SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2026, Vol. 69, Iss. 3, 132109:1–132109:14
https://doi.org/10.1007/s11432-025-4530-7

# A Stackelberg game based deception defense strategy against APT under resource constraints

Xin DENG[1,2], Pengdeng LI[3,4,5*], Chenguang WANG[3,5], Rui WANG[3,5], Yuan LIU[3,4,5], Weihong HAN[2] & Zhihong TIAN[3,4,5*]

[1]*School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2]*Department of New Networks, Pengcheng Laboratory, Shenzhen 518055, China*
[3]*Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou 510006, China*
[4]*Guangdong Key Laboratory of Industrial Control System Security, Guangzhou University, Guangzhou 510006, China*
[5]*Huangpu Research School of Guangzhou University, Guangzhou 510530, China*

Received 12 March 2025/Revised 21 May 2025/Accepted 30 July 2025/Published online 28 January 2026

**Abstract** The advanced persistent threat (APT) has become a major challenge in cybersecurity due to its concealment, persistence, and complexity. Traditional passive defense methods, owing to their static and reactive characteristics, struggle to provide sufficient protection against APT attacks. By contrast, active defense solutions have gained increasing attention due to their ability to shift the defender from passivity to taking the initiative. Deception defense is a widely used active defense method to prevent threats in advance by deploying deception resources. Despite its development, existing deception methods often assume that the defender and the APT attacker take actions simultaneously. In practice, due to his advanced nature, the APT attacker can first observe the defender's strategy through reconnaissance and then make his best responses to the defender's strategy. In this paper, we develop a game model to accurately characterize this worst-case scenario from the defender's perspective. Specifically, we establish a Stackelberg game, called cyber deception Stackelberg game (CDSG), where the defender first announces the allocation strategy of limited deception resources to a set of services with the anticipation that the APT attacker will best respond to her strategy, and the attacker determines his action after observing the defender's strategy. In the game model, we also consider that different types of deception resources have varied probabilities of successfully capturing the APT attacker. Given the game model, we then devise a gradient descent based algorithm to solve CDSG for the equilibrium, which offers the defender a robust deception resource allocation strategy. Finally, experiments are conducted to verify the effectiveness of the deception defense strategy in defending against APT attackers and its superiority over several baselines.

**Keywords** cybersecurity, advanced persistent threat, deception defense, Stackelberg game, equilibrium strategy

**Citation** Deng X, Li P D, Wang C G, et al. A Stackelberg game based deception defense strategy against APT under resource constraints. Sci China Inf Sci, 2026, 69(3): 132109, https://doi.org/10.1007/s11432-025-4530-7

## 1 Introduction

In today's digital era, networks have become an indispensable and integral component of human society, profoundly shaping personal lifestyles, business operations, and governmental governance models. The Internet and various information systems not only enable seamless global communication and collaboration, but also underpin the operation of critical infrastructures, such as healthcare, energy, and transportation [1]. As society's reliance on networks and digital technologies continues to deepen, ensuring the security and stability of cyberspace has become increasingly critical.

However, with the rapid advancement of network technologies, cyber threats have also exhibited trends of increasing scale and complexity. From financial fraud and data breaches to attacks targeting critical infrastructures, malicious actors exploit system vulnerabilities to carry out disruptive activities, steal sensitive information, or pursue political and economic gains [2,3]. Among these threats, APT [4] stands out as one of the most serious challenges in the field of cybersecurity due to its stealthiness, persistence, highly targeted nature, and sophisticated attack techniques. APT attackers typically target high-value entities such as government agencies, military organizations, and multinational corporations, aiming to maintain long-term unauthorized access to or disrupt their systems. The risks posed by such attackers not only threaten corporate security but also have serious implications for national development.

* Corresponding author (email: pdli@gzhu.edu.cn, tianzhihong@gzhu.edu.cn)

 

Passive defense, such as firewalls and intrusion detection systems, mainly relies on fixed rules and known features of cyber attacks. However, APT attacks often use zero-day vulnerabilities, social engineering, and advanced stealth techniques, and constantly adjust strategies during the attack to evade detection tools and defense systems [5]. This makes it difficult for passive defense to identify these changing attack activities. In addition, APT attackers will also establish persistent channels by implanting backdoors, Web Shells, and intranet proxies, making it difficult for passive defense to detect their subsequent activities.

Therefore, as discussed earlier, relying solely on passive defense can no longer effectively resist APT attacks, and building an active defense system has become an inevitable choice [6]. The core difference between active defense and passive defense lies in the change in their response methods, from simple passive responses to active threat perception and countermeasures [7]. Especially when facing complex and highly concealed attack scenarios, threat perception capability is crucial in the defense system. Compared with traditional defense methods, by deploying deception defense resources such as honeypoints [8] in the network, it is possible to perceive the threatening behavior of the attacker in advance before the attacker completes the penetration [9, 10]. In addition, deception defense can also mislead attackers. By guiding attackers to false targets for observation, the technical tools and attack strategies used by them are exposed, providing key clues for subsequent defense response and attack tracing [11].

Since APT attacks are persistent, attackers often conduct long-term detection, penetration, and attacks on target systems. This persistence makes static deception defense measures easily ineffective when facing APT attacks. If defense resources remain unchanged, attackers can gradually identify bait resources through repeated probing, thereby avoiding deception traps and causing deception defense failure. More seriously, smart attackers may also deliberately detect identified deception resources, causing a large amount of log redundancy and interfering with the judgment of defenders. Moreover, the resources of defenders are often limited. Deploying too many deception resources will affect the normal operation of the network. Thus, it is impossible to deploy deception resources for every service in the network. In order to deal with the above challenges, the deception defense system must introduce a dynamic decision-making mechanism to more effectively and efficiently deploy the limited deception resources.

Game theory, as a systematic decision-making analysis method, has been widely used to help defenders achieve optimal defense in dynamic environments [12, 13]. Although existing game-theoretic based deception defenses have achieved some success, they are overly optimistic when facing APT, as they fail to fully consider the characteristics of APT attacks.

To better deal with APT, a cyber deception Stackelberg game model (CDSG) is proposed in this paper. The model aims to optimize the deployment strategy of cyber deception resources to combat APT under resource constraints. Unlike other network threats, APT is characterized by its long duration and strong concealment. Therefore, CDSG assumes that APT attackers are sufficiently powerful and considers the most pessimistic scenario: the defender has no information about the attacker, while the attacker has complete knowledge of the defender's strategy before taking any action and responds with the best strategy. The defender minimizes the lower bound of loss by minimizing the maximum loss. The main contributions of this study are summarized as follows.

- Taking the characteristics of APT attacks and limited defense resources into account, a cyber dynamic deception method based on the Stackelberg game is proposed. This method captures the sequential relationship in which the defender always deploys deception resources in the system first, while the APT attacker conducts information gathering before launching an intrusion.
- To solve the Nash equilibrium of the game, we first transform the problem into a constrained optimization problem, and then solve the problem by constructing a gradient descent method.
- Experiments are conducted using the configuration of a real subnet, and the effectiveness of CDSG was validated through numerical simulations and comparisons with different decision-making methods. The experimental results demonstrate that CDSG significantly reduces the maximum loss across all services by coordinating different types of deception resources, and it performs even better when the quantity of resources is relatively small.

