*.

Let $C(t)$ denote the cost per unit time at time t for patch developing. Obviously, $C(t)$ is increasing with $\gamma_I(t)$. In this paper we simply assume that $C(t)$ is linearly proportional to $\gamma_I(t)$. That is, $C(t) = c\gamma_I(t)$, where c is a constant which we refer to as the *cost coefficient*. As a result, the total patch-developing cost is $\int_0^T c\gamma_I(t)dt$ units.

Remark 2. In practice, $C(t)$ may be dependent on $\gamma_I(t)$ in a more complex way. For example, $C(t)$ may be proportional to the square of $\gamma_I(t)$. That is, $C(t) = c\gamma_I^2(t)$, where c is a constant. If this is the case, the total patch-developing cost would be $\int_0^T c\gamma_I^2(t)dt$ units. The exact form in which $C(t)$ depends on $\gamma_I(t)$ is yet to be determined through analysis of massive actual data. Nevertheless, our research approach can easily be applied to any other dependence relationship.

On the other hand, we assume that the average loss per unit time caused by the infected node v_i is w_i unit. Then, the expected total loss of all network users is $\int_0^T \sum_{i=1}^N w_i I_i(t)dt$ units. Let $\mathbf{w} = (w_1, \dots, w_N)$. Therefore, we get a measure of the balance of a patch-injecting strategy γ_I as follows.

$$J(\gamma_I) = \int_0^T c\gamma_I(t)\,dt + \int_0^T \sum_{i=1}^N w_i I_i(t)\,dt. \qquad (7)$$

By combining the above discussions, the VPT problem is reduced to the following optimal control problem:

$$\begin{aligned} \underset{\gamma_I \in \Gamma}{\text{Minimize}} \quad & J(\gamma_I) = \int_0^T c\gamma_I(t)\,dt + \int_0^T \sum_{i=1}^N w_i I_i(t)\,dt \\ \text{subject to} \quad & \frac{d\mathbf{E}(t)}{dt} = \mathbf{F}(\mathbf{E}(t), \gamma_I(t)), \quad 0 \le t \le T, \\ & \mathbf{E}(0) = \mathbf{E}_0. \end{aligned} \qquad (8)$$

We refer to the optimal control problem as the *VPT control problem*. In this problem, each admissible control stands for a feasible patch-injecting strategy, and the objective functional measures the balance of a feasible patch-injecting strategy. Each instance of the VPT control problem is given by the 11-tuple

$$\mathscr{M} = \left(G \mid U, \beta_I, \beta_P, \gamma_P, \delta, \underline{\gamma_I}, \overline{\gamma_I}, c, \mathbf{w}, T, \mathbf{E}_0\right). \qquad (9)$$

4. Theoretical Study of the VPT Control Problem

This section is dedicated to the theoretical study of the VPT control problem. First, we show that the problem is solvable. Second, we present a method for solving this problem.

4.1. Solvability. Let $L(\mathbf{E}, \gamma) = c\gamma_I + \sum_{i=1}^N w_i I_i$. We have the following lemma [30].

Lemma 3. *The VPT game problem (8) admits an optimal control if the following five conditions are met.*

(C_1) *Γ is closed and convex.*

(C_2) *There is $\gamma_I \in \Gamma$ such that the differential system $d\mathbf{E}(t)/dt = \mathbf{F}(\mathbf{E}(t), \gamma_I(t))$ $(0 \le t \le T)$ is solvable.*

(C_3) *$\mathbf{F}(\mathbf{E}, \gamma_I)$ is bounded by a linear function in* $\mathbf{E}$.

(C_4) *$L(\mathbf{E}, \gamma_I)$ is concave on Γ.*

(C_5) *$L(\mathbf{E}, \gamma_I) \ge c_1\gamma_I^\rho + c_2$ for some $\rho > 1$, $c_1 > 0$ and c_2.*

The solvability of the VPT control problem is guaranteed by the following theorem.

Theorem 4. *The VPT control problem (8) admits an optimal control.*

Proof. (a) Let γ_I be a limit point of Γ. Then there is a sequence of points in Γ, denoted $\gamma_I^{(1)}, \gamma_I^{(2)}, \dots$, that approaches γ_I. As $L[0,T]$ is complete [36], we get that $\gamma_I \in L[0,T]$. As $\underline{\gamma_I} \le \gamma_I = \lim_{n\longrightarrow\infty}\gamma_I^{(n)} \le \overline{\gamma_I}$, we get that $\gamma_I \in \Gamma$. So, Γ is closed. (b) Let $\gamma_I^{(1)}, \gamma_I^{(2)} \in \Gamma$, $0 < \alpha < 1$. $\gamma_I = \alpha\gamma_I^{(1)} + (1-\alpha)\gamma_I^{(2)}$. As $L[0,T]$ is a real vector space [36], we have $\gamma_I \in L[0,T]$. As $\underline{\gamma_I} \le \gamma_I \le \overline{\gamma_I}$, we get that $\gamma_I \in \Gamma$. So, Γ is convex. (c) As $\mathbf{F}(\mathbf{E}, \overline{\gamma_I})$ is continuously differentiable, it follows from Continuation Theorem for Differential Systems [37] that the differential system $d\mathbf{E}(t)/dt = \mathbf{F}(\mathbf{E}(t), \overline{\gamma_I})$ $(0 \le t \le T)$ is solvable. (d) Obviously, for $v_i \in U$,

$$\begin{aligned} -\overline{\gamma_I} I_i &\le \left(\beta_I + \beta_P \sum_{j=1}^N a_{ij} I_j\right)(1 - I_i - P_i) - \gamma_I I_i \\ &\le \beta_I + \beta_P \sum_{j=1}^N a_{ij} I_j, \\ -\delta P_i &\le \gamma_I (1 - P_i) - \delta P_i \le \overline{\gamma_I} - \delta P_i, \end{aligned} \qquad (10)$$

for $v_i \in V - U$,

$$\begin{aligned} -\gamma_P \sum_{j=1}^N a_{ij} P_j &\le \left(\beta_I + \beta_P \sum_{j=1}^N a_{ij} I_j\right)(1 - I_i - P_i) \\ &\quad - \gamma_P I_i \sum_{j=1}^N a_{ij} P_j \le \beta_I + \beta_P \sum_{j=1}^N a_{ij} I_j, \\ -\delta P_i &\le \gamma_P (1 - P_i) \sum_{j=1}^N a_{ij} P_j - \delta P_i \le \gamma_P \sum_{j=1}^N a_{ij} P_j. \end{aligned} \qquad (11)$$

