Game Theory Approaches for Evaluating the Deception-based Moving Target Defense

Duohe Ma
Institute of Information Engineering, Chinese Academy of Sciences
School of Cyber Security, University of Chinese Academy of Sciences
Beijing, China
maduohe@iie.ac.cn

Zhimin Tang
Institute of Information Engineering, Chinese Academy of Sciences
School of Cyber Security, University of Chinese Academy of Sciences
Beijing, China
tangzhimin@iie.ac.cn

Xiaoyan Sun
Department of Computer Science, California State University, Sacramento
Sacramento, USA
xiaoyan.sun@csus.edu

Lu Guo
TravelSky Technology Limited
Beijing, China
guolu@travelsky.com.cn

Liming Wang
Institute of Information Engineering, Chinese Academy of Sciences
Beijing, China
wangliming@iie.ac.cn

Kai Chen*
Institute of Information Engineering, Chinese Academy of Sciences
School of Cyber Security, University of Chinese Academy of Sciences
Beijing, China
chenkai7274@iie.ac.cn

ABSTRACT

Moving target defense (MTD) is a proactive defensive mechanism proposed to disrupt and disable potential attacks, thus reversing the defender's disadvantages. Cyber deception is a complementary technique that is often used to enhance MTD by utilizing misinformation to deceive and mislead attackers. Deception elements, such as honeypot, honey bait, honey token, breadcrumb, and well-constructed deception scenes, can significantly increase the uncertainties for attackers. Deception-based MTD techniques can change the asymmetry situation between defenders and attackers through affecting the attacker's perception of the system. However, there is still a lack of understanding about the role of cyber deception in MTD, and few research works have evaluated the effectiveness of cyber deception.

In this paper, we propose a concept of deception attack surface to illustrate deception-based moving target defense. Moreover, we propose a quantitative method to measure deception, which includes two core concepts: exposed falseness degree and hidden truth degree. We further formulate a deception game model between an attacker and a defender, in which the defender attempts to protect the entry points on the attack surface by creating or changing a deception attack surface. Furthermore, we provide a detailed example scenario and analyze the deception game's equilibrium. Finally We verify the effectiveness of our proposed method through a real attack and defense experiment.

*Corresponding Author: Kai Chen. Email: chenkai7274@iie.ac.cn

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

MTD '22, November 7, 2022, Los Angeles, CA, USA

© 2022 Association for Computing Machinery.

ACM ISBN 978-1-4503-9878-7/22/11…$15.00
https://doi.org/10.1145/3560828.3563995

CCS CONCEPTS

• **Security and privacy** → *Network security*; • **Computing methodologies** → *Modeling and simulation*;

KEYWORDS

Cyber Deception, Moving Target Defense, Security Evaluation

ACM Reference Format:
Duohe Ma, Zhimin Tang, Xiaoyan Sun, Lu Guo, Liming Wang, and Kai Chen. 2022. Game Theory Approaches for Evaluating the Deception-based Moving Target Defense. In *Proceedings of the 9th ACM Workshop on Moving Target Defense (MTD '22), November 7, 2022, Los Angeles, CA, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3560828.3563995

1 INTRODUCTION

In recent years, increasing security problems in cyberspace present new requirements to the existing security defense mechanisms. The certainty, homogeneity, and static nature of computer systems provide plenty of time and space for attackers to analyze the target systems and implement intrusion. Furthermore, current cyber defenses are mostly reactive as the response comes after attacks have happened. These characteristics provide attackers with significant advantages over defenders. Therefore, innovative defense techniques are needed to break such asymmetric situation.

Moving target defense (MTD) [20] and cyber deception [27] are two main proactive defensive techniques proposed to disrupt and disable potential attacks, thus reversing the defender's disadvantages [31]. The fundamental idea of MTD is to dynamically and randomly alter the attributes of a system, increasing the uncertainty and complexity for attackers. In comparison, cyber deception utilizes plausible-looking and carefully crafted misinformation to deceive and mislead attackers. Researches show that MTD techniques can be enhanced by the addition of deception [10]. MTD and cyber deception are complementary techniques that can be deployed by defenders simultaneously with common objectives to defeat the attackers. Due to its diversified deceptive methods,

as well as the characteristics of low construction cost and easy construction of deceptive attributes, cyberspace deception provides a new direction for expanding the attack surface shifting space, and has become an important technical direction for MTD research.

Approaches combining MTD and deception can be used in various domains (e.g., deception network [30], web deception [12], [35], data deception[36]). However, there are few systematic approaches to evaluate the effectiveness of deception in MTD techniques. Current approaches include evaluating by metrics, mathematical models or simulation methods. They have three shortcomings: 1) it is not clear how to quantify the deception effect in the network; 2) many MTD evaluation methods are applicable to specific technologies and limited to specific application scenarios, so they lack broad applicability; 3) the MTD system has the characteristics of dynamics and interaction complexity that traditional evaluation methods cannot propose a formal specification to describe.

In order to illustrate deception-based moving target defense and evaluate its effectiveness, in this paper we propose a concept of deception attack surface, which is defined as the attack surface observed and perceived by the attackers after the introduction of deception techniques. To formally measure deception in the deception attack surface, we bring in the concepts of exposed falseness degree and hidden truth degree. This enables the quantification of deception measurement. To better understand the role of deception in MTD, we construct the deception game model between the attacker and the defender. In this model, the defender aims to protect the system by creating or changing a deception attack surface. Through the proposed reward function for attackers and defenders, we can measure the payoffs and costs of the changes to the deception attack surface. These changes are usually caused by defenders' deception action or attacker's move. To demonstrate our model, we provide an example scenario in which a defender protects real attributes on attack surface via adding deceptive attributes, and then analyze the deception game's equilibrium. With evaluation, we demonstrate that our method can effectively measure the deceptive degree of the deception-based MTD model, and can successfully model the attack-defense relationship and give a strategy that satisfies the Nash equilibrium.

