Computer Systems
Science and Engineering

Tech Science Press

DOI: 10.32604/csse.2023.038917
Article

Social Engineering Attack-Defense Strategies Based on Reinforcement Learning

Rundong Yang[1,*], Kangfeng Zheng[1], Xiujuan Wang[2], Bin Wu[1] and Chunhua Wu[1]

[1]School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing, 100876, China
[2]School of Computer Science, Beijing University of Technology, Beijing, 100124, China
*Corresponding Author: Rundong Yang. Email: rundyang@bupt.edu.cn
Received: 04 January 2023; Accepted: 20 April 2023; Published: 28 July 2023

Abstract: Social engineering attacks are considered one of the most hazardous cyberattacks in cybersecurity, as human vulnerabilities are often the weakest link in the entire network. Such vulnerabilities are becoming increasingly susceptible to network security risks. Addressing the social engineering attack defense problem has been the focus of many studies. However, two main challenges hinder its successful resolution. Firstly, the vulnerabilities in social engineering attacks are unique due to multistage attacks, leading to incorrect social engineering defense strategies. Secondly, social engineering attacks are real-time, and the defense strategy algorithms based on gaming or reinforcement learning are too complex to make rapid decisions. This paper proposes a multiattribute quantitative incentive method based on human vulnerability and an improved Q-learning (IQL) reinforcement learning method on human vulnerability attributes. The proposed algorithm aims to address the two main challenges in social engineering attack defense by using a multiattribute incentive method based on human vulnerability to determine the optimal defense strategy. Furthermore, the IQL reinforcement learning method facilitates rapid decision-making during real-time attacks. The experimental results demonstrate that the proposed algorithm outperforms the traditional Q-learning (QL) and deep Q-network (DQN) approaches in terms of time efficiency, taking 9.1% and 19.4% less time, respectively. Moreover, the proposed algorithm effectively addresses the non-uniformity of vulnerabilities in social engineering attacks and provides a reliable defense strategy based on human vulnerability attributes. This study contributes to advancing social engineering attack defense by introducing an effective and efficient method for addressing the vulnerabilities of human factors in the cybersecurity domain.

Keywords: Social engineering; game theory; reinforcement learning; Q-learning

1 Introduction

With the continuous development of network technology, communication is not limited by traditional distance or the various social networks, e-mail, or network communication methods that satisfy daily needs for communication and entertainment. The internet is becoming increasingly important, and we cannot live without it. However, there are also nefarious actors lurking in the network; they attack by taking advantage of users' psychological weaknesses and inducing them to disclose sensitive information [1].

In the second quarter of 2022, the APWG (Anti-Phishing Working Group) observed 1,097,811 phishing attacks, a new record, and this was the worst quarter for phishing ever followed by the APWG. The number of phishing attacks reported to the APWG has quadrupled since the beginning of 2020, when the APWG started to keep phishing attacks. A total of 68,000 to 94,000 episodes per month were followed by the APWG in early 2020 [2].

Unlike traditional cyber attacks, social engineering attacks mainly exploit the psychological weaknesses of the target to execute the attack, and Reference [3] designed a general architecture for social engineering attacks. The main features of the structure are attack preparation, attack implementation, and attack gain. In the preparation stage, information and relationships are collected for the target, usually using web crawlers, social network information collection, etc. A social engineering script design is carried out in the attack preparation stage for the social engineering targets. In the attack gain stage, all information obtained is evaluated and judged on whether the attack was successful [4].

Network security has recently received increased attention, especially for social engineering attack defense strategies research. Many studies have focused on reinforcement learning, some based on game theory [4–7]. However, there are two problems with these technical approaches. First, social engineering attacks exploit human vulnerabilities to deceive and trick users into revealing sensitive information. The traditional social engineering defense strategy does not consider user vulnerability, leading to ineffective defense strategies. Therefore, we need to design a new model considering user vulnerability. Second, social engineering attacks are real-time, requiring real-time reactions to the attacks to avoid serious harm. Current game theory and reinforcement learning-based defense strategy responses are delayed because of their high time complexity [8,9].

A summary of the related work on reinforcement learning applied to cybersecurity, game-learning programs, and secure game-theoretic modeling are shown in Table 1.

Table 1: Summary of the related work

Category	Reference	Algorithms	Main contributions
Reinforcement learning in network security	Zhong et al. [10]	DNN, SVM,RL	An RL-based system is proposed to protect users from malicious traffic. Generate agents through network attack and defense based on the deep neural network environment, surpass the traditional ML algorithm, and can detect adversarial samples.

(Continued)

Table 1 (continued)

Category	Reference	Algorithms	Main contributions
	Elderman et al. [11]	MMQL, NQL	A method for modeling the decision-making process of network security monitoring using a game-theoretic approach.
	Chung et al. [12]	MDP, optimal attacker policy	A solution to attack graph transformation is proposed. Transform attack graphs into MDPs and use policy search to address defense policy generation.
	Durkota et al. [13]	DRL, RL	Integrate traditional reinforcement learning into deep learning, and use deep reinforcement learning to build an autonomous network defense system to control and protect network security.
Game-learning programs	Durkota et al. [14]	MDP, RF	Consequences of using the Markov game framework instead of MDPs in reinforcement learning. Solve the optimal strategy of a two-person zero-sum game.
	Ridley [15]	nash Q-Learning	Based on the framework of random games, Q-learning is extended to multi-agent systems. Nash Q-learning (NashQ) is proposed, which uses multi-agent Q to learn the best defense strategy under the random game framework.
	Littman [16]	Neural fictitious self-play, NFSP	Introduced Neural Fictitious Self-Play (NFSP), the first end-to-end deep reinforcement learning method for learning an approximate Nash equilibrium for imperfect information games from self-play NFSP requires no prior domain knowledge can be expanded.
Game theoretic modeling in cyber security	Hu et al. [17]	FLIPIT	Defines the FLIPIT game and the application of FLIPIT in various computer security scenarios (including APT).