(e) Let $\gamma_I^{(1)}, \gamma_I^{(2)} \in \Gamma$, $0 < \alpha < 1$. As

$$\begin{aligned} & L\left(\mathbf{E}, \alpha\gamma_I^{(1)} + (1-\alpha)\gamma_I^{(2)}\right) \\ &\quad = \alpha L\left(\mathbf{E}, \gamma_I^{(1)}\right) + (1-\alpha) L\left(\mathbf{E}, \gamma_I^{(2)}\right), \end{aligned} \qquad (12)$$

we get that $L(\mathbf{E}, \gamma_I)$ is convex with respect to γ_I. (f) Obviously, $L(\mathbf{E}, \gamma_I) \ge c\gamma_I \ge (c/\overline{\gamma_I}){\gamma_I}^2$. Hence, the five conditions in Lemma 3 are met. By Lemma 3, the VPT control problem admits an optimal control.

This theorem implies that the VPT problem admits a best-balanced patch-injecting strategy. □

4.2. The Optimality System. According to optimal control theory [29], when the solvability of an optimal control problem is guaranteed, we may solve the problem by solving the optimality system associated with the problem. Now, let us derive the optimality system associated with the VPT control problem (8). The associated Hamiltonian is

$$\begin{aligned} H(\mathbf{E}, \gamma_I, \mathbf{p}) = {} & c\gamma_I + \sum_{i=1}^{N} w_i I_i + \sum_{i=1}^{M} \mu_i \left[\gamma_I (1 - P_i) - \delta P_i\right] \\ & + \sum_{i=M+1}^{N} \mu_i \left[\gamma_P (1 - P_i) \sum_{j=1}^{N} a_{ij} P_j - \delta P_i\right] \\ & + \sum_{i=1}^{M} \lambda_i \left[\left(\beta_I + \beta_P \sum_{j=1}^{N} a_{ij} I_j\right)(1 - I_i - P_i) - \gamma_I I_i\right] \\ & + \sum_{i=M+1}^{N} \lambda_i \left[\left(\beta_I + \beta_P \sum_{j=1}^{N} a_{ij} I_j\right)(1 - I_i - P_i) \right. \\ & \left. - \gamma_P I_i \sum_{j=1}^{N} a_{ij} P_j\right], \end{aligned} \tag{13}$$

where $\mathbf{p} = \mathbf{p}(t) = (\lambda_1(t), \ldots, \lambda_N(t), \mu_1(t), \ldots, \mu_N(t))$ $(0 \le t \le T)$ is the adjoint.

The following result is a necessary condition for the optimal control of the VPT control problem.

Theorem 5. *Suppose γ_I is an optimal control for the VPT control problem (8). Let $\mathbf{E}$ be the solution to the differential system (5). Then there exists $\mathbf{p}$ with $\mathbf{p}(T) = \mathbf{0}$ such that*

$$\begin{aligned} & \frac{d\lambda_i(t)}{dt} \\ & = -w_i + \lambda_i(t) \left[\beta_I + \beta_P \sum_{j=1}^{N} a_{ij} I_j(t) + \gamma_I(t)\right] \\ & \quad - \beta_P \sum_{j=1}^{N} a_{ij} \lambda_j(t) \left[1 - I_j(t) - P_j(t)\right], \\ & \qquad 0 \le t \le T, \ v_i \in U, \\ & \frac{d\lambda_i(t)}{dt} \\ & = -w_i + \lambda_i(t) \left[\beta_I + \beta_P \sum_{j=1}^{N} a_{ij} I_j(t) + \gamma_P \sum_{j=1}^{N} a_{ij} P_j(t)\right] \\ & \quad - \beta_P \sum_{j=1}^{N} a_{ij} \lambda_j(t) \left[1 - I_j(t) - P_j(t)\right], \\ & \qquad 0 \le t \le T, \ v_i \in V - U, \\ & \frac{d\mu_i(t)}{dt} \\ & = \mu_i(t) \left[\delta + \gamma_I(t)\right] + \lambda_i(t) \left[\beta_I + \beta_P \sum_{j=1}^{N} a_{ij} I_j(t)\right] \\ & \quad + \gamma_P \sum_{j=M+1}^{N} a_{ij} \left\{I_j(t) \lambda_j(t) + \left[1 - P_j(t)\right] \mu_j(t)\right\}, \\ & \qquad 0 \le t \le T, \ v_i \in U, \\ & \frac{d\mu_i(t)}{dt} \\ & = \mu_i(t) \left[\delta + \gamma_P \sum_{j=1}^{N} a_{ij} P_j(t)\right] \\ & \quad + \lambda_i(t) \left[\beta_I + \beta_P \sum_{j=1}^{N} a_{ij} I_j(t)\right] \\ & \quad + \gamma_P \sum_{j=M+1}^{N} a_{ij} \left\{I_j(t) \lambda_j(t) + \left[1 - P_j(t)\right] \mu_j(t)\right\}, \\ & \qquad 0 \le t \le T, \ v_i \in V - U, \\ & \gamma_I(t) \\ & = \begin{cases} \underline{\gamma_I} & if \ c + \sum_{i=1}^{M} \mu_i(t) \left[1 - P_i(t)\right] > \sum_{i=1}^{M} I_i(t) \lambda_i(t), \\ \overline{\gamma_I} & if \ c + \sum_{i=1}^{M} \mu_i(t) \left[1 - P_i(t)\right] < \sum_{i=1}^{M} I_i(t) \lambda_i(t), \end{cases} \\ & \qquad 0 \le t \le T. \end{aligned} \tag{14}$$

Proof. According to Pontryagin Minimum Principle [29], there exists $\mathbf{p}$ such that

$$\begin{aligned} \frac{d\lambda_i(t)}{dt} &= -\frac{\partial H(\mathbf{E}(t), \gamma_I(t), \mathbf{p}(t))}{\partial I_i}, \\ \frac{d\mu_i(t)}{dt} &= -\frac{\partial H(\mathbf{E}(t), \gamma_I(t), \mathbf{p}(t))}{\partial P_i}, \\ & \qquad t \in [0, T], \ 1 \le i \le N. \end{aligned} \tag{15}$$

Thus, the first $2N$ equations in the system (14) follow by direct calculations. As the terminal cost is unspecified and the final state is free, the transversality condition $\mathbf{p}(T) = \mathbf{0}$ holds true. Again by Pontryagin Minimum Principle, we have

$$\gamma_I(t) \in \arg\min_{\gamma_I^* \in \Gamma} H(\mathbf{E}(t), \gamma_I^*(t), \mathbf{p}(t)), \quad 0 \le t \le T. \tag{16}$$

The last equation in the system (14) follows by direct calculations.