Our ultimate aim is to measure the degree of deception and evaluate the deception-based model. To achieve this, we use a combination of deception metrics and incomplete information dynamic games. The advantages of our approach are: 1) Metrics can effectively measure the deception effect of the system, and the game method can evaluate deception benefits; 2) Dynamic changes of active defense model can be well reflected in the multi-step decision-making of offensive side and defensive side in signaling deception game; 3) The fusion method of metrics and game method can help defenders to continuously optimize the deployment plan, so as to further improve the deception effect; 4) Our method has applicability and is not limited by scenarios.

The major contribution of this paper is as follows:

- It introduces the concept of deception attack surface and proposes an approach to quantitatively measure deception;
- It applies game model to analyze and understand deception based MTD;
- Using an example scenario, it studies the equilibrium of the game model and demonstrates how to compute the payoffs and costs for different strategies of attackers and defenders.

2 RELATED WORK

A main research methodology of MTD is to first propose a new MTD technique, and then evaluate the effectiveness of the method. In general, existing methods for evaluating the effectiveness of MTD techniques can be divided into three categories: (i) metric-based, (ii) mathematical model-based, and (iii) simulation-based.

Metric-based. Kinds of metrics have been proposed to measure the security of MTD system. Thomas et. al. [7] assessed the performance of network address shuffling in the light of the attacker's success rate. Jafarian et. al. [19] proposed three metrics: deterrence, deception, and detectability. Warren Connell et. al. [11] presented a quantitative analysis model to evaluate the resource availability and performance of MTD via queuing theory. Multiple metrics (productivity, success, confidentiality, and integrity) were analyzed for the attacker and defender [33]. A quantitative evaluation method of MTD based on entropy is proposed in [22]. Alavizadeh et. al. [3] proposed four metrics for the comprehensive security assessment of MTD technology: system risk, attack cost, return on attack, and availability. Taylor et. al. [28] designed a series of metrics to evaluate the costs of mission activities and the benefits in the face of attacks. Security metrics, such as system risk and reliability, were used to evaluate MTD techniques [4]. Tuan et. al. [25] used key performance metrics to comprehensively evaluate complex behaviors of the operating system and MTD strategies. There are few metrics to measure the effect of deception. The possible metrics include the proportion of decoys, the shifting rate of deception strategies, the benefits of deception, etc.

Mathematical model-based. Zhuang et. al. [38] first proposed using Markov model to analyze MTD, in which each state in Markov chain represented the current system configuration, and then [23], [9] have been proposed. Al Amin et. al. [1] used transition probabilities in hidden Markov models to predict paths. One of the problems with Markov model based MTD methods is that the number of states increases exponentially with the increase of system components.

Game theory is also the main method widely used in MTD analysis and modeling. Game Theory can be used to maximize the security of defenders while minimizing the shifting rate of the system, that is, to achieve the greatest benefits at the lowest cost. Manadhata et. al. [24] used the complete and perfect information game to represent the attack surface shifting. Zhu et. al. [37] proposed a game theoretic framework to analyze multi-layer attack surface shifting. Some MTD schemes introduce deception and assess the security of computer system. Feng et. al. [14] demonstrated that the security of MTD can be further improved when combined with information disclosure. Carroll et. al. [8] modeled the interactions between defender and attacker via a signaling game to evaluate the effect of deception. Fang et. al. [13] developed a series of game theory models of network deception and algorithms to calculate the equilibrium in the game. Ye et. al. [32] proposed a new differentially private game theoretic approach to model cyber deception.

Table 1: Comparison between MTD and Cyber Deception

		MTD	Cyber Deception
Similarity		proactive defense aiming at protecting the system.	
Difference	Technical Idea	change the configuration of the system, increasing the complexity and diversity of the system, making the shifting rate of the system faster than the attack rate.	Instead of focusing on changing the configuration of the system, cyber deception hides the real target via proactively disclosing some information
	Information Disclosure	no, prevent the attacker from gathering information.	yes, only leak unnecessary, part of the real information or carefully designed decoy information.
	Cost	high load on the system	low and simple deployment settings, and lower load on the system
	Object	understanding the system	understanding the attacker
Relationship		complementary defensive approaches, can be deployed simultaneously	

Attack graph describes the attack path under a certain network configuration. Hong and Kim [18] proposed a two-layer hierarchical attack representation model (HARM) by incorporating MTD in attack graph security models. Hamlet et. al. [16] stated that the feasibility of MTD resulted from the moving target controls breaking critical system dependencies and increasing the complexity of an attack. Jin et. al. [21] established a multi-dimensional attack graph model to formalize various complex attack scenarios, and combined this model to effectively evaluate and optimize MTD strategy.

Markov model, attack graph, and game theory are main methods of MTD evaluation. How to measure the effects of deception on MTD is a key issue in MTD research. However there are few research methods which have been proposed.