(Continued)

Table 1 (continued)

Category	Reference	Algorithms	Main contributions
	Heinrich et al. [18]	MPC, DRL, RL, game theory	Existing security games in computer networks are reviewed and compared in terms of players, games, etc., with the overall goal of identifying and addressing security and privacy issues, where game theory can be applied to model and evaluate security issues and be used to design effective protocols.

In this paper, a new social engineering defense model is designed by combining the essential attributes of users to provide an optimal defense strategy with a low-computational-complexity social engineering defense. In addition, this paper presents a mechanism for quantifying user characteristics to model the vulnerability of users for the first time quantitatively, and a stochastic game is used to simulate the interaction between attackers and defenders. Finally, this paper applies Q-learning to stochastic games, constructs a reinforcement learning model for multiple intelligences, proposes a Q-learning algorithm based on user attributes, and optimizes the algorithm. Multiple attackers are treated as independent intelligence that can learn actively and independently to collect more information for the system proposes a proposed Q-learning (IQL) algorithm to reduce the algorithm's complexity algorithm and improve its efficiency. The main research contributions of this paper are as follows.

This paper proposes a mechanism for quantifying user vulnerability based on target attributes that consider the interaction between user vulnerability and attackers and design a more comprehensive social engineering model approach to improve social engineering security.

This paper considers attackers and defenders as two sides of a game and designs a multi-intelligence reinforcement learning model using stochastic game theory combined with Q-learning. For the first time, this paper proposes a multiobjective attribute structure learning algorithm that can provide optimal decision strategies.

This paper proposes an optimization algorithm IQL. This paper can quickly obtain an optimal defense strategy by combining target attributes and user vulnerability information strategy. It is experimentally demonstrated that the algorithm performs better than QL and DQN.

This paper is composed of five sections. Following this introduction is Chapter 2, Problem Definition. In Chapter 3, Presenting the Model, an improved QL algorithm is proposed. This is followed by Chapter 4, Experimental Results and Analysis. Chapter 5 concludes the paper.

2 Definition of the Problem

Usually, when attackers engineer user attacks, this paper considers the attack method, attack technique, attack detection, etc., however, all of these factors must be identified through human judgment. Therefore, the threat comes from combining these attacks and interaction with people during attack reinforcement. Existing social engineering attack defense models ignore the role of human attributes, so this paper proposes a new quantitative approach that combines human characteristics to quantitatively evaluate each attack node using the standard notation in Table 2.

Table 2: Frequently used symbols

Notation	Definition
w	A constant
S	State space of the game model
X	Players of the game model
A	Attack actions in the game model
D	Defensive actions in the game model
R_A	Attacker's reward in the game model
R_B	Defender's reward in the game model
a^*	The optimal attack strategy
d^*	The optimal defensive strategy
δ	The probability of a successful attack
$U(a, d)$	The utility function of the game model
$R(s, a, d)$	The immediate reward
$E(U')$	The expected utility in the next state
$Q(s, a, d)$	The Q function of the Q-learning algorithm
α	Learning rate
γ	Discount factor

The attributes of the nodes are divided into two types: physical attributes and target attributes. A physical analysis mainly considers the impact size of the nodes in the entire system. Each node's physical characteristics include the importance level of the node and the connection level in the node. The node's target attributes mainly have the features of the target, security knowledge, character, and security awareness attributes. These attributes are directly related to the strength of the security defense.

Definition 1: The physical attributes mainly include the importance level and connection level of the node.

The importance level (IL) mainly indicates the node's importance in the entire social engineering system. This importance level primarily three factors: the valid information that can be obtained, the impact on subsequent attacks, and whether trust is established [19].

The connection level (CL), which indicates the importance of the node's associations in the social engineering system, is determined by the stage of the social engineering model in which the node is located and the number of other nodes connected to this node.

Definition 2: Target attributes (TA) mainly include defense technology, defense means, basic information of the target, security knowledge level, personality type, security awareness registration, and target cognitive paths [20]. The higher the target's attributes value, the stronger its defense capability and the higher the node security. If the physical characteristics of the node are high, the assigned defense is enhanced accordingly [21].

The above values of the physical attributes and target attributes are set by the system administrator and mapped to a vector F, where f_i denotes the result of the administrator's evaluation of i using F. The ratings for the social engineering security system are divided into three primary levels: high, medium, and low. Therefore, we have F $\in [1, 2, 3]$. The attribute-based social engineering security metric (SESM) is defined as

$$\text{SESM} = w * \log_2 \left(1 + \frac{IL \cdot CL}{TA} \right) \tag{1}$$

In the above equation, w is a constant. This constant is related to the state space and action space for the node: $w = \frac{A + D}{S}$, where the greater the value of the SESM is, the higher the importance of the node, the greater the loss due to an attack, and the lower the node defense capability. The value of SESM indicates the target receiving the social engineering attacks and the impact of each node on itself. For the social engineering attacker, who mainly uses the vulnerability of the person at the node to implement the attacks, this paper defines a time-based function in the form of attack resource consumption, defense resource consumption, and loss recovery consumption.

Attack resource consumption (AR): the consumption of attack resources in the attack preparation phase, the attack implementation phase, target information collection, scripting, trust building, and other actions that consume time [22].

Defense resource (DR) consumption: resource consumption in resisting social engineering attacks; time consumed in preventing attackers from obtaining protected information, detecting attacks, and identifying attacks for information collection [23].

Loss recovery consumption (LR): the time consumed in recovering from the loss caused by the attack, such as replacing a secret key, changing a password, or taking other actions to protect one's property and information [24].