The optimality system associated with the VPT control problem (8) consists of the system (5), the system (14), and

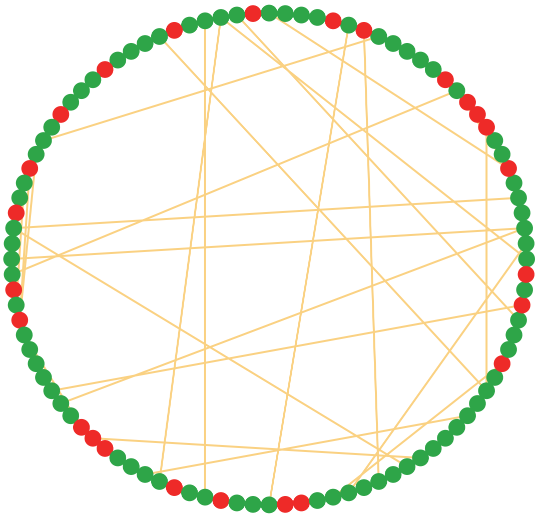


FIGURE 2: A synthetic small-world network G_{SW}, where U_{SW} consists of the red nodes.

$\mathbf{p}(T) = \mathbf{0}$. In practice, we may apply the well-known Forward-Backward Euler Scheme [38] to solve the optimality system. □

5. Examples of Best-Balanced Patch-Injecting Strategy

In this section, we present a few best-balanced patch-injecting strategies by solving the corresponding instances of the VPT control problem. For comparative purpose, for the VPT control problem (8) and the admissible control γ_I, we define the cumulative balance function as

$$J(\gamma_I, t) = \int_0^t c\gamma_I(s)\,ds + \int_0^t \sum_{i=1}^{N} w_i I_i(s)\,ds, \quad 0 \le t \le T. \quad (17)$$

Obviously, $J(\gamma_I, T) = J(\gamma_I)$. For convenience, let $\mathbf{1}$ denote an all-one row vector with appropriate number of dimensions.

Small-world networks are networks with small diameter and high clustering coefficient [39]. By invoking Pajek [40], the well-known social network analysis software, we get a synthetic small-world network G_{SW}, which is plotted in Figure 2, where the patch injection subset U_{SW} consists of the red nodes.

Example 1. Consider the following instance of the VPT control problem:

$$(G_{SW} \mid U_{SW}, 0.1, 0.4, 0.4, 0.1, 0.2, 0.8, 1, \mathbf{1}, 20, 0.1 \times \mathbf{1}). \quad (18)$$

By solving the corresponding optimality system, we get an optimal control, which is depicted in Figure 3(a). Figure 3(b) plots the cumulative balance functions for the optimal control and three static controls, from which it is seen that the optimal control is superior to these static controls in terms of the balance.

Scale-free networks are networks with power-law degree distribution [41]. Again by invoking Pajek, we get a synthetic scale-free network G_{SF}, which is portrayed in Figure 4, where the patch injection subset U_{SF} consists of the red nodes.

Example 2. Consider the following instance of the VPT control problem:

$$(G_{SF} \mid U_{SF}, 0.1, 0.4, 0.4, 0.1, 0.2, 0.8, 1, \mathbf{1}, 20, 0.1 \times \mathbf{1}). \quad (19)$$

By solving the corresponding optimality system, we get an optimal control, which is exhibited in Figure 5(a). Figure 5(b) plots the cumulative balance functions for the optimal control and three static controls, from which it is seen that the optimal control outperforms these static controls in terms of the balance.

Figure 6 exhibits a real-world email network G_{EM}, which comes from [42]. Here, the patch injection subset U_{EM} consists of the red nodes.

Example 3. Consider the following instance of the VPT control problem:

$$(G_{EM} \mid U_{EM}, 0.1, 0.4, 0.4, 0.1, 0.2, 0.8, 1, \mathbf{1}, 20, 0.1 \times \mathbf{1}). \quad (20)$$

By solving the corresponding optimality system, we get an optimal control, which is shown in Figure 7(a). Figure 7(b) plots the cumulative balance functions for the optimal control and three static controls, from which it is seen that the optimal control overmatches these static controls in terms of the balance.

We conclude from the above examples that a best-balanced patch-injecting strategy first stays at the maximum allowable patch injection rate, then sharply jumps to the minimum allowable patch injection rate, and finally stays at this rate.

6. Further Discussions

In this section, we examine the effects of some factors on the best balance of a patch-injecting strategy. For convenience, let γ_I^{opt} denote a best-balanced patch-injecting strategy, $J^{opt} = J(\gamma_I^{opt})$ the corresponding balance.

6.1. The Effects of the Four Rates.

First, we inspect the effect of the four rates, β_I, β_P, γ_P, and δ, on the best balance.

Experiment 6. Consider the following instances of the VPT control problem:

$$(G \mid U, \beta_I, 0.4, 0.4, 0.1, 0.2, 0.8, 1, \mathbf{1}, 20, 0.1 \times \mathbf{1}), \quad (21)$$

where $G \mid U \in \{G_{SW} \mid U_{SW}, G_{SF} \mid U_{SF}, G_{EM} \mid U_{EM}\}$, $\beta_I \in \{0.1, 0.2, \ldots, 0.9\}$. Figure 8 exhibits the best balances of these instances.