The rest of this paper is organized as follows. Section 2 presents the related work. Section 3 describes the modeling process of CDSG in detail. Then, the solution of the optimal deception strategy is introduced in Section 4. We evaluate and compare CDSG in Section 5. Finally, Section 6 summarizes this work and discusses potential future work.

## 2 Related work

In this section, we comprehensively review previous studies related to our work.

### 2.1 Game theory for APT and deception defense

Game theory is widely used to model the interaction between attackers and defenders in cybersecurity [14,15]. Feng et al. [16] enhanced the MTD strategy by incorporating information leakage, proposing a Bayesian Stackelberg game model to study the optimal moving strategies of the defender. They also demonstrated how signaling can be used to influence the attacker's behavior, thereby achieving a more efficient active defense. Liu et al. [17] proposed an APT deception game to describe the interactions between the defender and the attacker. To reduce computational complexity, they simplified the problem of finding the optimal deployment strategy into a decoy credential sequence problem and a decoy credential allocation problem. Wan et al. [18] proposed a hypergame model to address structural uncertainties in APTs. In this model, multiple APT attackers and a single defender engage in repeated games based on their perceptions. Zhang et al. [19] proposed an APT competitive game to demonstrate how strategy adjustments can reduce information leakage. Additionally, two learning mechanisms were introduced to assist the defender in identifying appropriate defense levels and efficiently allocating resources.

Many methods combine the SIR epidemic model to describe the evolution of node states in the attack and defense game. Yang et al. [20] modeled the APT repair problem as a game problem and obtained effective repair strategies through SIR models and differential game theory. Mi et al. [21] introduced Gaussian noise to characterize the stochastic effects during strategy execution and established a stochastic differential game model for attack-defense scenarios by integrating the SIR epidemic model. The optimal defense strategy is determined by solving for the saddle-point equilibrium, thereby achieving real-time defense decision-making. Bi et al. [22] addressed the lateral movement phase of the APT attack in the Industrial Internet of Things (IIoT) by establishing a node-level state evolution model and a Stackelberg game model, thereby proposing an effective APT defense strategy. Considering the different protection capabilities of different devices in IIoT, Ref. [23] classified the devices and constructed differential games to solve the optimal defense strategy. He et al. [24] utilized the SIR model to derive the differential equations of node states in a deception defense system and employed the deep Q-network to solve for the optimal configuration of deception assets. However, due to the stealthiness of APT attacks, it is difficult to accurately obtain the node states during the attack-defense process.

Given the constraints of technology, information, and other factors, both players may not be able to make optimal decisions during the game. Tan et al. [25] developed an evolutionary model for moving target defense strategies based on the Wright-Fisher process, taking into account the bounded rationality of both attackers and defenders. Jin et al. [26] addressed the differences and limitations in cognitive abilities between attackers and defenders, integrating regret minimization algorithms with evolutionary game theory to propose a decision-making method for cyber attack-defense games under bounded rationality. In [27], the authors introduced the decision-maker's degree of irrationality and the level of environmental security into the replicator dynamics equation of the differential game.

In specific scenarios, Xiao et al. [28] applied prospect theory to cloud storage defense, examining the interactions between defenders and subjective APT attackers under both pure and mixed strategies, and proposed a defense strategy based on Q-learning. Regarding the integration of information and physical systems in power grids, Ge et al. [29] proposed an optimal allocation method for smart grid defense resources based on game theory. They designed a two-layer game model for resource optimization: one layer involves an evolutionary game between defense nodes and attackers, while the other layer is a non-cooperative game among multiple defense nodes. Aiming at APT attacks in the Industrial Internet of Things, Tian et al. [30] proposed a dynamic bounded rational honeypot APT game model based on SDN. To describe bounded rationality, they employed prospect theory to characterize players' utilities when they have incomplete information about their opponent.

Additionally, Ye et al. [31] integrated differential privacy with game theory, proposing the use of a differential privacy-based approach to add noise to the system's configuration. Dong et al. [32] incorporated intuitionistic fuzzy theory into attack-defense games to more accurately depict the uncertainties and subjective preferences inherent in such games. He et al. [33] extended the single-stage FlipIt game model by introducing discount factors and transition probabilities, thereby constructing a multi-stage cyber deception model.

### 2.2 Other method

In addition to game theory, attack graphs and reinforcement learning (RL) have also been often used to optimize deception defense strategies. Ngo et al. [34] used attack graphs to model the active directory to help understand and respond to complex active directory attack environments. They combined mixed integer programming and heuristic methods to solve the problem of honeypot deployment in large-scale active directory attack graphs and generate near-optimal interception plans. Yoon et al. [35] proposed a three-layer attack graph to analyze the vulnerability and topology information of the network, helping identify high-risk hosts and attack paths, and providing decision

support for MTD to ensure better protection of critical assets. To defend against reconnaissance attacks in cloud environments, Li et al. [36] proposed a defense deception framework based on deep RL to generate optimal deception strategies to confuse reconnaissance attacks. Zhang et al. [37] proposed a host address mutation method based on deep RL for the reconnaissance phase to reduce the number of scan hits by attackers. To find adaptive policies for MTD, Eghtesad et al. [38] proposed a POMDP model. Ref. [39] proposed a grouped multi-agent deep reinforcement learning defense algorithm to divide all defenders into cooperative groups and allow defenders within each group to jointly optimize defense strategies by sharing information and experience. In [40], the authors leveraged multi-agent reinforcement learning to improve the effectiveness of MTD in web application protection. Zhu et al. [41] solved the optimal strategy for the multi-defender multi-attacker scenario based on multi-agent deep reinforcement learning.

### 2.3 Summary

In summary, the existing game theory based methods mainly solve Nash equilibrium according to the attacker's strategy space and maximize the overall expected effect. The attack graph based method mainly builds an attack graph based on known vulnerabilities to find the key path. Methods based on deep reinforcement learning need to establish a Markov decision process, which requires knowledge of the state space of the system and the transition probabilities between states. However, it seems too "optimistic" for complex attacks such as APT. APT is concealed and often uses zero-day vulnerabilities or other complex attack methods. It is difficult to build an attack graph and obtain all the attacker's strategy space. Meanwhile, APT attacks are very complex; it is difficult to accurately construct the state space and action space of APT attacks. Therefore, in this paper, based on the characteristics of APT, we construct a Stackelberg game from the goal of optimizing the lower limit of loss, and solve the optimal deception strategy by minimizing the maximum loss of the protected system.