3 Material and Methods

In the human vulnerability-based social engineering model, an attacker can use human vulnerabilities to perform social engineering attacks and obtain sensitive information. For the attacker, the greater the vulnerability found, the greater the harm and the greater the social engineering gain. Defense against social engineering focuses on the corresponding social engineering defenses for detected social engineering attacks. Ideally, the target has no exploitable vulnerabilities and is safe. However, in practice, the target attributes vary, the vulnerability performance ranges, the attacker can always find vulnerabilities to attack, and the defenses can lag and collapse when an attack occurs. The attacker and the defender are similar to the two sides of a game; the attacker tries to obtain the maximum reward, and the defender pursues the minimum loss. The two sides of the attacker and defender can be considered a stochastic game, and a stochastic game model can be used to analyze the best defense strategy for the defender. This paper design a new reward quantification method to quantify the role of vulnerability in the model, considering the properties of the target and the interaction between the vulnerability and the attack. The social engineering attack model is shown in Fig. 1 below.

3.1 Basic Model

The structure of the social engineering system model is complex, with multiple stages. At each stage, the attacker does not have access to information about the entire system and takes random actions based on the information obtained at this stage. At the same time, neither the attacker nor the defender has access to the game information of the adversary or the gain of each action. This paper describes the game between the attacker and the defender in the social engineering system as a stochastic game model with incomplete information. The model is defined as follows.

$$G = \langle S, X, A, D, R_A, R_D \rangle \tag{2}$$

where S denotes the state set, X denotes the players involved in the game, A drepresents the attack action space, D drepresents the defense action space, R_A denotes the attack gain, and R_D denotes the defense gain. The players in the different states have corresponding sets of response actions; attackers have attack action sets, and defenders have defense action sets. The attackers and defenders have action sets that implement offensive and defensive games. The current state, attack action, and defense action result in a state change, and the attacker and the defender obtain live gains.

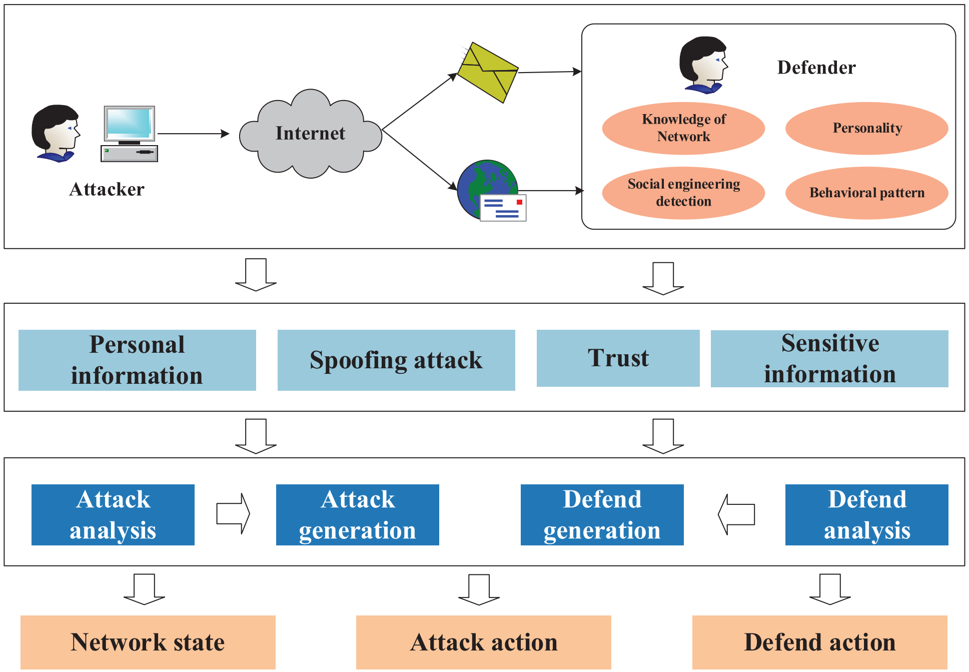

Figure 1: Social engineering attack model diagram

The attacker's utility consists of the attack gain and the period of the next state, and it can be expressed as

$$U_A(\mathbf{a}, \mathbf{d}) = R(s, a, d) + E\left(U_A'\right) \tag{3}$$

$E\left(U_A'\right)$ in the above equation represents the expectation of the next state, where $R(s, a, d)$ represents the attacker's attack gain, a represents the attack action and d represents the defense action. Then, the dynamic policy is defined as

$$(\mathbf{a}, \mathbf{d}) = \begin{bmatrix} a_1, d_1, & a_1, d_2, & \ldots, & a_1, d_I \\ a_2, d_1, & a_2, d_2, & \ldots, & a_2, d_I \\ \ldots & \ldots & \ddots & \ldots \\ a_J, d_1, & a_J, d_2, & \ldots, & a_J, d_I \end{bmatrix} \tag{4}$$

In the above equation, $\left(a_j, d_i\right)$ denotes an action pair with attack action a_j and defense action d_i in state s. In the game between the attacker and defender, the utility of the attacker is not only determined by the attack action but also depends on the defense action; then, the reward function is

$$R\left(s, a, d\right) = \mathrm{SESM} \cdot \left(\delta \cdot RT\left(a\right) + DT\left(d\right) - AT\left(a\right)\right) \tag{5}$$

Here, the SESM is an attribute-based social engineering security level indicator, and the SESM mainly reflects the node's importance. δ indicates the success probability of the social engineering attack, and we analyze this probability by studying real social engineering attack cases. RT is the set of resources required to recover from social engineering attack action a. The response of the target is affected by the degree of the attack, where DT indicates the defense resource consumption of the attack action. In addition to the physical resource consumption, the target's mental resource consumption must be considered. The effectiveness function U represents the interaction between the target vulnerability and the social engineering attack. The purpose of the defense is to reduce the loss from the social engineering attack, so we can assume that the attack gain of the optimal attacker is 0. The game between the attacker and the defender in the social engineering process is zero-sum [25].

By formalizing the representations of the social engineering attack participants, the entire defined social engineering attack process is constructed as a game model G, where the attacker seeks the maximum gain from the attack, and the defender aims to minimize the loss. This use a stochastic game model to construct the social engineering attack model. In this model, the different targets have different attributes, and the other attributes have different vulnerabilities. There are interactions between the target's vulnerability and the social engineering attack, and finally, a stochastic game model of social engineering based on the attributes of the target is designed. The model optimization problem can be defined as follows.