It is concluded from this experiment that J^{opt} is increasing with β_I. As a result, the best balance can be improved by persuading the network users not to install suspicious software.

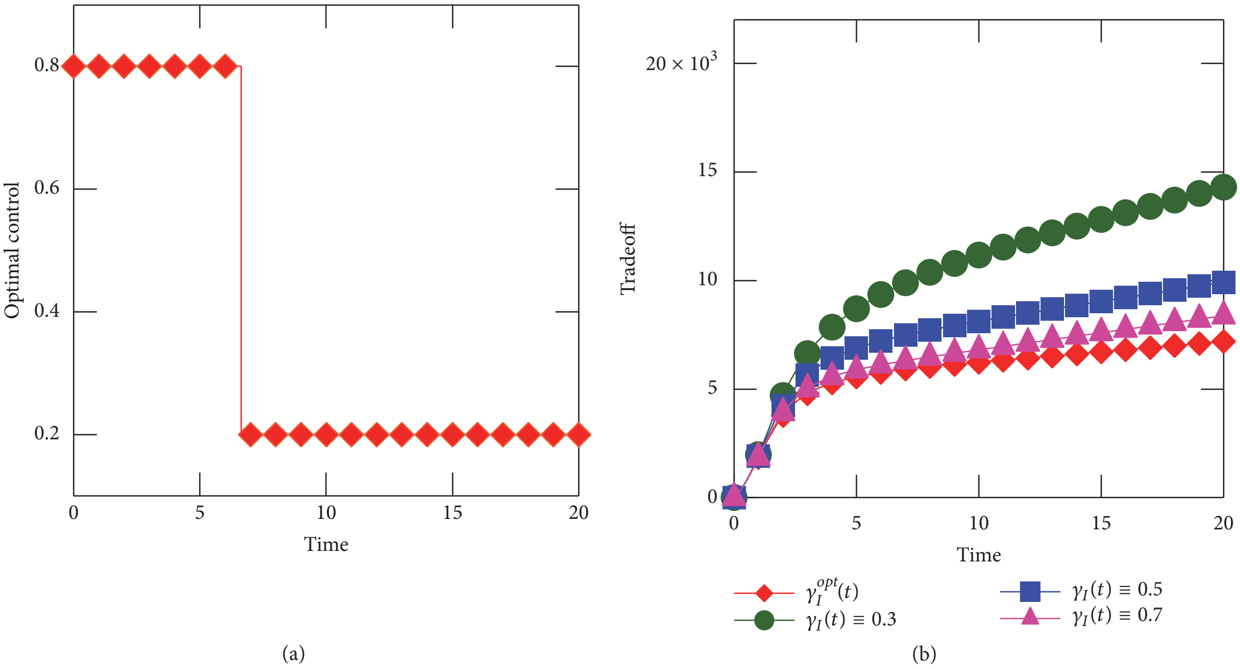


FIGURE 3: Results in Example 1: (a) an optimal control; (b) a comparison between the optimal control and three static controls.

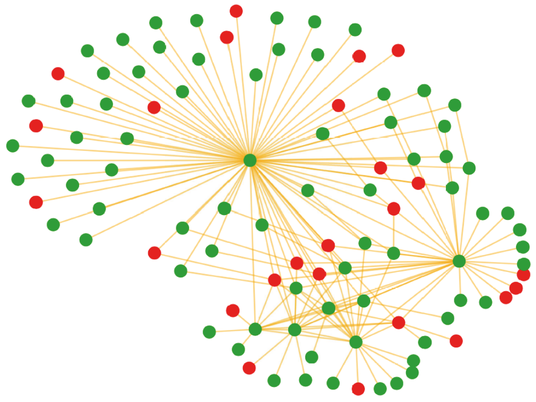


FIGURE 4: A synthetic scale-free network G_{SF}, where U_{SF} consists of the red nodes.

Experiment 7. Consider the following instances of the VPT control problem:

$$(G \mid U, 0.1, \beta_P, 0.4, 0.1, 0.2, 0.8, 1, \mathbf{1}, 20, 0.1 \times \mathbf{1}), \qquad (22)$$

where $G \mid U \in \{G_{SW} \mid U_{SW}, G_{SF} \mid U_{SF}, G_{EM} \mid U_{EM}\}$, $\beta_P \in \{0.1, 0.2, \ldots, 0.9\}$. Figure 9 displays the best balances of these instances.

It is concluded from this experiment that J^{opt} is increasing with β_P. Again, this conclusion demonstrates that warning the network users not to install suspicious software would improve the best balance.

Experiment 8. Consider the following instances of the VPT control problem:

$$(G \mid U, 0.1, 0.4, \gamma_P, 0.1, 0.2, 0.8, 1, \mathbf{1}, 20, 0.1 \times \mathbf{1}), \qquad (23)$$

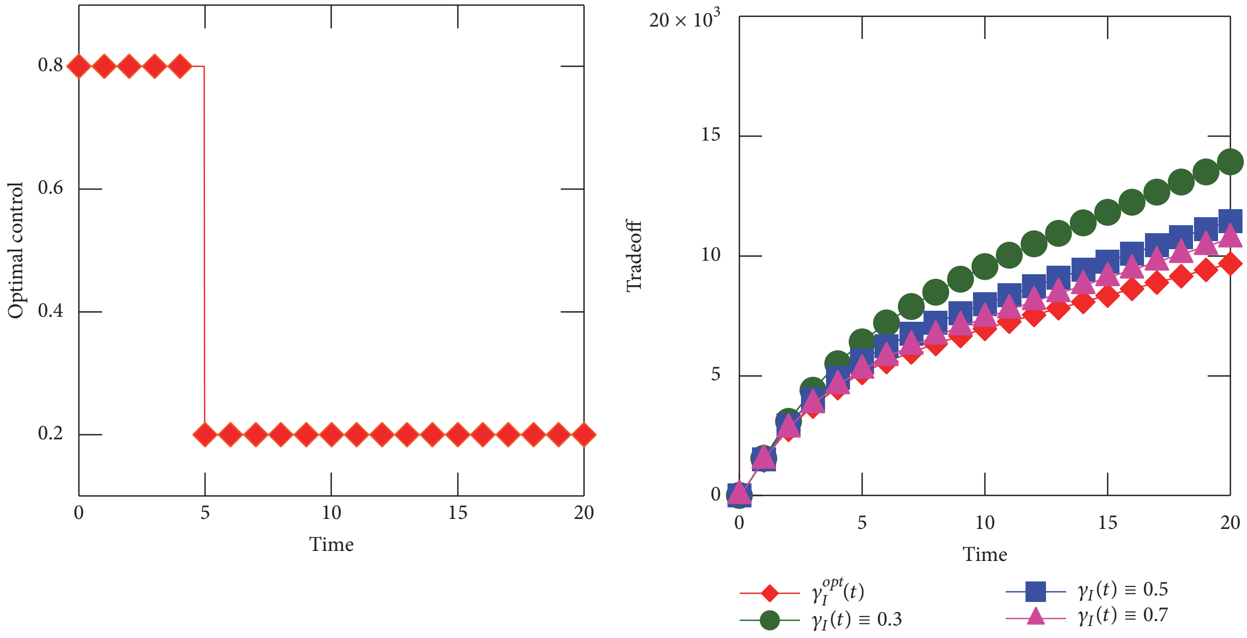
where $G \mid U \in \{G_{SW} \mid U_{SW}, G_{SF} \mid U_{SF}, G_{EM} \mid U_{EM}\}$, $\gamma_P \in \{0.1, 0.2, \ldots, 0.9\}$. Figure 10 exhibits the best balances of these instances.