Simulation-based. A typical example of simulation-based methods is [39]. In [39] the authors proposed a preliminary design of a network MTD system based on simulator NeSSi2 and captured the effectiveness of the proposed MTD system depending on the probability of a successful attack. The idea of deception is often combined with MTD to enhance the computer security. Al-Shaer et. al. [2] implemented a proof-of-concept for Random Host Mutation (RHM) in a university campus network. The authors evaluated the RHM effectiveness against scanning external and internal scanners. Border et. al. [6] evaluated the effectiveness of decoy IP addresses to mislead remote network attacks via obfuscating the results at the reconnaissance phase. Gao et. al. [15] used virtual network topology to confuse the target network and evaluate the system. Poschinger et. al. [26] developed a hybrid platform called OpenMTD to evaluate

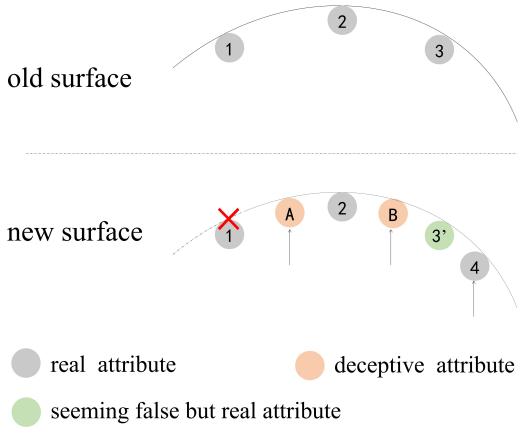

Figure 1: Deception-based MTD. We have two ways to shift the deception attack surface: adding a decoy or disguising the real attributes.

MTD techniques at the network level. Zhang et. al. [34] developed new models to aggregate the attack surface of different network resources as a formal security measure to evaluate the ability of the network to resist zero day attacks. Han et. al. [17] implemented a network deception framework to evaluate the use of deception in network applications through experiments. Torquato et. al. [29] presented a tool for evaluating the effectiveness of time-based MTD against availability attacks. Existing simulation based MTD methods are case-dependent and lack extensive applicability. Although techniques for evaluating MTD already exist, there is a lack of methods for evaluating the deception degree of MTD systems. Therefore, this paper proposes to use the deception attack surface and deception indicators to well quantify the deception of the MTD system. At the same time, the evaluation method based on game theory can effectively model and evaluate the deception-based MTD system.

3 DECEPTION-BASED MOVING TARGET DEFENSE

Moving target defense and cyber deception are complementary proactive defensive approaches, both of which aim at defeating the attackers. The difference is that MTD increases the complexity, diversity, and randomness of a system by constantly changing the attack surface of the system. Cyber deception mainly misleads the attacker's actions by providing seemingly real but false information. Deception defense increases the shifting space of system's attack surface. Deception elements, such as honeypot, honey bait, honey token, breadcrumb, well-designed deception stories, and well-constructed deception scenes, create huge information entropy that forces the attacker into a deceptive environment where it's difficult to distinguish the real target from the deceptive scenarios. Table 1 shows the comparison between MTD and cyber deception.

3.1 Deception Attack Surface

Attack surface measurement is generally used as an indicator of a system's security. A system's attack surface is the subset of the system resources that an attacker can exploit to penetrate the system.

The larger of the attack surface, the more insecure of the target system. Manadhata et. al. [24] first proposed the use of attack surface shifting in the context of moving target defense. The basic idea of MTD is to continuously shift the attack surface of the target system. Albanese et. al. [5] proposed the concept of virtual attack surface, which is the perceived view of the attacker to the system attack surface. The proposed method generates an external view of the system by manipulating outbound traffic. There is a limitation that manipulation is limited to certain types of traffic. In order to expand the shifting space of the attack surface, in this paper we introduce the concept of **deception attack surface**. Deception techniques do not modify the real attack surface of the system, but change the attacker's perception of the system's attack surface. Hence the deception attack surface is defined as the attack surface observed and perceived by the attacker after the introduction of deception techniques. Different from the previous work, our proposed deception attack surface implements deception technology by deploying various types of attributes, and can acquire knowledge by capturing the attacker's behavior. Furthermore, the deception attack surface can be applied to measure the deception degree of MTD system, which also provides a basis for the subsequent attack and defense modeling based on game theory.

Deception-based MTD is a more advanced defense mechanism where the defenders continuously shift the deception attack surface. As shown in Figure. 1, there are two ways to shift the deception attack surface: adding a decoy or disguising the real attributes. IP address, port, routing, protocol, etc. can all constitute the attribute of the attack surface, it may be a flaw or a vulnerability that can be exploited by an attacker.

- Adding a decoy. This technique is mainly used to draw the attacker's attention away from the critical resource of a system. Decoys require the system to invent a number of carefully-prepared decoys with the goal of making the attacker believe that they are real. Honeypot is a typical example to deceive the attacker.
- Disguising a real attribute. We can disguise a real attribute as a deceptive one through several ways. For example, we can deceive the attacker via making the system mimic the response of a different version. In addition, we can hide the service by responding as if it's not working. Or we can respond that all services are open so that the attacker can't determine which service is running.

In Figure. 1, the defender inserts two decoy attributes, A and B, to distract the attacker's attention from the real target. Moreover, the real attribute 3 is disguised as a deceptive attribute 3', thus reducing the attacker's attention.

3.2 Deception Measurement

To measure deception in the deception attack surface, we propose the key concepts of exposed falseness degree and hidden truth degree to quantify deception.

Exposed Falseness Degree. Exposed falseness degree indicates the similarity between real attribute and deceptive attribute. Based on indistinguishable confusion theory, in this paper we introduce the concept of indistinguishable deception to measure the exposed

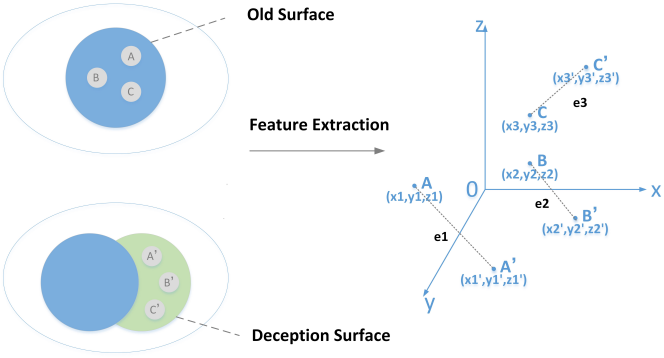

Figure 2: Exposed Falseness Degree. Assuming n = 3, it is expressed through feature extraction and deception distance.