Definition 3: Social engineering defense security decision optimization problem: Using quantified rewards as input, the social engineering defense security decision optimization problem is to find the optimal attack strategy a^* and defense strategy d^* for $a \in A$ such that $U_A\left(a^*, d^*\right) = MaxU_A\left(\mathrm{a}, d^*\right)$ and for $d \in D$ such that $U_D\left(a^*, d^*\right) = MaxU_D\left(a^*, d\right)$.

The utility function in the entire social engineering system, mainly considering the interaction between the target vulnerability and attacker, is quantified by the attack utility function $U_A\left(a^*, d^*\right)$ and the defense utility function $U_D\left(a^*, d^*\right)$. The final social engineering defense security decision optimization objective is to solve the Nash equilibrium. Therefore, $\left(a^*, d^*\right)$ is the optimal strategy for social engineering defense security decisions. The attacker's optimal attack strategy is a^* because this yields the maximum attack gain, and the defender's optimal defense strategy is d^* because this yields the minimum system loss.

3.2 Improved and Optimized Q-Learning Algorithm

In social engineering defense strategies, traditional approaches use Q-learning. This is because Q-learning algorithms converge quickly and can compute optimal policies. This method is widely used; however, as the system's complexity increases, the system's unstable and dynamic nature leads to an increase in the convergence time of the Q-learning method. Researchers have proposed relevant solutions combined with deep learning to improve the convergence speed of Q-learning algorithms. However, these solutions require a large amount of computation and often do not guarantee the algorithm's convergence in computing the optimal policy. The traditional formulation of the Q algorithm can be expressed as

$$Q(s,a,d) = (1-\alpha)\,Q(s,a,d) + \alpha\,(R(s,a,d) + \gamma V(s'))) \tag{6}$$

where s' is the next state, γ is the discount factor and the iterative learning rate can be denoted as $\alpha = 1/(t+1)^{\omega}$. $R(s,a,d)$ denotes the gain under attack action a and defense action d, where $V(s')$ denotes the maximum expected Q value for the next state. $V(s')$ can be set as

$$V(s') = maxmin\pi\,(s')\,Q(s',a',d') \tag{7}$$

Theorem 1: The multi-attribute quantitative reinforcement learning model based on human weakness proposed in this paper is convergent, and the reward sequence {Q_t }_(t→∞) is the optimal rewards Q^∗.

Proof:

According to Eqs. (6) and (7).

$$\begin{aligned} &Q^{*}(s,a,d)\\ &\quad=\sum\nolimits_{s'\in S} P(s' \mid s,a,d)\,(R(s,a,d)\\ &\quad+\alpha\,(maxmin\pi\,(s')\,Q(s',a',d')))\\ &\quad=E[F_tQ^{*}] \end{aligned} \tag{8}$$

then it can be computed $\| F_tQ - F_tQ^{*} \|$

$$\begin{aligned} &\| F_tQ - F_tQ^{*} \|\\ &=\underset{s,s'\in S}{min}\ max\,|\gamma\pi(s)Q(s) - \gamma\pi\,(s')\,Q^{*}\,(s')|\\ &\le \gamma\,|\pi(s)Q(s) - \pi\,(s')\,Q^{*}\,(s')| \end{aligned} \tag{9}$$

In order to prove convergence, it is necessary to prove that the sequence $\{Q_t\}_{t\to\infty}$ is the most optimal strategy Q^{*}

$$|\pi(s)Q(s) - \pi\,(s')\,Q^{*}\,(s')\,| \le \| Q - Q^{*} \| \tag{10}$$

According to π(s) and π(s′), we have

$$\begin{aligned} &|\pi(s)Q(s) - \pi\,(s')\,Q^{*}\,(s')\,|\\ &=\pi\,(s')\,Q\,(s') - \pi\,(s')\,Q^{*}(s)\\ &\le \pi\,(s')\,Q\,(s') - \pi\,(s')\,Q^{*}(s)\\ &\le \pi\,(s')\,\|\,Q(s) - Q^{*}\,(s')\,\|\\ &=\|\,Q(s) - Q^{*}\,(s')\,\| \end{aligned} \tag{11}$$

If π (s) Q (s) − π (s^′) Q^∗ (s^’) ≥ 0

$$\begin{aligned} &|\pi(s)Q(s) - \pi\,(s')\,Q^{*}\,(s')\,|\\ &=\pi(s)Q(s) - \pi\,(s')\,Q^{*}\,(s')\\ &\le \pi(s)Q(s) - \pi(s)Q^{*}\,(s')\\ &\le \pi(s)\,\|\,Q(s) - Q^{*}\,(s')\,\|\\ &=\|\,Q(s) - Q^{*}\,(s')\,\|\,. \end{aligned} \tag{12}$$

can get

$$\begin{aligned} & \| F_t Q - F_t Q^* \| \\ & \leq \gamma \mid \pi(S) Q(s) - \pi\,(s')\, Q^*\,(s') \\ & \leq \gamma \parallel Q - Q^* \parallel \\ & \leq \gamma \parallel Q - Q^* \parallel + \lambda_t \end{aligned} \tag{13}$$

According to the above proof, the model is convergent, where t he sequence $\{Q_t\}_{t\to\infty}$ is the most optimal strategy Q^*. To speed up the computation of Q-learning, a parallel computation method is designed in this paper to implement the IQL algorithm. By setting up multiple learning processes, each of which is attacker-centric, and ensuring that all learning is performed independently, the computational complexity of IQL is lower than that of the QL algorithm, and it converges more easily.

The Q-value update mechanism is also optimized to further improve the computation speed. Different learning processes can be updated simultaneously in parallel for the current state of the Q-value. There is no need to update after learning. All Q-learning procedures are performed simultaneously and synchronously to update the values, using the updated Q-values to update the previous Q-values. This is faster and more effective; the algorithm is shown in Algorithm 1.