It is concluded from this experiment that J^{opt} is decreasing with γ_P. Therefore, the best balance can be improved by reminding the network users of timely installing new patches.

Experiment 9. Consider the following instances of the VPT control problem:

$$(G \mid U, 0.1, 0.4, 0.4, \delta, 0.2, 0.8, 1, \mathbf{1}, 20, 0.1 \times \mathbf{1}), \qquad (24)$$

where $G \mid U \in \{G_{SW} \mid U_{SW}, G_{SF} \mid U_{SF}, G_{EM} \mid U_{EM}\}$, $\delta \in \{0.1, 0.2, \ldots, 0.9\}$. Figure 11 exhibits the best balances of these instances.



(a)

(b)

FIGURE 5: Results in Example 2: (a) an optimal control; (b) a comparison between the optimal control and three static controls.

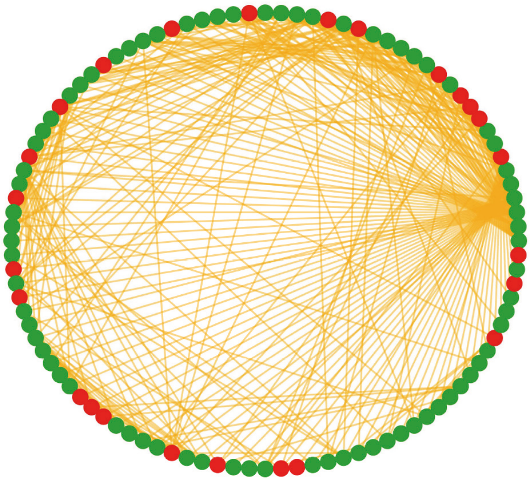


FIGURE 6: An email network G_{EM}, where U_{EM} consists of the red nodes.

It is concluded from this experiment that J^{opt} is increasing with δ. It follows that the best balance can be improved by developing patches that can defend against future viruses.

6.2. The Effects of the Two Bounds. Second, let us investigate the effects of the minimum allowable patch injection rate and the maximum allowable patch injection rate on the best balance.

Experiment 10. Consider the following instances of the VPT control problem:

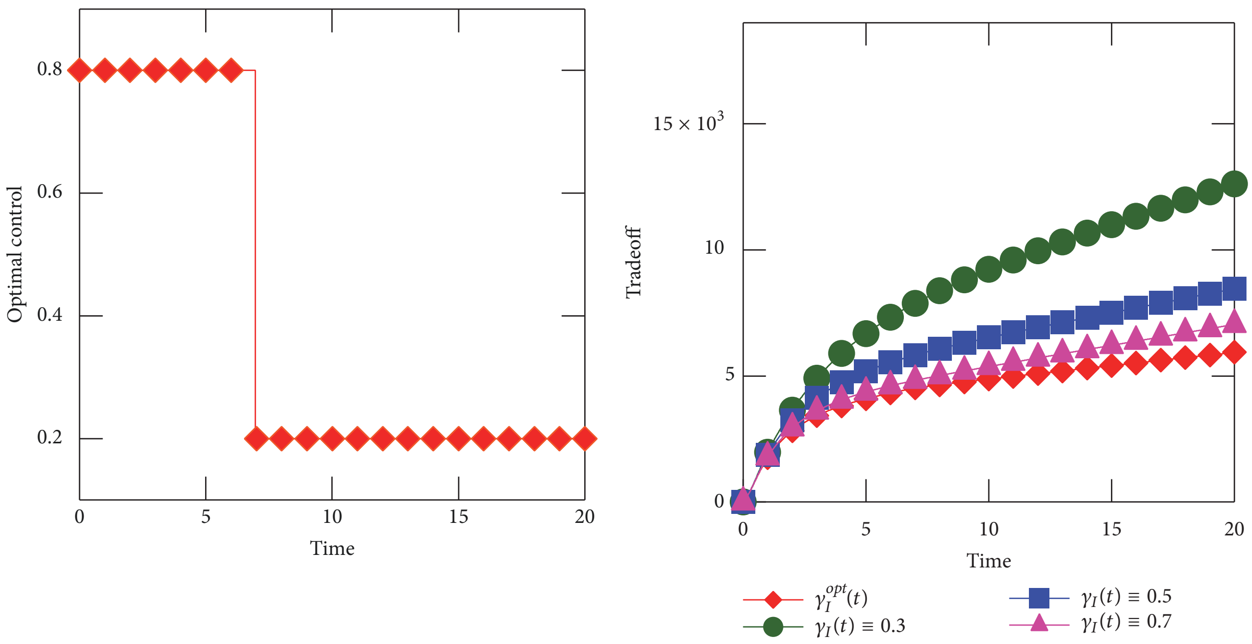
$$\left(G \mid U, \beta_I, 0.1, 0.4, 0.4, 0.1, \underline{\gamma_I}, \overline{\gamma_I}, 1, \mathbf{1}, 20, 0.1 \times \mathbf{1}\right), \qquad (25)$$

where $G \mid U \in \{G_{SW} \mid U_{SW}, G_{SF} \mid U_{SF}, G_{EM} \mid U_{EM}\}$, $\underline{\gamma_I}, \overline{\gamma_I} \in \{0.1, 0.2, \dots, 0.9\}$. Figure 12 exhibits the best balances of these instances.