## 3 Cyber deception Stackelberg game model

In the defense against APT attacks, defenders typically deploy defensive resources in advance, aiming to increase the cost of attacks and delay their progress as much as possible. Attackers, after observing or inferring the defensive strategy, will choose the optimal attack path, which consequently results in a dynamic strategic game between the two sides. The core characteristic of the Stackelberg game lies in its "leader-follower" relationship, where the leader acts first and designs strategies to maximize its own payoff, while the follower observes or infers the leader's strategy and makes rational and optimal responses based on its own interests. This is highly consistent with the solution to the optimal strategy for deception resource deployment in APT confrontation. Through Stackelberg game modeling, this dynamic sequence relationship can be effectively described, and defenders can be helped to formulate the optimal defense strategy by analyzing the possible responses of attackers, thereby maximizing the security of the system and increasing the cost to attackers. Therefore, CDSG models the problem of deception resource deployment using the Stackelberg game and derives the optimal deception resource deployment strategy by solving for the game's equilibrium. Specifically, in CDSG, the defender, as the leader, acts first by deploying deception resources in the network. The APT attacker, as the follower, takes action after observing the defender's deployment. Furthermore, to fully capture the high technical sophistication and difficulty in defending against APT attackers, they are considered to be smart enough to know the deployment strategy of deception defense when taking action, and will always choose the service with the highest intrusion benefit based on the deception strategy. The main notations mentioned in the method and their explanations are listed in Table 1.

### 3.1 Basic setting

As shown in Figure 1, during network planning, the network is usually divided into several functional subnets based on functions, departmental roles, etc. For instance, in the context of a school's network design, the network is often segmented into subnets for dormitories, teaching buildings, libraries, and other functional zones. This division serves as an effective and secure strategy, as it not only facilitates network management but also reduces potential security risks and lateral attack paths by isolating different subnets. Once attackers gain access to the internal network, they tend to conduct reconnaissance and perform lateral movements across the network. Therefore, deploying deception defenses within each subnet is essential to detect and respond to attackers' lateral movements.

Although deploying deception defense resources in each subnet is necessary, in practical applications, a functional subnet typically contains a large number of services. Deploying deception resources incurs costs related to hardware, software, and management, making these resources inherently limited. As a result, it is not feasible to deploy deception resources to protect every service. Therefore, when deploying deception defenses in each subnet, a critical

**Table 1** Main notations in CDSG.

| Notation | Explanations |
|---|---|
| $L$, $F$ | Players in the CDSG, $L$ represents the leader, i.e., the defender, and $F$ represents the follower, i.e., the attacker. |
| $\mathbf{N}$ | The set of services in a subnet. |
| $n$ | The number of services in a subnet. |
| $\mathbf{R}, \mathbf{C}$ | The vector of reward for successful intrusion and cost for failure. |
| $r_i, c_i$ | Reward for success of attacking service $i$ and cost for failure. |
| $k$ | The number of types of deception resources. |
| $\mathbf{S}$ | The number of all types of deception resources, a $k$-dimensional vector. |
| $s_j$ | The number of $j$-th deception resources. |
| $\mathbf{B}$ | Deception resource attraction ability, a $k$-dimensional vector. |
| $b_j$ | When the $j$-th deception resource protection service is used, if an attacker attacks the service, the probability of being attracted and discovered. |
| $p_1(i)$ | The probability that service $i$ deploys deception resources, and the attacker is attracted by the resources when attacking service $i$, resulting in the attack failure. |
| $p_2(i)$ | The probability that service $i$ deploys deception resources, but it does not attract attackers, and the attacker successfully invades service $i$. |
| $p_3(i)$ | The probability that service $i$ does not deploy deception resources and the attacker successfully invades service $i$. |
| $p_s(i)$ | The probability that an attacker successfully invades service $i$. |
| $p_f(i)$ | The probability that an attacker fail to invades service $i$. |
| $U^L(i), U^F(i)$ | The utility function of attacker and defender on service $i$. |
| $p_{i,j}$ | The probability of deploying a deception resource of type $j$ on service $i$. |
| $I_i$ | The importance of the service $i$. |

question arises: what strategy should be adopted to allocate deception resources to protect the services within the subnet such that the defensive effect is maximized?

### 3.2 Model formulation

Based on the problem given in the previous section, we formally describe our CDSG. As shown in Figure 2, within a subnet $N$, there are $n$ services, each providing different functions to users. Different types of deception resources have varying levels of effectiveness in deceiving the attacker. The defender will first deploy deception resources within the subnet. For example, when only ports are opened, the information provided is limited, and the attacker can easily detect that they are targeting a fake objective. However, if the deployed deception resource is an application, it is much harder for the attacker to identify that it is a fake target. The deception effect of a deception resource is defined as the deception resource's attraction ability. Specifically, if the attraction ability of a certain deception resource is 0.5, it means that when an attacker attacks a service, if this service deploys this type of deception resource for protection, the probability that the attacker is attracted by the deception resource and thus detected is 0.5. At the same time, we classify the defender's deception resources according to the difference in deception resource attraction ability. Specifically, let $k$ be the number of deception resource types, $\mathbf{B} = [b_1, b_2, \ldots, b_k]$ is a vector of length $k$, and $b_i$ represents the attraction ability of the $i$-th type of deception resource to attract attackers. Since each type of resource consumes different amounts of resources, the availability of each type of resource may not be the same, and the deception resources run independently without affecting each other. The quantity of deception resources is represented by $\mathbf{S} = [s_1, s_2, \ldots, s_k]$, where $s_i$ denotes the number of deception resources of type $i$. Due to resource constraints, we set $s_j < n$, $1 \leqslant j \leqslant k$. In the defender's mixed strategy, the probability that the defender deploys the $j$-th type of deception resource for service $i$ is denoted by $p_{i,j}$. Then, the defender's strategy is denoted as $p = (p_{i,j})_{1 \leqslant i \leqslant n}^{1 \leqslant j \leqslant k}$. From the constraint of the number of resources, we have $0 < p_{i,j} < 1$.

The APT attacker will act after the defender. When carrying out an intrusion, the attacker will choose one service from the $n$ services to target. Each service has a different level of importance in the business system, and