Algorithm 1. Improved Q-Learning Algorithm (IQL)

```
Require: Game Model G, SE (social engineering) environment.
Ensure: Rewards $U_A$, $U_B$; Strategies $\pi_A, \pi_B$
Initialize global Q tables $QT_A$, $QT_D$
Initialize $p = 0$, cycles_nums = M, and max_len = L
for each learning process i parallel
    for $p < M$ or $\left(\| U_A^{i*} - U_A^i \| \leq \dot{\xi}$ and $\| U_D^{i*} - U_D^i \| \leq \xi\right)$ do
        Initialize $s^i = s_0$ and $r^i = 0$
        While $s^i \neq finalstate$ or $r^i = 0$ do
            Use the greedy algorithm to select action pairs $(a^i, d^i)$
            next state $s_{p+1}^i$
            Calculate attack gain $R_A^i\left(s_p^i, a^i, d^i\right)$, Calculate defense gain $R_D^i\left(s_p^i, a^i, d^i\right)$
            Update the game attack matrix $GM_A^i$ and game defense matrix $GM_D^i$ according to $QT_A$,$QT_D$
            Computational Attack Defense Strategy $\left(\pi_A^i\left(s_{p+1}^i\right), \pi_D^i\left(s_{k+1}^i\right)\right)$
            Update Q value
            Update $QT_A$, $QT_D$
           $r^i + +$
        end while
      $p + +$
    end for
  end foreach
  Calculate rewards $U_A$, $U_B$
  Return Optimal attack strategy $\pi_A$, defense strategy $\pi_B$, attack gain $U_A$, and defense gain $U_B$
```

According to Algorithm 1, this paper quantifies two global q variables, Q_A and Q_D. The algorithm uses parallel computation by calculating multiple attackers' learning processes at the same time, and all the computations constantly update the states of Q_A and Q_D. Therefore, the Q value can be guaranteed to be updated synchronously. For attacker j in the algorithm, the attacker performs an attack action a^j, and the defender performs the corresponding defense action d^j. Based on the attacker's attack action and the defender's defense action, an attack gain is obtained, and the state when the reward is oreceived A gain matrix GM is generated based on the states and rewards. A final gain and an optimal strategy are output by continuously cycling until all the attackers' learning processes are completed.

4 Experimental Results and Analysis

In this chapter, we construct a simplenatural social engineering system and analyze the simulation results.

4.1 Experimental Setup

The social engineering system mainly simulates phishing attacks, and the 'ystem's architecture is shown in Fig. 2. The whole structure includes the attack preparation, attack route, attack implementation, and attack gain stages. Attack preparation comprises collecting information from public resources, performing information system queries, collecting data from users, and writing scripts; attack target selection includes determining attack route nodes, such as phishing websites, phishing emails, and phishing SMSs; the attack implementation stage includes influencing the target, conducting psychological exploitation and script exploitation, assessing participant behavior, and building trust; and the attack gain stage mainly involves acquiring device permissions, obtaining sensitive information, and influencing target behavior.

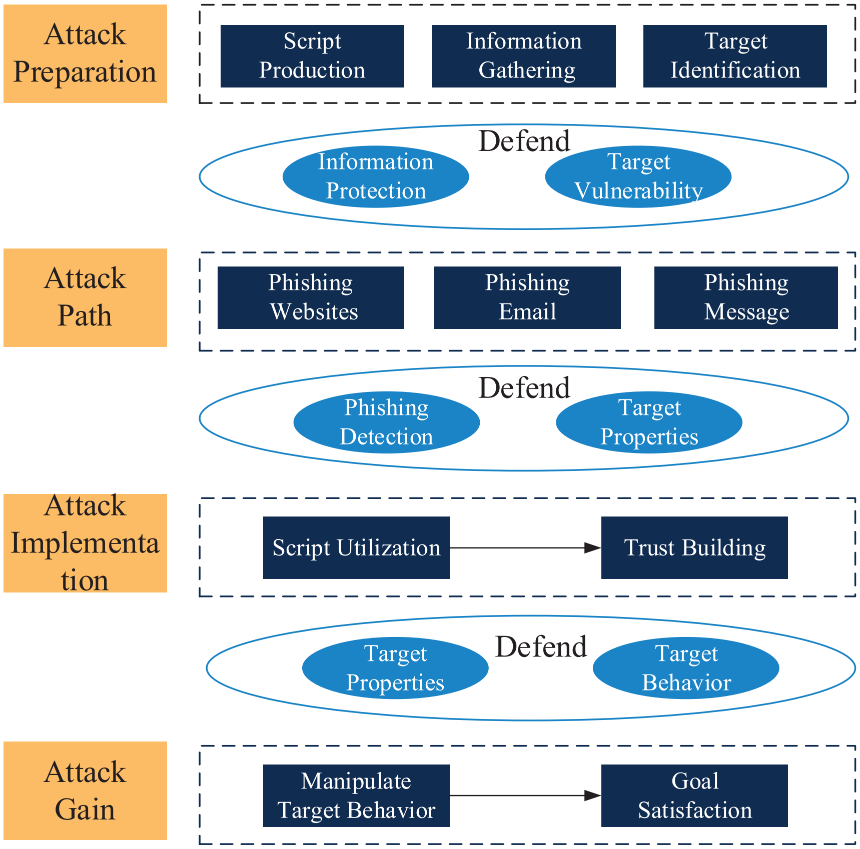

Figure 2: Experimental environment

The values of IL, CL, and TA for all nodes are shown in Table 3, where the three values of information search N1 are set to 2, 1, and 2; for the attack state N2, the three values are set to 1, 2, and 1; for gaining trust, N3, the three values are set to 2, 3, and 3; and for performing action N4, the three values are set to 3, 2, and 3.