It is concluded from this experiment that J^{opt} is increasing with $\underline{\gamma_I}$ and is decreasing with $\underline{\gamma_I}$. In practice, we should reduce the lowest allowable patch injection rate and enhance the highest allowable patch injection rate to achieve a better balance.

7. Concluding Remarks

Virus patches play an important role in restraining computer viruses. This paper has addressed the problem of seeking patch-injecting strategies that achieve the best balance



(a)

(b)

FIGURE 7: Results in Example 3: (a) an optimal control; (b) a comparison between the optimal control and three static controls.

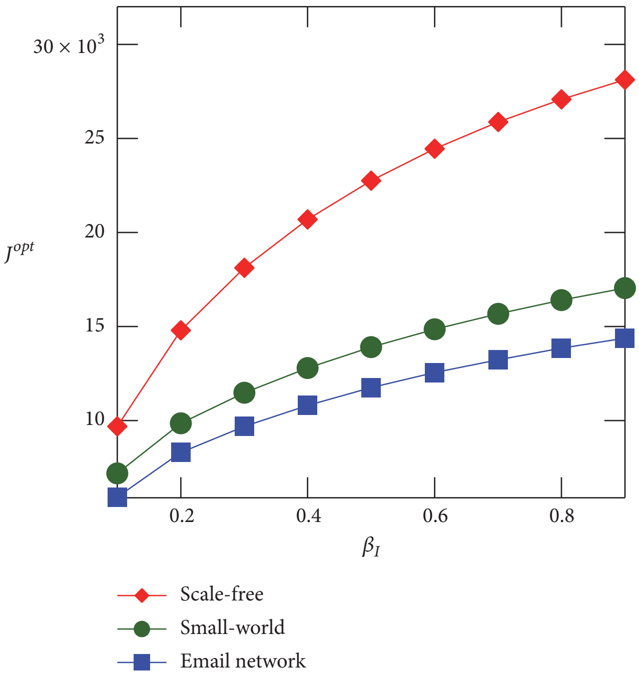


FIGURE 8: The best balances in Experiment 6.

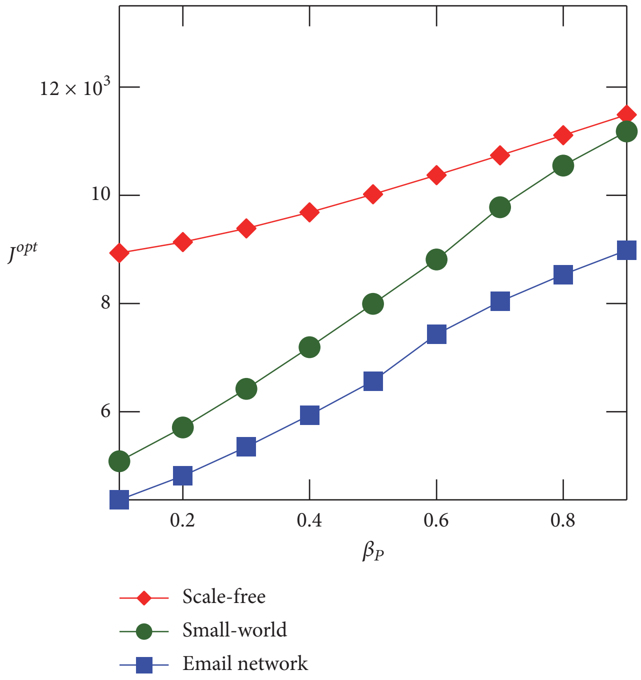


FIGURE 9: The best balances in Experiment 7.

between the patch-developing cost and the impact of virus. The problem has been reduced to an optimal control problem, and a scheme for solving the optimal control problem has been presented. Finally, the effects of some factors on the best balance of a patch-injecting strategy have been examined.

Some relevant problems are yet to be resolved. First, the problem of how to select a given number of patch injection nodes so that the balance is optimized is worth study. Second, in this article it is assumed that the patch propagation rate is fixed. In practice, the network defender may change this rate flexibly through rewriting the communication protocol.

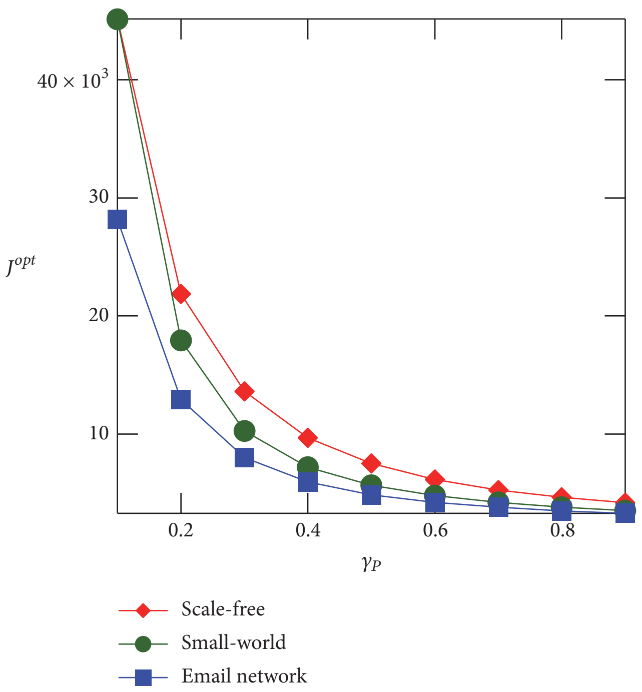


Figure 10: The best balances in Experiment 8.