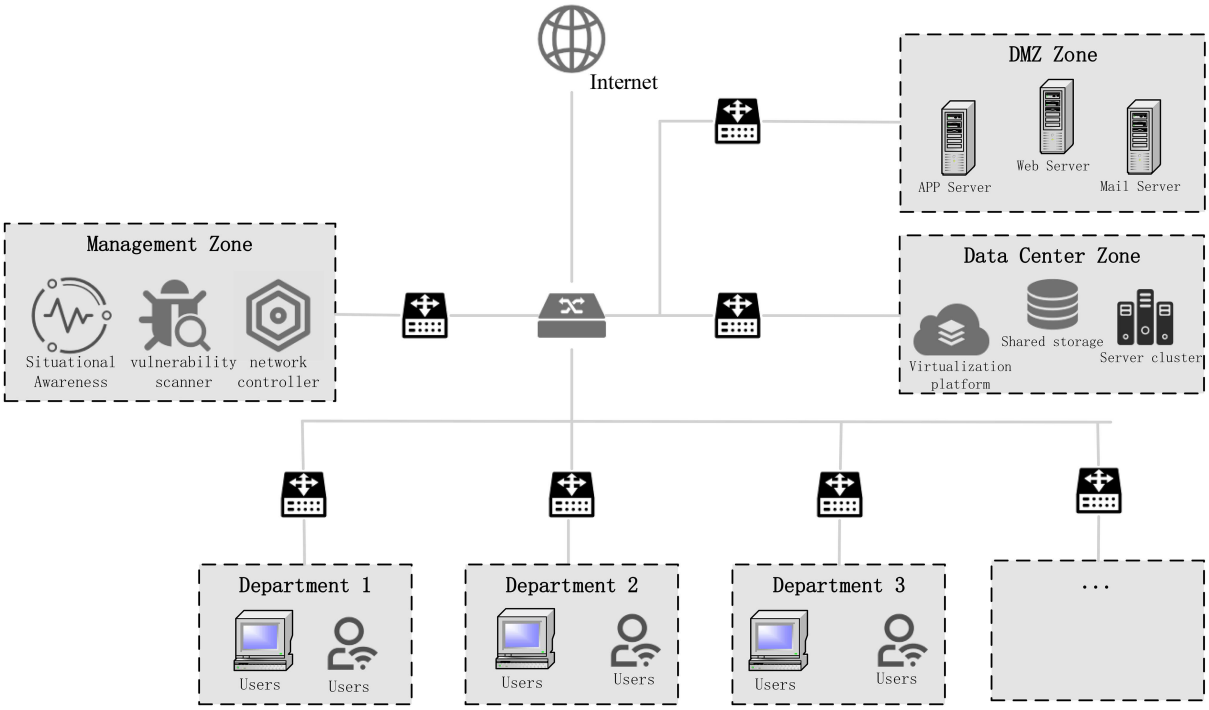


**Figure 1** Example of network planning.

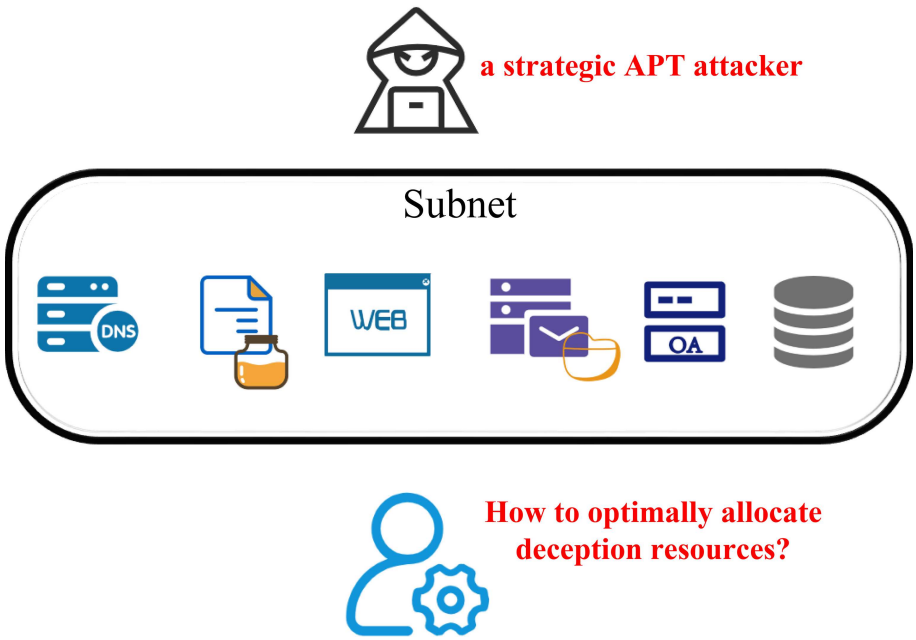


**Figure 2** Overview of the attack and defense scenarios described by the model.

the vulnerabilities present in each service vary. Therefore, each service has a unique reward and cost for the attacker. We define $\mathbf{R} = [r_1, r_2, \ldots, r_n]$ as the reward for a successful intrusion, and $\mathbf{C} = [c_1, c_2, \ldots, c_n]$ as the cost of a failed intrusion. It means that if the attacker chooses to attack service $i$, they will receive a reward of $r_i$ if successful, otherwise, incur a loss of $c_i$. The length of both vectors is $n$, and $r_i$ is always greater than $c_i$.

Because of the existence of the deception defense, there are three possible outcomes when the attacker targets service $i$. (1) Service $i$ has deployed deception resources, and the attacker is deceived by these resources, leading to a failed attack. (2) Service $i$ has deployed deception resources, but all of these resources do not deceive the attacker, and the attacker successfully infiltrates the service $i$. (3) Service $i$ has not deployed any deception resources, and the attacker successfully infiltrates service $i$. The probabilities of the three cases are denoted by $p_1$, $p_2$, $p_3$, respectively.

From the attacker's perspective, the first scenario is an unsuccessful attack, while the second and third scenarios are successful attacks. Therefore, for service $i$, the probability of a successful attack, $p_s(i)$, is equal to $p_1$, and the probability that the attack fails, $p_f(i)$, is equal to $p_2 + p_3$. The formulas are shown in (1) and (2).

$$p_f(i) = \prod_{j=1}^{k} (p_{i,j} \cdot (1 - b_j)) + \prod_{j=1}^{k} (1 - p_{i,j}), \tag{1}$$

$$p_s(i) = 1 - p_f(i). \tag{2}$$

For the attacker, the payoff at the $i$-th service is given by

$$U^F(p, i) = p_s(i) r_i - p_f(i) c_i. \tag{3}$$

For the defender, his task is to minimize system loss by utilizing deception resources. Therefore, the defender's payoff at the $i$-th service is

$$U^L(p, i) = -U^F(p, i). \tag{4}$$

In the attack-defense confrontation, a rational APT attacker always chooses the action strategy that maximizes their payoff. Therefore, in the cyber deception Stackelberg game, the attacker will always choose to compromise the service with the highest payoff. The defender, however, has an inherent disadvantage when taking action. Due to the concealment and complexity of APT attacks, it is impossible to accurately determine which service the APT attacker will attack. Thus, the optimal deception strategy for the defender can only be to minimize the maximum payoff of the attacker's targeted service, in order to reduce network losses to the greatest extent. Consequently, the optimal deception strategy corresponds to the Nash equilibrium of the Stackelberg game, which can be formalized as

$$\min \max U^F(p, i).$$

It can be observed that for the defender, CDSG considers a mixed strategy, while for the attacker, a pure strategy is considered. Let us now compare the case where the attacker adopts a mixed strategy: the probability of the attacker targeting services within the subnet is $q = [q_1, q_2, \ldots, q_n]$, where $q_i$ denotes the probability of the attacker selecting target $i$, and $\sum_{i=1}^{n} q_i = 1$, meaning the attacker must distribute their attack behavior across all targets with a total probability of 1. The total payoff for the attacker is the expected value of the payoffs across all targets, calculated as

$$U^F(p, q) = \sum_{i=1}^{n} q_i U^F(p, i),$$

where $U^F(p, i)$ is still given by (3). The attacker's objective is to choose a mixed strategy $q$ to maximize their total payoff, that is

$$\max_q U^F(p, q)$$
$$\text{subject to } \sum_{i=1}^{n} q_i = 1,\ \forall i \in n,\ q_i \geqslant 0.$$

Let $x$ represent the maximum payoff across all services. For $\forall i \in \{1, 2, \ldots, n\}$, we have $U^F(i) \leqslant x$, Therefore, we can conclude that $\forall q, U^F(p, q) \leqslant x$. The attacker always makes the best response when making decisions, so the attacker's strategy must be a pure strategy.