Table 3: Node information

Nodes	IL	AL	DS
Information research N1	2	1	2
Attack path N2	1	2	1
Trust building N3	2	3	3
Manipulating target behavior N4	3	2	3

The social engineering attacker, through the collection of target information, discovers the target attributes, finds the arget's vulnerability, and executes an attack on the target, and the purpose of the attack is to obtain sensitive information or goods. The Common Attack Pattern Enumeration and Classification (CAPEC) database, developed and maintained by MITRE, records known cyber attack patterns [26]. The vulnerabilities are shown in Table 4.

Table 4: Vulnerability information

Vulnerability no.	CAPEC ID	Description
V1	CAPEC 118	Obtain target information
V2	CAPEC 98	Detecting vulnerabilities
V3	CAPEC 427	Target properties
V4	CAPEC 416	Manipulate target behavior
V5	CAPEC 173	Action spoofing
V6	CAPEC 151	Identity spoofing
V7	CAPEC 137	Parameter injection

Table 5 lists all the states of the game; if the attacker discovers a vulnerability to an attack and exploits it, a state transition occurs. If the attacker does not find an attack state, the attacker takes no action. The attack actions are shown in Table 6, and the defense actions are shown in Table 7.

Table 5: State information

States	Description
S1	Original state
S2	Obtain target information
S3	Send social engineering attacks
S4	Gain the trust of the target
S5	Execute attacker operations

Table 6: Action descriptions

Action	Description
a1	Utilizing V1 on N1
a2	Utilizing V3 on N1
a3	Utilizing V7 on N2
a4	Utilizing V6 on N2
a5	Utilizing V3 on N3
a6	Utilizing V5 on N3
a7	Utilizing V5 on N4
a8	Utilizing V4 on N4

Table 7: Defense descriptions

Action	Description
d1	Check personal information exposure.
d2	Robust cybersecurity training
d3	Implement an audit log written to a separate host, validated before use
d4	Authentication processes, multifactor authentication
d5	Cybersecurity training
d6	Avoid clicking suspicious links; robust cybersecurity training
d7	Robust cybersecurity training
d8	Robust cybersecurity training

The corresponding sets of defense and attack strategies are shown in Table 9.

4.2 Experimental Analysis

In this paper, we introduce the concepts of players and strategies and construct a model of reinforcement learning using players and strategies with a custom nature. In the game model of this paper, there are two players, and each player learns a new strategy. Each player's strategy is formulated according to the player's current state and the actions taken. The player's probability is updated by observing the opponent's behavior and changing the action accordingly. The best behavioral strategy is finally acquired through reinforcement learning for maximum benefit. The model in this paper has five states, and the reward of each state is shown in Fig. 3. As shown, the stochastic game reaches a Nash equilibrium after 2300 iterations. Table 8 lists the average rewards for the five states. According to the table, the rewards of S1 and S2 are essentially the same.

Table 9 lists the optimal strategies of the attackers and defenders in all states for each actual situation. It can be seen that in the initial stage of IQL, the probabilities of the players' decisions are random. As the number of iterations of the overall tethered model increases, the agent can obtain information about the adversary, and the best defense strategies are finally accepted by using the solved game matrix. According to the IQL algorithm proposed in this paper, the optimal strategy for the attacker and the defender are solved. The implementation results show that the method proposed in this paper can effectively obtain the optimal defense strategies.

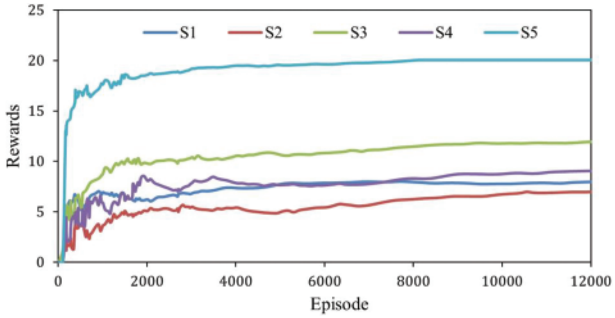

Figure 3: IQP process rewards in different states

Table 8: Attack and defense rewards

States	Attacker rewards	Defender rewards
S1	8.32	−8.32
S2	7.14	−7.14
S3	12.23	−12.23
S4	9.18	−9.18
S5	20	−20

Table 9: Optimal attack and defense strategies

States	Attacker strategies	Defender strategies
S1	{0.32, 0.21, 0.47}	{0.55, 0.24, 0.21}
S2	{0.40, 0.60}	{0.34, 0.33, 0.33}
S3	{0.40, 0.60}	{0.33, 0.34, 0.33}
S4	{0.35,0.65}	{0.46,0.54}
S5	{1,0}	{0.81,0.19}

In this paper, we propose an improved Q-learning algorithm that utilizes parallel computing to improve the efficiency of the computation. We conducted experiments using 1, 10, 20, and 30 parallel learning states to verify the relationship between multiple parallel learning processes and decision-makingsimilar. The results are shown in Fig. 4. Increasing the number of learning processes within a specific range can effectively improve learning efficiency and lead to fast convergence. However, if 30 learning processes are set, the efficiency will be small, and the decision time will increase due to the overly complicated iterative process. The experiment shows that increasing the number of learning methods within a specific range can effectively reduce the strategy learning time, but using too many learning processes will increase the strategy learning time.

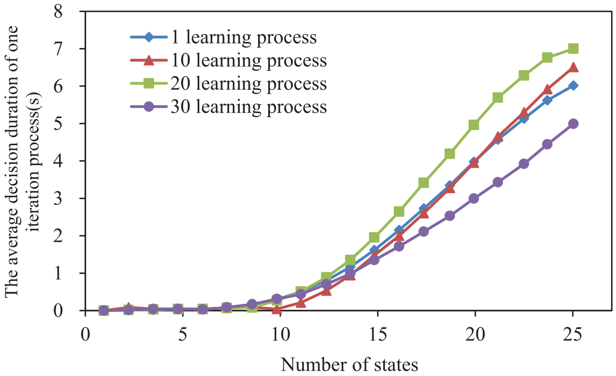

Figure 4: The decision duration of one iterative process for different numbers of learning processes

According to the literature survey, few studies apply game theory and reinforcement learning algorithms to the problem of social engineering defense strategy generation problem. Using different performance indicators and characteristics to compare and analyze the algorithm in this paper cannot prove the performance of the algorithm. Two commonly used reinforcement learning algorithms QL algorithm, and DQN algorithm [27], are used in this paper to compare and analyze the effectiveness and performance of the IQL algorithm proposed in this paper.