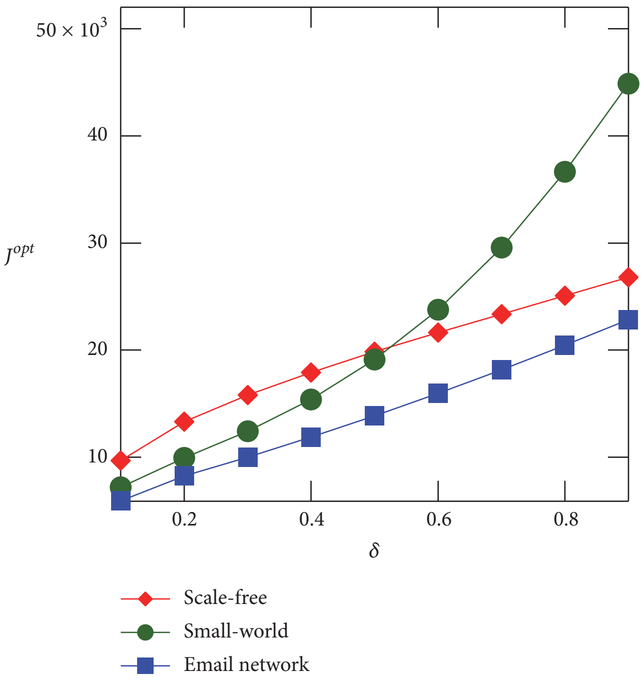
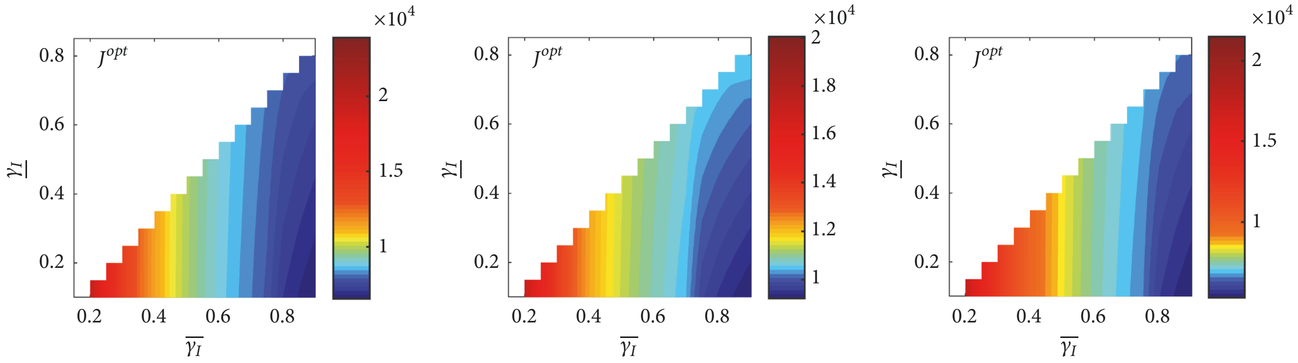


Figure 11: The best balances in Experiment 9.

In this situation, we will face a new and more complex balance problem. Next, in this article the virus injection rate is assumed to be fixed. In reality, the virus maker may flexibly change the rate to avoid detection. In this context, it is appropriate to deal with the balance problem through game-theoretic approach [43–46]. Finally, the research approach used in this article may be applied to some other areas such as cloud security [47, 48] and Internet of Things security [49].

Data Availability

The data used to support the findings of this study are included within the article.



(a) $G_{SW} \mid U_{SW}$

(b) $G_{SF} \mid U_{SF}$

(c) $G_{EM} \mid U_{EM}$

Figure 12: The best balances in Experiment 10.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was funded by National Natural Science Foundation of China (Grant No. 61572006), Chongqing Basic Research and Front Exploration Project (Grant No. cstc2018jcyjA3093), and Australian Research Council (Grant No. LP170100458).

References

[1] T. J. OLeary and L. I. OLeary, "Computing essentials 2014 complete edition, making it work for you," in *Computing Essentials 2014 Complete Edition, Making it Work for You, McGraw-Hill Education*, 2014.

[2] D. Comer, *Computer Networks and Internets*, Pearson, 6th edition, 2014.

[3] S. Mohurle and M. Patil, "A brief study of wannacry threat: ransomware attack 2017," *International Journal of Advanced Research in Computer Science*, vol. 8, no. 5, pp. 1938–1940, 2017.

[4] P. Szor, *The Art of Computer Virus Research and Defense*, Addison-Wesley Education, 2005.

[5] J. Freund and J. Jones, *Measuring and Managing Information Risk: A Fair Approach*, Butterworth-Heinemann, 1st edition, 2014.

[6] D. W. Hubbard and R. Seiersen, *How to Measure Anything in Cybersecurity Risk*, John Wiley & Sons, Inc., Hoboken, NJ, USA, 1st edition, 2016.

[7] L. C. Chen and K. M. Carley, "The impact of countermeasure propagation on the prevalence of computer viruses," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 34, no. 2, pp. 823–833, 2004.

[8] J. Goldenberg, Y. Shavitt, E. Shir, and S. Solomon, "Distributive immunization of networks against viruses using the "honey-pot" architecture," *Nature Physics*, vol. 1, no. 3, pp. 184–188, 2005.

[9] N. F. Britton, *Essential Mathematical Biology*, Springer Undergraduate Mathematics Series, Springer, 2003.

[10] L.-X. Yang and X. Yang, "The effect of infected external computers on the spread of viruses: a compartment modeling study," *Physica A: Statistical Mechanics and its Applications*, vol. 392, no. 24, pp. 6523–6535, 2013.

[11] A. K. Misra, M. Verma, and A. Sharma, "Capturing the interplay between malware and anti-malware in a computer network," *Applied Mathematics and Computation*, vol. 229, pp. 340–349, 2014.

[12] L.-X. Yang and X. Yang, "A novel virus-patch dynamic model," *PLoS ONE*, vol. 10, no. 9, Article ID e0137858, 2015.

[13] B. Liu and C. Li, "A new virus-antivirus spreading model," in *Proceedings of International Symposium on Neural Networks (ISNN2015)*, pp. 481–488, Springer, 2015.

[14] D.-W. Huang, L.-X. Yang, X. Yang, Y. Wu, and Y. Y. Tang, "Towards understanding the effectiveness of patch injection," *Physica A: Statistical Mechanics and its Applications*, vol. 526, p. 120956, 2019.