## 4 Solution of optimal deception strategy

From the above description, it can be seen that the core objective of the defender's optimal deception strategy is to minimize the maximum payoff that the attacker may achieve under their optimal strategy by adjusting the resource allocation probability $p_{i,j}$. This problem involves a nonlinear objective function and constraints, as well as coupling relationships between multiple variables, making direct solving quite challenging. Therefore, in the solution process, an auxiliary variable $z$ is first introduced to transform the problem into a single-level optimization

problem, expressed as

$$\begin{aligned}
&\underset{p,z}{\text{minimize}} && z \\
&\text{subject to} && i \in \{1,\ldots,n\}, \quad j \in \{1,\ldots,k\}, \\
&&& U^F(p,i) \leqslant z,\ 0 \leqslant p_{i,j} \leqslant 1,\ \sum_{i=1}^{n} p_{i,j} = s_j, \\
&\text{where} && U^F(p,i) = p_s(i) r_i - p_f(i) c_i, \\
&&& p_f(i) = \prod_{j=1}^{k} (p_{i,j} \cdot (1 - b_j)) + \prod_{j=1}^{k} (1 - p_{i,j}), \\
&&& p_s(i) = 1 - p_f.
\end{aligned}$$

It can be observed that $p_f(i)$ is a non-convex function in the form of a product, and the attacker's payoff function is a nonlinear function. Therefore, this is a single-level optimization problem with a nonlinear objective function and constraints. The constraints include the attacker's payoff constraint, the total resource constraint, and the variable range constraint.

Due to the nonlinearity and non-convexity of the problem, there may be multiple local optimal solutions, and it cannot be directly solved using analytical methods or simple optimization algorithms, which is a typical NP-hard problem. As a result, an iterative optimization approach is adopted to gradually adjust the variable values and approximate the global or an effective local optimal solution. Specifically, we construct a gradient descent method. In the optimization process, a reasonable initial value is determined as the starting point for iterative optimization. Then, the gradient is calculated, and adjustments are made to gradually approach the optimal solution while satisfying the constraint conditions.

The choice of the initial solution has a significant impact on both the efficiency of the solution process and the quality of the results. A reasonable initial solution can greatly reduce the number of iterations and accelerate convergence. To better select the initial values, the importance of the services is introduced. It is determined by the attack reward and the attack cost and is calculated using the formula: $I_i = r_i - c_i$. A very intuitive explanation is that the greater the importance, the more critical or vulnerable the service is. This means that the service provides higher benefits to the attacker, making the attacker more inclined to target it. Hence, the initial solution of $p_{i,j}$ is determined by the importance of services, as calculated by (5). This initialization method prioritizes allocating more resources to services with higher attack rewards or greater vulnerability.

$$p_{i,j}^0 = \frac{s_j I_i}{\sum_{i=1}^{n} I_i}. \tag{5}$$

After determining the initial solution, iterative optimization is performed using gradient descent on the objective function. When the change in the value of the objective function reaches a predefined threshold, it is considered converged, and the process stops. The overall process is described in Algorithm 1.

## 5 Experiment and analysis

This section takes a specific subnet within a network as an example. A simulation environment is constructed to evaluate the practical effectiveness of the optimal defense strategy derived from the proposed Stackelberg game model for cyber deception. First, we outline the specific services in the subnet and the vulnerabilities of each service. Then, we introduce the method and results of quantifying the attacker's reward when each service is successfully invaded and the cost of failed invasion based on the characteristics of the service and its vulnerabilities, as well as the classification and numerical quantification of deception resources. Finally, the effectiveness of CDSG is verified by experimental comparison with other methods. In addition, we are surprised to find that in CDSG, different types of deception resources do not work alone, but they cooperate to protect the system.

### 5.1 Setup of the experimental environment

To verify the effectiveness of the method, a simulated subnet environment is constructed to simulate the service configuration in the real network scenario. The subnet includes the following components.

• Database server. Used to store and manage enterprise data, providing efficient and reliable data storage and access services.

**Algorithm 1** Solving for the optimal deception strategy.

**Input:** The number of service $n$, the number of resource types $k$, rewards $\mathbf{R}$, cost $\mathbf{C}$, the ability to deceive for deception resource of each type $\mathbf{B}$, the number of deception resource of each type $\mathbf{S}$, threshold $\epsilon$, the maximum number of iterations $T$.
**Output:** The optimal resource allocation scheme $p$.
Initialize the importance vector: $\forall i,\ I_i = r_i - c_i$;
// Generate an initial defender strategy $p^0$
**for** $j = 1, \ldots, k$ **do**
  **for** $i = 1, \ldots, n$ **do**
    $p^0_{i,j} = \frac{s_j I_i}{\sum_{i=1}^{n} I_i}$;
  **end for**
**end for**
Calculate $z^0$ based on $p^0$;
**for** $t = 0, 1, 2 \ldots, T$ **do**
  // Calculate the attacker's current optimal goal
  $i^* = \arg\max_{i \in \{1,\ldots,n\}} U^F(p^t, i)$;
  // Compute the gradient
  $\nabla_p U^F(p, i^*) = \left[ \frac{\partial U^F(p, i^*)}{\partial p_{i,j}} \right]_{i,j}$;
  // Updated defender strategy
  $p^{(t+1)} = p^{(t)} - \eta_p \nabla_p U^F(p, i^*)$;
  $p^{(t+1)}_{i,j} = \max\left(0, \min(p^{(t+1)}_{i,j}, s_j)\right),\ 1 \leqslant i \leqslant n,\ 1 \leqslant j \leqslant k$;
  $z^{(t+1)} = \max_{i \in \{1,\ldots,n\}} U^F(p^{(t+1)}, i)$;
  $\delta = ||p^{(t+1)} - p^{(t)}|| + |z^{(t+1)} - z^{(t)}|$;
  **if** $\delta < \epsilon$ **then**
    Break;
  **end if**
**end for**
**return** $p^T$.

**Table 2** Service and vulnerability information.

| Service | Software | CVE ID | Vulnerability description |
|---|---|---|---|
| Database | MySQL | CVE-2024-21176 | Causing MySQL server to hang or crash repeatedly |
| OA | RockOA | CVE-2023-1773 | Leading to code injection |
| File | MinIO | CVE-2023-28432 | Causing information leakage and being able to log in as an administrator |
| Mail | Exchange server | CVE-2023-36439 | Gaining remote code execution privileges |
| DNS | Windows DNS server | CVE-2021-26877 | Leading to remote code execution |
| Security device | IBM security QRadar EDR | CVE-2024-45100 | Resulting in denial of service |

• OA server. A comprehensive software platform offering services to support the organization's daily office operations.

• File server. Designed to store and manage various internal enterprise files, which provides file-sharing capabilities and access control over permissions.

• Mail server. Responsible for managing and transmitting internal enterprise emails, including functionalities such as email storage, sending, receiving, and filtering.

• DNS server. Resolves domain names to IP addresses, enabling users to access the enterprise intranet as needed.

• Security device server. Records endpoint behaviors and uses data analysis along with contextual information to detect anomalies and malicious activities, while logging data related to such threats.

The specific service and the vulnerabilities presented in each service are detailed in Table 2.