In this paper, the simulation experiment of social engineering defense strategy generation is carried out in the same network simulation environment, and the statistical analysis of the experimental data is carried out. The experimental results show that the simulation results obtained using Q-learning, DQN, and the algorithm in this paper in the social engineering random game scenario are shown in Fig. 5.

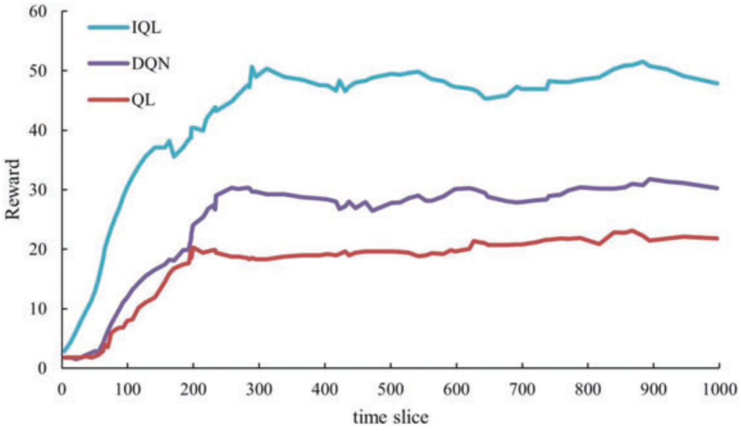

Figure 5: Comparison of simulation results obtained by using three different strategy-solving algorithms in social engineering attack-defense game

The above results show that in the social engineering random game scenario, the defender rewards obtained by the IQL solution are significantly greater than those obtained by the Q-Learning and DQN solutions. This experiment shows that defenders can reach their optimal defense strategy faster under the IQL reinforcement learning mechanism than other deep learning algorithms.

In a single iteration, the IQL algorithm can update different irrelevant states simultaneously, which can be considered an independent process. Therefore, the decision time of the IQL algorithm is always shorter than that of the QL algorithm. To prove this conclusion, we set up comparison experiments to calculate the policy learning time using the QL algorithm, the DQN algorithm, and the algorithm in this paper with different greedy values. The experimental results are shown in Fig. 6. The IQL algorithm has the shortest policy learning time for different greedy values. Under the condition of e = 0.04, the IQL algorithm performs best on the minicamp testbed, with the highest number of states. QL runs for 15400 steps, DQN runs for 17400 steps, and IQL runs for only 14000.

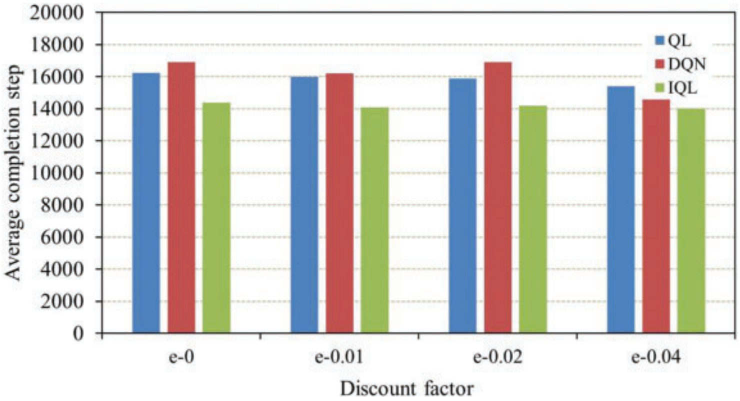

Figure 6: Average completion times of the QL, DQN, and IQL algorithms with different discount factors

This is because the more states there are in the iterative process, the more computation is required and the higher the computational complexity. In general, the best performing iteration over the least is when e-0.04, and the completion time of the IQL algorithm can be reduced by 9.1% and 19.4% compared to the DQN and QL algorithms. The calculation formula is as follows:

$$c_{ab} = (t_a - t_b)/t_a$$

In the above equation, c_{ab} denotes the rate of reduction of consumption time of algorithm b over algorithm a, t_a denotes the time consumed by algorithm a, and t_b denotes the time consumed by algorithm b.

5 Conclusion

In this paper, we propose a reinforcement learning model based on game theory that can generate optimal social engineering defense strategies for social engineering attack models to enhance social engineering defenses and reduce losses. Since the traditional methods of social engineering defense strategy generation consider only the technical aspects of defense, humans are idealized and regarded as having consistent properties, which leads to an unsatisfactory defense strategy. Considering the interaction between target vulnerabilities and social engineering attacks, a quantification mechanism based on multiple target attributes is proposed. The attacker and defender are also modeled as a two-sided stochastic game. The optimal defense strategies of the defender are analyzed. To improve the real-time performance and effectiveness of the defense, a Q-learning algorithm based on the game is optimized, and a multistate independent parallel learning optimization method is proposed to improve the learning efficiency and to generate the optimal defense strategy quickly. According to the experimental simulation results, the average time needed to create the optimal policy is reduced by 12.5%∼20% with the optimization method proposed in this paper compared with the QL and DQN algorithms. However, there are still some significant problems for the process in this paper; for example, the model construction could be more rough and sufficiently detailed for parallel task scheduling algorithm research, and the attack recovery method presented here could be improved.

Acknowledgement: Thanks are due to Kangfeng Zheng and Chunhua Wu for assistance with the experiments and to Xiujuan Wang and Bin Wu for valuable discussion. We are grateful for the assistance of the Beijing Natural Science Foundation (4202002).