[15] P. Van Mieghem, J. Omic, and R. Kooij, "Virus spread in networks," *IEEE/ACM Transactions on Networking*, vol. 17, no. 1, pp. 1–14, 2009.

[16] P. Van Mieghem, "The N-intertwined SIS epidemic network model," *Computing*, vol. 93, no. 2, pp. 147–169, 2011.

[17] I. Stojmenovic, *Handbook of Wireless Networks and Mobile Computing*, John Wiley & Sons, 2002.

[18] A. Boukerche, *Handbook of Algorithms for Wireless Networking and Mobile Computing*, Taylor & Francis, 2006.

[19] J. Rodriguez, *Fundamentals of 5G Mobile Networks*, John Wiley & Sons, 2015.

[20] S. Xu, W. Lu, and Z. Zhan, "A stochastic model of multivirus dynamics," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 1, pp. 30–45, 2012.

[21] S. Xu, W. Lu, and L. Xu, "Push- and pull-based epidemic spreading in networks: thresholds and deeper insights," *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol. 7, no. 32, 2012.

[22] S. Xu, W. Lu, L. Xu, and Z. Zhan, "Adaptive epidemic dynamics in networks: Thresholds and control," *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol. 8, no. 4, article no. 19, 2014.

[23] L.-X. Yang, M. Draief, and X. Yang, "The impact of the network topology on the viral prevalence: a node-based approach," *PLoS ONE*, vol. 10, no. 7, article e0134507, 2015.

[24] L. X. Yang, M. Draief, and X. Yang, "Heterogeneous virus propagation in networks: a theoretical study," *Mathematical Methods in the Applied Sciences*, vol. 40, pp. 1396–1413, 2016.

[25] L.-X. Yang, X. Yang, and Y. Yan Tang, "A bi-virus competing spreading model with generic infection rates," *IEEE Transactions on Network Science and Engineering*, vol. 5, no. 1, pp. 2–13, 2017.

[26] L.-X. Yang, X. Yang, and Y. Wu, "The impact of patch forwarding on the prevalence of computer virus: a theoretical assessment approach," *Applied Mathematical Modelling*, vol. 43, pp. 110–125, 2017.

[27] L. Yang, P. Li, X. Yang, Y. Xiang, and W. Zhou, "A differential game approach to patch injection," *IEEE Access*, vol. 6, pp. 58924–58938, 2018.

[28] J. Balthrop, S. Forrest, M. E. J. Newman, and M. M. Williamson, "Technological networks and the spread of computer viruses," *Computer Science*, vol. 304, no. 5670, pp. 527–529, 2004.

[29] E. K. Donald, *Optimal Control Theory: An Introduction*, Dover Publications, 2012.

[30] D. Liberzon, *Calculus of Variations and Optimal Control Theory: A Concise Introduction*, Princeton University Press, Princeton, NJ, USA, 2012.

[31] L. Chen, K. Hattaf, and J. T. Sun, "Optimal control of a delayed SLBS computer virus model," *Physica A: Statistical Mechanics and its Applications*, vol. 427, pp. 244–250, 2015.

[32] L.-X. Yang, M. Draief, and X. Yang, "The optimal dynamic immunization under a controlled heterogeneous node-based SIRS model," *Physica A: Statistical Mechanics and its Applications*, vol. 450, pp. 403–415, 2016.

[33] Y. Pei, H. Pei, X. Liang, and M. Zhu, "Optimal control of a computer virus model with network attacks," *Communications in Mathematical Biology and Neuroscience*, vol. 2016, Article ID 17, 2016.

[34] T. Zhang, L.-X. Yang, X. Yang, Y. Wu, and Y. Y. Tang, "Dynamic malware containment under an epidemic model with alert," *Physica A: Statistical Mechanics and its Applications*, vol. 470, pp. 249–260, 2017.

[35] J. Bi, X. Yang, Y. Wu, Q. Xiong, J. Wen, and Y. Y. Tang, "On the optimal dynamic control strategy of disruptive computer virus," *Discrete Dynamics in Nature and Society*, vol. 2017, Article ID 8390784, 2017.

[36] E. M. Stein and R. Shakarchi, *Real Analysis: Measure Theory, Integration, Hilbert Spaces*, Princeton University Press, Princeton, NJ, Texas, 2005.

[37] R. C. Robinson, *An Introduction to Dynamical Systems: Continuous and Discrete*, Prentice Hall, New York, NY, USA, 2004.

[38] K. E. Atkinson, W. Han, and D. Stewart, *Numerical Solution of Ordinary Differential Equations*, John Wiley & Sons, 2009.

[39] D. J. Watts and S. H. Strogatz, "Collective dynamics of "small-world" networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[40] W. de Nooy, A. Mrvar, and V. Batagelj, *Exploratory Social Network Analysis with Pajek*, Cambridge University Press, Cambridge, UK, 2005.

[41] A. Barabasi and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[42] http://konect.uni-koblenz.de/networks/arenas-email.

[43] M. J. Osborne, *An Introduction to Game Theory*, Oxford University Press, 2003.

[44] T. Alpcan and T. Başar, *Network Security: A Decision and Game-Theoretic Approach*, Cambridge University Press, Cambridge, UK, 2010.

[45] L. Yang, P. Li, Y. Zhang, X. Yang, Y. Xiang, and W. Zhou, "Effective repair strategy against advanced persistent threat: a differential game approach," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 7, pp. 1713–1728, 2019.

[46] L. Yang, P. Li, X. Yang, and Y. Tang, "A risk management approach to defending against the advanced persistent threat," *IEEE Transactions on Dependable and Secure Computing*, pp. 1-1, 2018.

[47] M. A. Khan, "A survey of security issues for cloud computing," *Journal of Network and Computer Applications*, vol. 71, pp. 11–29, 2016.

[48] A. Singh and K. Chatterjee, "Cloud security issues and challenges: A survey," *Journal of Network and Computer Applications*, vol. 79, pp. 88–115, 2017.

[49] F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, "Internet of things security: a survey," *Journal of Network and Computer Applications*, vol. 88, pp. 10–28, 2017.