### 5.2 Numerical quantification

In network attack-defense confrontations, the attacker's behavior is driven by the rewards for successful intrusions and the losses from failures.

The reward for successful intrusion represents the potential gain for an attacker after successfully breaching a target service. It reflects the level of interest an attacker has in a particular service. Objectively evaluating the rewards for intrusion is crucial for better protecting the target network. From the attacker's perspective, the goals of successfully intruding into a target service can be classified as follows.

• Stealing sensitive information or data for identity theft or selling on the dark web.

**Table 3** CIA scoring details.

| Attribute | Level | Score | Attribute | Level | Score | Attribute | Level | Score |
|---|---|---|---|---|---|---|---|---|
| Confidentiality | Highest | 25 | Integrity | Highest | 25 | Availability | Highest | 25 |
| | Higher | 20 | | Higher | 20 | | Higher | 20 |
| | Middle | 15 | | Middle | 15 | | Middle | 15 |
| | Lower | 10 | | Lower | 10 | | Lower | 10 |
| | Lowest | 5 | | Lowest | 5 | | Lowest | 5 |

**Table 4** Attack cost details.

| Metric | Metric value | Numerical value |
|---|---|---|
| Attack vector (AV) | Network (N) | 0.20 |
| | Adjacent network (A) | 0.55 |
| | Local (L) | 0.62 |
| | Physical (P) | 0.85 |
| Attack complexity (AC) | Low (L) | 0.44 |
| | High (H) | 0.77 |
| Privilege required (PR) | None (N) | 0.5 |
| | Low (L) | 0.68 |
| | High (H) | 0.85 |
| User interaction (UI) | None (N) | 0.62 |
| | Required (R) | 0.85 |

• Tampering with data to gain economic benefits, cover up attack traces, inject malicious code, or establish backdoors for long-term control.

• Disrupting critical services to extort ransom from the target organization or weakening the organization's competitiveness or reputation by disrupting business continuity.

From the attacker's viewpoint, the CIA triad is used to quantify the rewards of a successful intrusion of each service. The CIA triad is a foundational model in the field of information security, describing the core objectives of information security. It defines the security attributes and potential threats of an information system based on three dimensions.

• Confidentiality. The protection of sensitive information from unauthorized access.

• Integrity. The assurance that the data are accurate and untampered.

• Availability. The guarantee that systems and services are accessible and functional.

By quantifying intrusion rewards based on the CIA triad, it is possible to comprehensively reflect the attractiveness of each service to attackers. Specifically, each attribute is divided into five levels, with specific values shown in Table 3.

Attack cost is used to quantify the cost of failure, which refers to the loss an attacker incurs when attempting to compromise a target service but fails to exploit the vulnerabilities. To comprehensively evaluate attack costs, the common vulnerability scoring system (CVSS) [42] base score is referenced. The base metrics of CVSS include the exploitability submetric, which consists of attack vector (AV), attack complexity (AC), privileges required (PR), and user interaction (UI). However, it is important to note that CVSS is designed to measure the severity of vulnerabilities. A higher CVSS score only indicates that the vulnerability is more severe, but it does not necessarily mean that the exploitation is more difficult. Therefore, only specific attributes and calculation methods from CVSS were referenced in the evaluation. Adjustments were made to the specific values to better fit the context of the attack cost assessment. Detailed scoring adjustments are shown in Table 4. The calculation formula for the cost of failure is given below:

$$\text{cost} = 82.2 \times \text{AV} \times \text{AC} \times \text{PR} \times \text{UI}. \tag{6}$$

Finally, the calculation results of the reward for success and the cost for failure for each service are shown in Tables 5 and 6.

In the experiment, the deception resources were categorized into three types: high, medium, and low interaction, which are based on their effectiveness of deception and deployment costs. High-interaction provides complex interactions that simulate real service behaviors. It has the highest attractiveness to attackers but also incurs the highest deployment cost. Medium-interaction offers partial functional interactions, with moderate attractiveness and deployment costs. Low-interaction only provides static responses, resulting in the lowest attractiveness but

**Table 5** Reward for success for each service.

| Service | Confidentiality | Integrity | Availability | Reward for success |
|---|---|---|---|---|
| Database | Highest | Highest | Higher | 70 |
| OA | Higher | Higher | Middle | 55 |
| File | Higher | Middle | Middle | 50 |
| Mail | Higher | Higher | Higher | 60 |
| DNS | Lower | Highest | Highest | 60 |
| Security device | Highest | Highest | Higher | 70 |

**Table 6** Cost for failure for each vulnerability.

| CVE ID | AV | AV | PR | UI | Cost of failure |
|---|---|---|---|---|---|
| CVE-2024-21176 | N | H | L | N | 5.33695 |
| CVE-2023-1773 | N | L | L | N | 3.049686 |
| CVE-2023-28432 | N | L | N | N | 2.242416 |
| CVE-2023-36439 | A | L | L | N | 8.386636 |
| CVE-2021-26877 | N | L | N | N | 2.242416 |
| CVE-2024-45100 | N | L | H | N | 3.812107 |

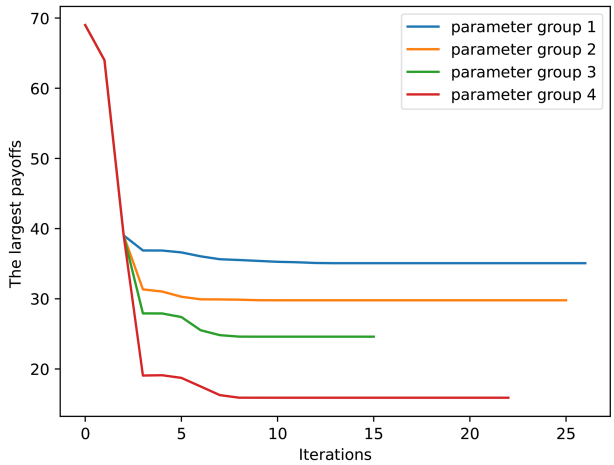


**Figure 3** Iterative optimization process with 6 services.

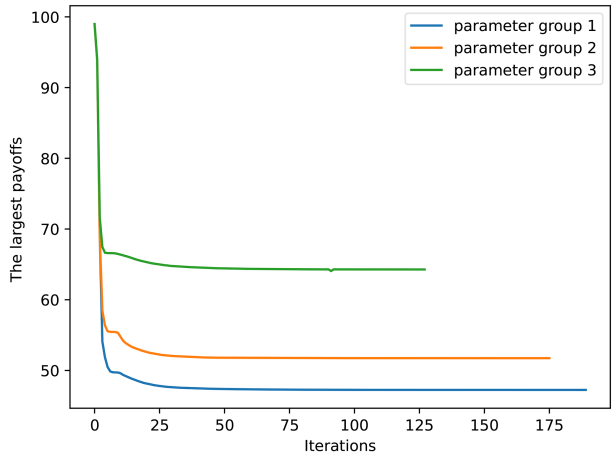


**Figure 4** Iterative optimization process with 500 services.

also extremely low deployment costs. The attractiveness to attackers $\mathbf{B}$ for each type was set as 0.7, 0.5, and 0.2, respectively.