Funding Statement: This research was funded by the Beijing Natural Science Foundation (4202002).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] APWG, *Phishing Activity Trends Reports*. 2022. [Online]. Available: https://apwg.org/trendsreports/

[2] Dmarcian, *2021 FBI Internet Crime Report*. 2022. [Online]. Available: https://dmarcian.com/2021-fbi-internet-crime-report/

[3] K. Zheng, T. Wu, X. Wang, B. Wu and C. Wu, “A session and dialogue-based social engineering framework,” *IEEE Access*, vol. 7, pp. 67781–67794, 2019.

[4] A. Yasin, L. Liu, T. Li, R. Fatima and W. Jianmin, “Improving software security awareness using a serious game,” *IET Software*, vol. 13, no. 2, pp. 159–169, 2019.

[5] J. M. Hatfield, “Social engineering in cybersecurity: The evolution of a concept,” *Computers & Security*, vol. 73, pp. 102–113, 2018.

[6] S. M. Albladi and G. R. S. Weir, “User characteristics that influence judgment of social engineering attacks in social networks,” *Human-Centric Computing and Information Sciences*, vol. 8, no. 1, pp. 5, 2018.

[7] S. Seo and D. Kim, “SOD2G: A study on a social-engineering organizational defensive deception game framework through optimization of spatiotemporal MTD and decoy conflict,” *Electronics*, vol. 10, no. 23, pp. 3012, 2021.

[8] Q. Zhu and T. Basar, “Game-theoretic methods for robustness, security, and resilience of cyberphysical control systems: Games-in-games principle for optimal cross-layer resilient control systems,” *IEEE Control Systems Magazine*, vol. 35, no. 1, pp. 46–65, 2015.

[9] J. Pawlick, E. Colbert and Q. Zhu, “A game-theoretic taxonomy and survey of defensive deception for cybersecurity and privacy,” *ACM Computing Surveys (CSUR)*, vol. 52, no. 4, pp. 1–28, 2019.

[10] K. Zhong, Z. Yang, G. Xiao, X. Li, W. Yang *et al.,* “An efficient parallel reinforcement learning approach to cross-layer defense mechanism in industrial control systems,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 11, pp. 2979–2990, 2022.

[11] R. Elderman, L. J. J. Pater, A. S. Thie, M. M. Drugan and M. Wiering, “Adversarial reinforcement learning in a cyber security simulation,” in *Proc. of the 8th Int. Conf. on Agents and Artificial Intelligence (ICAART)*, Rome, Italy, pp. 559–566, 2017.

[12] K. Chung, C. A. Kamhoua, K. A. Kwiat, Z. T. Kalbarczyk and R. K. Iyer, “Game theory with learning for cyber security monitoring,” in *Proc. of the 2016 IEEE 17th Int. Symp. on High Assurance Systems Engineering (HASE)*, Orlando, FL, USA, pp. 1–8, 2016.

[13] K. Durkota, V. Lisy, B. Bošansky and C. Kiekintveld, “Optimal network security hardening using attack graph games,” in *Proc. of the 24th Int. Conf. on Artificial Intelligence*, Buenos Aires, Argentina, pp. 526–532, 2015.

[14] K. Durkota, V. Lisý, B. Bošanský, C. Kiekintveld and M. Pěchouček, "Hardening networks against strategic attackers using attack graph games," *Computers & Security*, vol. 87, pp. 101578, 2019.

[15] A. Ridley, "Machine learning for autonomous cyber defense," *The Next Wave*, vol. 22, no. 1, pp. 7–14, 2018.

[16] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Proc. of the Eleventh Int. Conf. on Int. Conf. on Machine Learning*, Rutgers University, New Brunswick, NJ, pp. 157–163, 1994.

[17] J. Hu and M. P. Wellman, "Nash q-learning for general-sum stochastic games," *Journal of Machine Learning Research*, vol. 4, pp. 1039–1069, 2003.

[18] J. Heinrich and D. Silver, "Deep reinforcement learning from self-play in imperfect-information games," arXiv preprint arXiv:1603.01121, 2016.

[19] R. Yang, K. Zheng, B. Wu, D. Li, Z. Wang *et al.,* "Predicting user susceptibility to phishing based on multidimensional features," *Computational Intelligence and Neuroscience*, vol. 2022, pp. 7058972, 2022.

[20] R. Yang, K. Zheng, B. Wu, C. Wu and X. Wang, "Prediction of phishing susceptibility based on a combination of static and dynamic features," *Mathematical Problems in Engineering*, vol. 2022, pp. 2884769, 2022.

[21] C. Liu, F. Tang, Y. Hu, K. Li, Z. Tang *et al.,* "Distributed task migration optimization in MEC by extending multi-agent deep reinforcement learning approach," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 7, pp. 1603–1614, 2021.

[22] F. Mouton, M. M. Malan, L. Leenen and H. S. Venter, "Social engineering attack framework," in *IEEE 2014 Information Security for South Africa*, pp. 1–9, 2014.

[23] A. Ghasempour, "Internet of things in smart grid: Architecture, applications, services, key technologies, and challenges," *Inventions*, vol. 4, no. 1, pp. 22, 2019.

[24] K. P. Mayfield, M. D. Petty, T. S. Whitaker, J. A. Bland and W. A. Cantrell, "Component-based implementation of cyberattack simulation models," in *Proc. of the 2019 ACM Southeast Conf.*, New York, NY, ACM, pp. 64–71, 2019.

[25] M. S. Barnum, *Common attack pattern enumeration and classification (CAPEC) schema*. Department of Homeland Security, 2008. [Online]. Available:https://capec.mitre.org/

[26] R. Mitchell and R. Chen, "Modeling and analysis of attacks and counter defense mechanisms for cyber physical systems," *IEEE Transactions on Reliability*, vol. 65, no. 1, pp. 350–358, 2015.

[27] H. Van Hasselt, A. Guez and D. Silver, "Deep reinforcement learning with double q-learning," in *Proc. of the AAAI Conf. on Artificial Intelligence*, Phoenix, Arizona, USA, pp. 2094–2100, 2016.