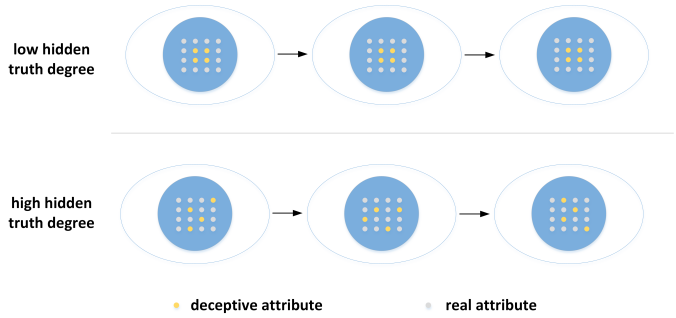

Figure 3: Hidden Truth Degree. The upper model has low hidden truth degree and the lower model has high hidden truth degree.

falseness degree. Indistinguishability deception indicates the indistinguishability of each deceptive attribute in MTD model, which is measured by deception distance. As shown in Figure. 2, the core idea of exposed falseness degree calculation is to map the deceptive attribute and original attribute to the feature space through feature extraction, and use the deception distance to represent the exposed falseness degree. According to the features and effects of each deceptive attribute, we extract the n-dimensional feature of the attribute through the mapping function, form a n-dimensional feature vector and establish the corresponding feature space. Then, we use the Euclidean distance to measure the deception distance of a single attribute, and calculate the deception distance of all attributes according to the weight value of each attribute.

Assuming that the feature dimension is 3, the features of an attribute can be extracted from three dimensions: the function, interaction and configuration of the attribute. The features of each dimension have a number of components. Each feature component includes two classification states of "1" and "0" ("1" means the feature component exists, "0" means the feature component does not exist). The feature value of each dimension is obtained by adding the binary feature components up, and then the value is normalized. The feature value of each dimension constitutes the three-dimensional feature vector of the attribute. The similarity between the attributes is obtained by calculating the distance between the feature vectors of the real attribute and the deceptive attribute, which is used to represent the exposed falseness degree. Taking the FTP service as

an example attribute, the feature component "listener" belongs to the function dimension, which indicates whether to listen to the file transfer status. The feature component "write" belongs to the interaction dimension, which indicates whether to enable the write permission. The feature component "anonymous_enable" belongs to the configuration dimension, which indicates whether anonymous users are allowed to access. The features of each dimension can have many feature components. The dimension of features can also be expanded.

Our deception-based MTD model is denoted as S. We assume that the attribute is $x = \{x_i\}_{i \in n}$, where x_i is the i-th feature and n is the number of features. Similarly, the corresponding deceptive attribute is $y = \{y_i\}_{i \in n}$, where y_i is the i-th deception feature and n is the number of deception features. So we define the exposed falseness degree of a single attribute, e_s, as follows:

$$e_S = \Delta(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

We use multi-attribute weighting theory to calculate the deception distance for all attributes. We assume the set of attributes for the model is $X = \{X_i\}_{i \in N}$, where X_i is the i-th attribute and N is the number of attributes. Similarly, the set of deceptive attributes for the model is $Y = \{Y_i\}_{i \in N}$, where Y_i is the i-th deceptive attribute and N is the number of deceptive attributes. The weight of attribute is expressed as $\omega = \{\omega_i\}_{i \in N}$. The sum of weights for all attributes is 1. So the exposed falseness degree of the whole model E_s is:

$$E_S = \Delta(X, Y) = \sum_{i=1}^{N} \omega_i \Delta(X_i, Y_i) = \sum_{i=1}^{N} \omega_i e_S^i$$

Hidden Truth Degree. Entropy is an important measure of MTD redundancy and diversity. In order to measure the hidden degree of real attributes in the MTD model, we propose the concept of hidden truth degree. We use the entropy in information theory to measure the size of hidden truth degree, which is used to evaluate the real scale of the whole system. Authenticity measures the proportion of real modules in the system, and hidden truth degree measures the hidden degree of the real attributes of the system. As shown in Figure. 3, yellow dots and grey dots represent deceptive attributes and real attributes respectively. In the upper model, the distribution of real attributes and deceptive attributes remain unchanged. Hence the model has a low hidden truth degree. It is easy for attackers to find out whether the attributes are deceptive. The lower model, however, changes the distribution of deceptive attributes frequently and thus has a high hidden truth degree. It is difficult for attackers to find out which attributes are deceptive.

Our deception-based MTD model is still denoted as S. The attribute of the model is expressed as $X = \{X_i\}_{i \in N}$, and its weight is $\omega = \{\omega_i\}_{i \in N}$, where X_i is the i-th attribute, ω_i is the weight of the i-th attribute, and N is the number of attributes. The sum of weights for all attributes is 1. The deceptive attribute of the model can be described by vector $v = \{v_i\}_{i \in N}$, where $v_i \in \{0, 1\}, 1 \leq i \leq N$ indicates whether the corresponding attribute in S has been altered. We say the attribute X_i is in a deceptive state if $v_i = 1$. So, the authenticity of model S is

$$H_S = \frac{\sum_{v_i=0} \omega_i}{\sum_{X_j \in S} \omega_j} = \sum_{v_i=0} \omega_i, 1 \leq i, j \leq N$$

Based on the information entropy theory, the possibility of deceptive attribute is uncertain, and it can be measured according to its probability. If the probability is large, the uncertainty is small; otherwise, the uncertainty is large. p represents the probability of occurrence of the attribute. The hidden truth degree of model S is

$$I_S = -\sum_{i=1}^{N} \omega_i p(v_i X_i) \log p(v_i X_i)$$

3.3 Game Model

In this section, we formulate the MTD game model between two players in which the defender tries to protect the system through shifting the attack surface or creating the deception attack surface. The attacker will attempt to attack and compromise the target. The attacker can successfully compromise the normal attributes, but not deceptive attributes. If the attacker attacks a honey attribute, the system can observe the attacker's actions and tracks, and further improve the defenses. We model the interaction between a defender and an attacker as follows.