### 5.3 Results and analysis

Figures 3 and 4 show the maximum loss of all services. Figure 3 shows the experimental results of the above 6 services with parameters {1,2,4}, {1,3,5}, {2,3,5}, {3,4,5}. Figure 4 corresponds to a large-scale network scenario containing 500 services. The reward and cost of each service are randomly generated, with values ranging from $[80, 100]$ and $[20, 40]$, respectively. In the three experiments in Figure 4, the number of deceptive resources is set to {100, 200, 300}, {200, 300, 400}, and {300, 350, 400}, respectively. Since the number of services in Figure 3 is small, $\epsilon$ is set to 1e−20 with higher accuracy, and $\epsilon$ in Figure 3 is set to 1e−6. The results show that the optimization process can converge quickly. The more deception resources there are, the smaller the maximum loss of all services will be.

Tables 7–9 show the probabilities of deploying different deception resources for each service when the numbers of high, medium, and low interaction deception resources are set to {1,0,0}, {0,0,1}, and {1,0,1}, respectively.

From the results of the first two sets of experiments, it can be observed that services with high rewards and low costs are assigned relatively higher protection weights. In the third set of experiments, the number of each type of deception resource is the sum of the first two sets. Comparing the results of the three experiments, it is evident that the experimental results with parameters {1,0,1} are not simply the additive result of the experiments with parameters {1,0,0} and {0,0,1}, where each protects the system individually. This shows that even though different

**Table 7** Experimental results with parameter {1,0,0}.

| Service | High | Middle | Lower |
|---|---|---|---|
| Database | 0.3210 | 0 | 0 |
| OA | 0.0475 | 0 | 0 |
| File | 0 | 0 | 0 |
| Mail | 0.1448 | 0 | 0 |
| DNS | 0.1591 | 0 | 0 |
| Security device | 0.3277 | 0 | 0 |

**Table 8** Experimental results with parameter {0,0,1}.

| Service | High | Middle | Lower |
|---|---|---|---|
| Database | 0 | 0 | 0.4949 |
| OA | 0 | 0 | 0 |
| File | 0 | 0 | 0 |
| Mail | 0 | 0 | 0 |
| DNS | 0 | 0 | 0 |
| Security device | 0 | 0 | 0.5051 |

**Table 9** Experimental results with parameter {1,0,1}.

| Service | High | Middle | Lower |
|---|---|---|---|
| Database | 0.3715 | 0 | 0 |
| OA | 0.0293 | 0 | 0.2989 |
| File | 0 | 0 | 0 |
| Mail | 0 | 0 | 0.7011 |
| DNS | 0.2201 | 0 | 0 |
| Security device | 0.3791 | 0 | 0 |

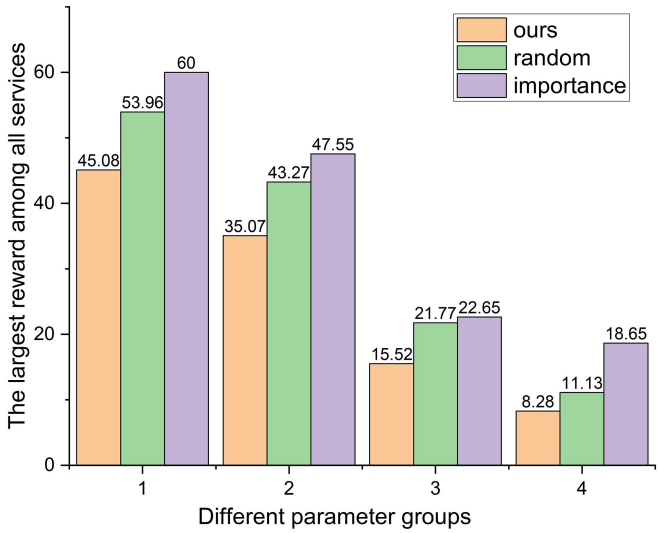


**Figure 5** Comparison chart of maximum loss for three different methods.

deception resources operate independently, they work together to protect the system and collectively enhance the overall security of the system.

Figure 5 compares the experimental results of CDSG with the random method and the maximum payoff-based method under different parameter settings. The $y$-axis represents the attacker rewards the largest value among all services. The random method assigns equal probabilities for all deception resources to protect each service. The maximum payoff-based method dynamically deploys deception resources to protect the service with the current highest attacker payoff at each step. Four sets of experiments were conducted with the numbers of high, medium, and low interaction deception resources set to {1,1,1}, {1,2,4}, {3,4,6}, and {5,5,5}, respectively.

It is evident that, in each parameter setting, CDSG outperforms all other methods in terms of protection effec-

tiveness. The value of the service with the highest attacker payoff among all services is consistently the smallest when using CDSG, and the fewer the resources, the more significant the effect. The reason is that CDSG takes the rational behavior of attackers into account and optimizes defense strategies to minimize the potential gains of the attacker. It demonstrates that, in resource-constrained scenarios, even in the worst-case scenario (the attacker chooses the service with the highest revenue to attack), the attacker's gains under CDSG are significantly lower than those of the other two methods. This means that the method can effectively deal with APTs and improve the risk resistance of the overall system.

## 6 Conclusion

In view of the strong concealment, long duration, and evolving attack methods of APT attacks, as well as the constraints of limited deception resources available to the defender, this paper proposes an optimal dynamic deception defense strategy based on the Stackelberg game. It is a more conservative strategy when defending against APT attacks, aiming to minimize the maximum potential loss across all services by solving the Nash equilibrium of the Stackelberg game. The effectiveness of this method is validated through the comparative experiment with other approaches. Additionally, the experiments demonstrated that CDSG improves system protection by enabling effective cooperation between various types of deception resources.

In future work, efforts will be made to extract traces of attackers from protected systems, combining threat intelligence, using AI technology to infer the attacker's attack intentions, and realizing adaptive adjustment of deception resource strategies based on the attacker's intentions. At the same time, we only considered rational attackers. In practice, attackers' ability could be limited by technology, information, etc., and thus, they may make bounded rational decisions.

**Acknowledgements** This work was supported in part by Guangdong S&T Program (Grant No. 2024B0101010002), National Natural Science Foundation of China (Grant Nos. U2436208, 62372129), Project of Guangdong Key Laboratory of Industrial Control System Security (Grant No. 2024B1212020010), and Major Key Project of Pengcheng Laboratory (Grant No. PCL2024A05).