Our stochastic game model is defined as a three tuple $< N, A, R >$, where,

- $\mathcal{N} = \{\mathcal{A}, \mathcal{D}\}$ is the set of players, where player $\mathcal{A}$ is the attacker and player $\mathcal{D}$ represents the defender. Nature chooses the type of each attribute on the attack surface. With probability α, a normal attribute is disguised as deceptive. With probability γ, the system adds a decoy and disguises it as real. And otherwise the system shifts the attribute among a series of real variants with probability $1 - \alpha - \gamma$.
- $\mathcal{A} = \mathcal{A}^a \times \mathcal{A}^d$ is the game action space, where $\mathcal{A}^a$ and $\mathcal{A}^d$ is the set of actions for attackers and defenders respectively.
  - $\mathcal{A}^d = \mathcal{A}_d^d \cup \mathcal{A}_n^d \cup \mathcal{A}_r^d$, where $\mathcal{A}_d^d$ represents set of actions for adding decoys, $\mathcal{A}_n^d$ denotes set of actions for disguising a normal attribute as honey, $\mathcal{A}_r^d$ represents a set of real variants for a normal attribute.
  - $\mathcal{A}^a = \mathcal{A}_a^a \cup \mathcal{A}_t^a \cup \mathcal{A}_r^d$ indicates that an attacker will either attack the attribute without determining the attribute type ($\mathcal{A}_a^a$), determine the attack based on prior test on a attribute type ($\mathcal{A}_t^a$), or retreat ($\mathcal{A}_r^a$).
- $\mathcal{R} = \{\mathcal{R}^d, \mathcal{R}^a\}$ represents the defender's and attacker's reward function respectively.

Reward Function. When a defender adopts deception based moving target defense techniques, the defender may benefit from two ways. First, the defender may mitigate the system's risk by shifting the attack surface to deception attack surface. The defender may enable a deceptive attribute to deceive the attacker's perception. Second, the defender may shift the attack surface among series of real attribute variants. However, the defender's deception action may be identified by the attacker and used as a springboard to attack the real target, thus increasing the attack surface measurement. Hence the defender's reward function depends on the change on the attack surface, the change on the attack surface measurement, and the cost of shifting attack surface. Similarly, when an attacker

takes a move, the attacker may benefit from the increase in the attack surface measurement. The attacker will cost more on a successful attack because of the shift of the attack surface. Hence the attacker's reward function depends on the change on the attack surface measurement and the cost on launching an attack.

If the defender performs a deception action, a^d, and the attacker takes a move, a^a, then we denote the change on the deception attack surface by enabling a honey attribute as ΔHA, the change on the deception attack surface by disguising a real attribute as ΔRA, and the change in the attack surface measurement as ΔASM. So we define the defender's reward function, $\mathcal{R}^d$, and the attacker's reward function, $\mathcal{R}^a$, as follows:

$$\begin{aligned}\mathcal{R}^d(a^a, a^d) &= \Phi_1(\Delta HA) + \Phi_2(\Delta RA) \\ &\quad - \Theta_1(\Delta ASM) - \Theta_2(Cost_1) \\ \mathcal{R}^a(a^a, a^d) &= \Phi_3(\Delta ASM) - \Theta_3(Cost2)\end{aligned}$$

Φ_is and Θ_is are mapping functions that map the changes on attack surface and deception attack surface, attack surface measurement, and cost to real numbers. The numbers reflect the payoffs and costs related with changes. Game theoretic approaches to attack surface shifting of non-deceptive MTD are shown in [24].

4 EXAMPLE SCENARIO

A successful attack needs to find an entry point in the attack surface and determine the attack path from the entry point to the target. How to ensure the entry points' (attributes') security on the attack surface is a critical part. Deception-based moving target defense techniques enhance security protection and further change the information asymmetry between attackers and defenders.

4.1 Deception Game

We consider that a defender protects attributes on the attack surface via randomly adding deceptive attributes. The attacker wins if he compromises a real attribute, not deceptive attributes. We propose to use a Stackelberg game model where the defender as the leader determining the deceptive attributes' placement to constitute a deception attack surface with the original real attributes, the attacker as the follower attempting to compromise an attribute on the attack surface. Attackers can attack multiple attributes at once, or attack different attributes continuously. The benefits and costs of both sides in each round of the game are accumulated by the results of multiple attributes. The system chooses either type deceptive attribute (D) or normal attribute (N) with probability P_d and $1 - P_d$, respectively. The attacker will randomly select an attribute to compromise. After identifying an attribute on the attack surface, the attacker will either attack the attribute without determining the attribute type (A), determining the attack based on prior test on the attribute type (T), or retreat (R). In the next round of the game, the defender and the attacker will choose strategies and actions according to the results of the previous round of the game. The main notation used throughout the paper is summarized in Table 2.

We assume that the cost of placing deceptive attributes is c_d. If one of the real attribute is compromised, the defender suffers a loss of c_d^l, which includes the financial loss and the cost of restoring the attribute. If the attribute that the attacker compromised is deceptive, the defender gains g_d^o by observing and learning the attacker's actions. If the attacker retreats before attacking the attribute, the defender gains g_d^r. When the attacker attacks an attribute, it incurs cost c_a, regardless of whether the attack succeeds or fails. The attacker gains g_a when he compromises a real attribute, and the profit is $g_a - c_a$. The attacker will lose c_a^l when he attacks a honey attribute. In addition, the attacker may test the attribute's type before launching an attack. The prior tests to determine the attribute type cost c_t. After obtaining the tests results, the attacker will either perform his attack action or abandon the attack. If the attacker tests for a normal attribute, his payoff is $g_a - c_a - c_t$. Otherwise if he tests for a honey attribute, he loses c_t.

Table 2: Main notations used throughout the paper

Notation	Meaning
D	The attribute is deceptive
R	The attribute is real
P_d	Probability that the attribute is deceptive
$1 - P_d$	Probability that the attribute is normal
c_d	Cost of placing and disguising a deceptive attribute
c_d^l	Loss of being compromised a normal attribute
g_d^o	Benefit of observing an attack on a deceptive attribute
g_d^r	Benefit gotten because the attacker retreats
A	Attack the attribute without determining the attribute type
T	Determining the attack based on prior tests on the attribute type
R	Retreat, not attack
c_a	Cost of attacking an attribute
g_a	Benefit of compromising a normal attribute
c_a^l	Loss of attacking a deceptive attribute
c_t	Cost of prior test for an attribute
h	signal that an attribute is deceptive
n	signal that an attribute is normal

The benefit and cost of the offensive and defensive sides depend on multiple indicators and historical statistics. The defense benefit(g_d^o) is measured by the quantity of attack information captured by the honeypot. The defense benefit(g_d^r) is measured by the actual value of the attribute. The defense cost(c_d) is represented by software or hardware resources, funds and time used to deploy deceptive attributes. The defense cost(c_d^l) is represented by same indicators for recovery. The attack benefit(g_a) are determined by the attack target, including the utilization of resources, the occupation of network bandwidth and the destruction of the system. The attack cost(c_a or c_a^l) is represented by the attack time, attack domain knowledge and consumed software or hardware resources. The test cost(c_t) is represented by time, knowledge and resources for detection. The weights between indicators can be calculated by principal component analysis or regression analysis.

4.2 Signaling Deception Game

The deception game set above can be further improved by introducing a deception signal. The defender can signal that the attribute is deceptive (h) or normal (n), regardless of the attribute's actual type.

Table 3: Attacker's nine pure strategies

signal	$w1$	$w2$	$w3$	$w4$	$w5$	$w6$	$w7$	$w8$	$w9$
n	A	A	A	T	T	T	R	R	R
h	A	T	R	A	T	R	A	T	R

After receiving the signal, the attacker will choose action A, action T, or action R.

The defender can adopt four pure strategies: 1) the defender can signal the attribute's type is normal (Strategy $s1$) or honey (Strategy $s4$), independent of the system's actual type. 2) the defender signals the attribute is normal or honey that is in accordance with the attribute's actual type (Strategy $s2$). 3) the defender can signal the opposite of the attribute's actual type, that is, the defender signals that the attribute is honey, while the attribute is normal and signals the attribute is normal if the attribute is honey (Strategy $s3$). The attacker can adopt nine pure strategies, as in Table 3, indicating that the attacker can perform different actions for each signal received from the defender: the attacker can choose action A (T, R) if he receives signal n and choose action T (A, R) if he receives signal h.

4.3 Signaling Deception Game's Equilibrium

In this section, we will investigate the existence of Perfect Bayesian Equilibria (PBE) for the deception game. Final equilibrium should involve from Strategy $s1$ to Strategy $s4$.

If the defender adopts Strategy $s1$ (the defender will signal that the attribute is normal irrespective of the attribute's actual type), and the attacker receives the signal n, he will choose action A while the expected payoff induced by this action is greater than the expected payoff of action T and action R.

$$\begin{cases} P_d \cdot (-c_a - c_a^l) + (1 - P_d) \cdot (g_a - c_a) \geq P_d \cdot (-c_t) \\ \qquad\qquad\qquad\qquad +(1 - P_d) \cdot (g_a - c_a - c_t) \\ P_d \cdot (-c_a - c_a^l) + (1 - P_d) \cdot (g_a - c_a) \geq 0 \end{cases}$$

which satisfies,

$$P_d \leq \frac{c_t}{c_a + c_a^l} \tag{1}$$

$$P_d \leq \frac{g_a - c_a}{g_a + c_a^l} \tag{2}$$

Next, we further analyze the interaction of off-equilibrium path if the defender sends signal h. In this case, we will analyze the attacker's strategy $w1$, $w2$, and $w3$. We assume that the attacker's belief at the signal that the attribute is honey or normal is expressed by q and $1 - q$ $(0 \leq q \leq 1)$. For strategy $w3$, the defender's optimal strategy is $s3$, not $s1$, so we may analyze this equilibrium in subsequent section. The defender will change his deception strategy from sending signal n to signal h if the payoff $g_d^r > c_d^l$. Therefore if the attacker adopts strategy $w1$, the attacker's expected payoff of action A should be greater than the expected payoff of action T and action R, so we have,

$$\begin{cases} q \cdot (-c_a - c_a^l) + (1 - q) \cdot (g_a - c_a) \geq q \cdot (-c_t) \\ \qquad\qquad\qquad\qquad +(1 - q) \cdot (g_a - c_a - c_t) \\ q \cdot (-c_a - c_a^l) + (1 - q) \cdot (g_a - c_a) \geq 0 \end{cases}$$