**References**

1 Yang L X, Li P, Yang X, et al. A risk management approach to defending against the advanced persistent threat. IEEE Trans Dependable Secure Comput, 2018, 17: 1163–1172
2 Nakip M, Gelenbe E. Mirai botnet attack detection with auto-associative dense random neural network. In: Proceedings of IEEE Global Communications Conference (GLOBECOM), 2021. 1–6
3 Algarni S. Cybersecurity attacks: analysis of "Wannacry" attack and proposing methods for reducing or preventing such attacks in future. In: Proceedings of ICT Systems and Sustainability, 2021. 763–770
4 Ren Y, Xiao Y, Zhou Y, et al. CSKG4APT: a cybersecurity knowledge graph for advanced persistent threat organization attribution. IEEE Trans Knowl Data Eng, 2023, 35: 5695–5709
5 Wang C, Lu Z. Cyber deception: overview and the road ahead. IEEE Secur Privacy, 2018, 16: 80–85
6 Wang R, Yang C, Deng X, et al. Development of deception defense technology and exploration of its large language model applications. J Comput Res Dev, 2024, 61: 1230–1249
7 Wang R, Yang C, Deng X, et al. Turn the tables: proactive deception defense decision-making based on Bayesian attack graphs and Stackelberg games. Neurocomputing, 2025, 638: 130139
8 Tian Z, Fang B, Liao Q, et al. Cybersecurity assurance system in the new era and development suggestions thereof: from self-defense to guard. Strategic Study CAE, 2023, 25: 96
9 Liu H, Zhou Y, Fang B, et al. PHCG: PLC honeypoint communication generator for industrial IoT. IEEE Trans Mobile Comput, 2025, 24: 198–209
10 Xu Y, Li M, Fang B, et al. Neural honeypoint: an active defense framework against model inversion attacks. IEEE Trans Neural Netw Learn Syst, 2025, 36: 16186–16197
11 Wang Z, Zhou Y, Liu H, et al. ThreatInsight: innovating early threat detection through threat-intelligence-driven analysis and attribution. IEEE Trans Knowl Data Eng, 2024, 36: 9388–9402
12 Tan J, Lei C, Zhang H, et al. Optimal strategy selection approach to moving target defense based on Markov robust game. Comput Secur, 2019, 85: 63–76
13 Li X X, Meng M, Hong Y G, et al. A survey of decision making in adversarial games. Sci China Inf Sci, 2024, 67: 141201
14 Yang Y, Wang W, Liu L, et al. AoI optimization in the UAV-aided traffic monitoring network under attack: a Stackelberg game viewpoint. IEEE Trans Intell Transp Syst, 2023, 24: 932–941
15 Wang W, Srivastava G, Lin J C W, et al. Data freshness optimization under CAA in the UAV-aided MECN: a potential game perspective. IEEE Trans Intell Transp Syst, 2023, 24: 12912–12921
16 Feng X, Zheng Z, Cansever D, et al. A signaling game model for moving target defense. In: Proceedings of IEEE Conference on Computer Communications, 2017. 1–9
17 Liu J, Wang Z, Yang J, et al. Deception maze: a Stackelberg game-theoretic defense mechanism for intranet threats. In: Proceedings of IEEE International Conference on Communications, 2021. 1–6
18 Wan Z, Cho J H, Zhu M, et al. Resisting multiple advanced persistent threats via hypergame-theoretic defensive deception. IEEE Trans Netw Serv Manage, 2023, 20: 3816–3830
19 Zhang L, Zhu T, Hussain F K, et al. A game-theoretic method for defending against advanced persistent threats in cyber systems. IEEE Trans Inform Forensic Secur, 2022, 18: 1349–1364
20 Yang L X, Li P, Zhang Y, et al. Effective repair strategy against advanced persistent threat: a differential game approach. IEEE Trans Inform Forensic Secur, 2018, 14: 1713–1728
21 Mi Y, Zhang H, Hu H, et al. Optimal network defense strategy selection method: a stochastic differential game model. Secur Commun Netws, 2021, 2021: 1–16
22 Bi J, He S, Luo F, et al. Defense of advanced persistent threat on Industrial Internet of Things with lateral movement modeling. IEEE Trans Ind Inf, 2023, 19: 9619–9630

23 Gan C, Lin J, Huang D W, et al. Equipment classification based differential game method for advanced persistent threats in Industrial Internet of Things. Expert Syst Appl, 2024, 236: 121255
24 He W, Tan J, Guo Y, et al. A deep reinforcement learning-based deception asset selection algorithm in differential games. IEEE Trans Inform Forensic Secur, 2024, 19: 8353–8368
25 Tan J, Jin H, Hu H, et al. WF-MTD: evolutionary decision method for moving target defense based on Wright-Fisher process. IEEE Trans Dependable Secure Comput, 2022, 20: 4719–4732
26 Jin H, Zhang S, Zhang B, et al. Evolutionary game decision-making method for network attack and defense based on regret minimization algorithm. J King Saud Univ-Comput Inf Sci, 2023, 35: 292–302
27 Liu L, Tang C, Zhang L, et al. A generic approach for network defense strategies generation based on evolutionary game theory. Inf Sci, 2024, 677: 120875
28 Xiao L, Xu D, Xie C, et al. Cloud storage defense against advanced persistent threats: a prospect theoretic study. IEEE J Sel Areas Commun, 2017, 35: 534–544
29 Ge H, Zhao L, Yue D, et al. A game theory based optimal allocation strategy for defense resources of smart grid under cyber-attack. Inf Sci, 2024, 652: 119759
30 Tian W, Du M, Ji X, et al. Honeypot detection strategy against advanced persistent threats in Industrial Internet of Things: a prospect theoretic game. IEEE Int Things J, 2021, 8: 17372–17381
31 Ye D, Zhu T, Shen S, et al. A differentially private game theoretic approach for deceiving cyber adversaries. IEEE Trans Inform Forensic Secur, 2021, 16: 569–584
32 Dong Y, Liu J, Ren J, et al. Attack and defense game with intuitionistic fuzzy payoffs in infrastructure networks. Tsinghua Sci Technol, 2024, 30: 384–401
33 He W, Tan J, Guo Y, et al. Flipit game deception strategy selection method based on deep reinforcement learning. Int J Intell Syst, 2023, 2023: 5560416
34 Ngo H Q, Guo M, Nguyen H. Catch me if you can: effective honeypot placement in dynamic ad attack graphs. In: Proceedings of IEEE Conference on Computer Communications, 2024. 451–460
35 Yoon S, Cho J H, Kim D S, et al. Attack graph-based moving target defense in software-defined networks. IEEE Trans Netw Serv Manage, 2020, 17: 1653–1668
36 Li H R, Guo Y F, Huo S M, et al. Defensive deception framework against reconnaissance attacks in the cloud with deep reinforcement learning. Sci China Inf Sci, 2022, 65: 170305
37 Zhang T, Xu C, Shen J, et al. How to disturb network reconnaissance: a moving target defense approach based on deep reinforcement learning. IEEE Trans Inform Forensic Secur, 2023, 18: 5735–5748
38 Eghtesad T, Vorobeychik Y, Laszka A. Adversarial deep reinforcement learning based adaptive moving target defense. In: Proceedings of the 11th International Conference on Decision and Game Theory for Security, 2020. 58–79
39 Chen J, Lan X, Zhang Q, et al. Defending against APT attacks in cloud computing environments using grouped multiagent deep reinforcement learning. IEEE Int Things J, 2025, 12: 19459–19470
40 Sengupta S, Kambhampati S. Multi-agent reinforcement learning in Bayesian Stackelberg Markov games for adaptive moving target defense. 2020. ArXiv:2007.10457
41 Zhu T, Ye D, Cheng Z, et al. Learning games for defending advanced persistent threats in cyber systems. IEEE Trans Syst Man Cybern Syst, 2023, 53: 2410–2422
42 Schiffman M. Common vulnerability scoring system (CVSS). 2011. https://www.first.org/cvss/v3-0/