which satisfies,

$$q \leq \frac{c_t}{c_a + c_a^l} \tag{3}$$

$$q \leq \frac{g_a - c_a}{g_a + c_a^l} \tag{4}$$

Similarly if the attacker adopts strategy $w2$, the attacker's expected payoff of action T should be greater than the expected payoff of action A and action R, so we have,

$$\begin{cases} q \cdot (-c_t) + (1 - q) \cdot (g_a - c_a - c_t) \geq q \cdot (-c_a - c_a^l) \\ \qquad\qquad\qquad\qquad +(1 - q) \cdot (g_a - c_a) \\ q \cdot (-c_t) + (1 - q) \cdot (g_a - c_a - c_t) \geq 0 \end{cases}$$

which satisfies,

$$q \geq \frac{c_t}{c_a + c_a^l} \tag{5}$$

$$q \leq 1 - \frac{c_t}{g_a - c_a} \tag{6}$$

If the defender adopts Strategy $s1$ (the defender will signal that the attribute is normal irrespective of the attribute's actual type), and the attacker receives the signal n, he will choose action T while the expected payoff induced by this action is greater than the expected payoff of action A and action R.

$$\begin{cases} P_d \cdot (-c_t) + (1 - P_d) \cdot (g_a - c_a - c_t) \geq P_d \cdot (-c_a - c_a^l) \\ \qquad\qquad\qquad\qquad +(1 - P_d) \cdot (g_a - c_a) \\ P_d \cdot (-c_t) + (1 - P_d) \cdot (g_a - c_a - c_t) \geq 0 \end{cases}$$

which satisfies,

$$P_d \geq \frac{c_t}{c_a + c_a^l} \tag{7}$$

$$P_d \leq 1 - \frac{c_t}{g_a - c_a} \tag{8}$$

In this case, we will analyze the attacker's strategy $w4$, $w5$, and $w6$. If the attacker adopts strategy $w4$, the defender's optimal action will be $s2$ because his payoff is greater than the payoff $-c_d$ when he adopts strategy $s1$. Moreover, if the attacker adopts strategy $w6$, the defender's optimal action will be $s3$. The attacker's strategy $s5$ leads to the equilibrium that satisfies equation (5) and (6).

If the defender adopts Strategy $s1$ (the defender will signal that the attribute is normal irrespective of the attribute's actual type), and the attacker receives the signal n, he will choose action R while the expected payoff induced by this action is greater than the expected payoff of action A and action T.

$$\begin{cases} 0 \geq P_d \cdot (-c_a - c_a^l) + (1 - P_d) \cdot (g_a - c_a) \\ 0 \geq P_d \cdot (-c_t) + (1 - P_d) \cdot (g_a - c_a - c_t) \end{cases}$$

which satisfies,

$$P_d \geq \frac{g_a - c_a}{g_a + c_a^l} \tag{9}$$

$$P_d \geq 1 - \frac{c_t}{g_a - c_a} \tag{10}$$

Under this case, we will analyze the attacker's strategy $w7$, $w8$, and $w9$. If the attacker adopts strategy $w7$, the defender's optimal action will be $s2$. And the attacker's strategy $w8$ leads to the equilibrium that satisfies equation (5) and (6). Furthermore, if the attacker adopts

Table 4: The optimal strategies and corresponding conditions if $c_t = c_t^h = c_t^n$

Game's Equilibrium Defender strategy	Attacker strategy	On-equilibrium path condition	Off-equilibrium path condition
$s1$	$w1$	$P_d \le \frac{c_t}{c_a + c_a^l}, P_d \le \frac{g_a - c_a}{g_a + c_a^l}$	$q \le \frac{c_t}{c_a + c_a^l}, q \le \frac{g_a - c_a}{g_a + c_a^l}$
	$w2$	$P_d \le \frac{c_t}{c_a + c_a^l}, P_d \le \frac{g_a - c_a}{g_a + c_a^l}$	$q \ge \frac{c_t}{c_a + c_a^l}, q \le 1 - \frac{c_t}{g_a - c_a}$
	$w5$	$P_d \ge \frac{c_t}{c_a + c_a^l}, P_d \le 1 - \frac{c_t}{g_a - c_a}$	$q \ge \frac{c_t}{c_a + c_a^l}, q \le 1 - \frac{c_t}{g_a - c_a}$
	$w8$	$P_d \ge \frac{g_a - c_a}{g_a + c_a^l}, P_d \ge 1 - \frac{c_t}{g_a - c_a}$	$q \ge \frac{c_t}{c_a + c_a^l}, q \le 1 - \frac{c_t}{g_a - c_a}$
	$w9$	$P_d \ge \frac{g_a - c_a}{g_a + c_a^l}, P_d \ge 1 - \frac{c_t}{g_a - c_a}$	$q \ge \frac{g_a - c_a}{g_a + c_a^l}, q \ge 1 - \frac{c_t}{g_a - c_a}$
$s4$	$w1$	$P_d \le \frac{c_t}{c_a + c_a^l}, P_d \le \frac{g_a - c_a}{g_a + c_a^l}$	$q \le \frac{c_t}{c_a + c_a^l}, q \le \frac{g_a - c_a}{g_a + c_a^l}$
	$w4$	$P_d \le \frac{c_t}{c_a + c_a^l}, P_d \le \frac{g_a - c_a}{g_a + c_a^l}$	$q \ge \frac{c_t}{c_a + c_a^l}, q \le 1 - \frac{c_t}{g_a - c_a}$
	$w5$	$P_d \ge \frac{c_t}{c_a + c_a^l}, P_d \le 1 - \frac{c_t}{g_a - c_a}$	$q \ge \frac{c_t}{c_a + c_a^l}, q \le 1 - \frac{c_t}{g_a - c_a}$
	$w6$	$P_d \ge \frac{g_a - c_a}{g_a + c_a^l}, P_d \ge 1 - \frac{c_t}{g_a - c_a}$	$q \ge \frac{c_t}{c_a + c_a^l}, q \le 1 - \frac{c_t}{g_a - c_a}$
	$w9$	$P_d \ge \frac{g_a - c_a}{g_a + c_a^l}, P_d \ge 1 - \frac{c_t}{g_a - c_a}$	$q \ge \frac{g_a - c_a}{g_a + c_a^l}, q \ge 1 - \frac{c_t}{g_a - c_a}$

Table 5: The optimal strategies and corresponding conditions if $c_t^h \ne c_t^n$

Game's Equilibrium Defender strategy	Attacker strategy	On-equilibrium path condition	Off-equilibrium path condition
$s1$	$w1$	$P_d \le \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, P_d \le \frac{g_a - c_a}{g_a + c_a^l}$	$q \le \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, q \le \frac{g_a - c_a}{g_a + c_a^l}$
	$w2$	$P_d \le \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, P_d \le \frac{g_a - c_a}{g_a + c_a^l}$	$q \ge \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, q \le 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$
	$w5$	$P_d \ge \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, P_d \le 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$	$q \ge \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, q \le 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$
	$w8$	$P_d \ge \frac{g_a - c_a}{g_a + c_a^l}, P_d \ge 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$	$q \ge \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, q \le 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$
	$w9$	$P_d \ge \frac{g_a - c_a}{g_a + c_a^l}, P_d \ge 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$	$q \ge \frac{g_a - c_a}{g_a + c_a^l}, q \ge 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$
$s4$	$w1$	$P_d \le \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, P_d \le \frac{g_a - c_a}{g_a + c_a^l}$	$q \le \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, q \le \frac{g_a - c_a}{g_a + c_a^l}$
	$w4$	$P_d \le \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, P_d \le \frac{g_a - c_a}{g_a + c_a^l}$	$q \ge \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, q \le 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$
	$w5$	$P_d \ge \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, P_d \le 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$	$q \ge \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, q \le 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$
	$w6$	$P_d \ge \frac{g_a - c_a}{g_a + c_a^l}, P_d \ge 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$	$q \ge \frac{c_t^n}{c_a + c_a^l + c_t^n - c_t^h}, q \le 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$
	$w9$	$P_d \ge \frac{g_a - c_a}{g_a + c_a^l}, P_d \ge 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$	$q \ge \frac{g_a - c_a}{g_a + c_a^l}, q \ge 1 - \frac{c_t^h}{g_a - c_a + c_t^h - c_t^n}$

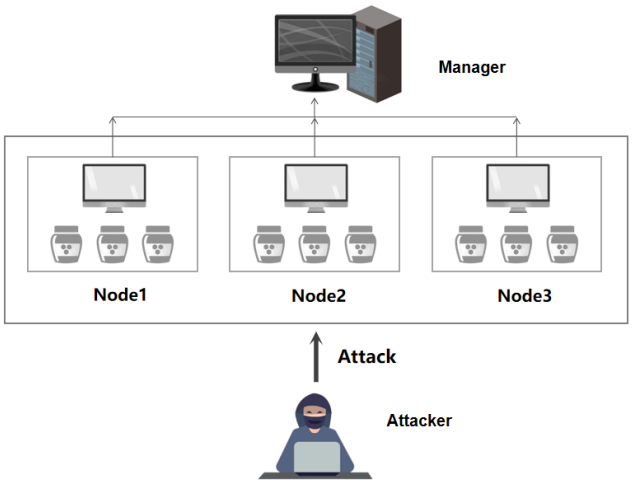

Figure 4: Attack Scenario. The attack and defense topography consists of the manager, the node ends and the attacker.

strategy $w9$, the attacker's expected payoff of action R should be greater than the expected payoff of action A and action T, so we have,

$$\begin{cases} 0 \geq q \cdot (-c_a - c_a^l) + (1-q) \cdot (g_a - c_a) \\ 0 \geq q \cdot (-c_t) + (1-q) \cdot (g_a - c_a - c_t) \end{cases}$$

which satisfies,

$$P_d \leq \frac{g_a - c_a}{g_a + c_a^l} \tag{11}$$

$$P_d \geq 1 - \frac{c_t}{g_a - c_a} \tag{12}$$

The analysis of the signaling deception game's equilibrium when the defender deploys strategy $s4$ is similar. If the defender gives a different signal for each attribute type, including strategy $s2$ and strategy $s3$, this game cannot lead to an equilibrium. The optimal strategies and corresponding conditions are summarized in Table 4.

Configurations of a honey attribute and a normal attribute are generally different and costs of tests on determining if an attribute is honey or normal can also be different. Assume that the test for a honey attribute costs c_t^h and test for a normal attribute costs c_t^n, the corresponding equilibrium and conditions are listed in Table 5.

5 EXPERIMENTAL ANALYSIS

The game model can quantify the benefits of attackers and defenders, and provide the strategic choice of active defense which meets the Nash equilibrium. We set up a real attack and defense scenario and demonstrate the efficacy of the proposed approaches.

We build a real honeypot system based on moving target defense, and use our game method to model and evaluate the system in practice. As shown in the Figure. 4, the experimental platform includes a manager and several node-ends. The manager is used to generate and manage the node-ends, and to receive, analyze and display the data returned by the node-ends. The node end is used to build a honeypot service and accept the control of the manager. Since increasing the deception attack surface will expand the total attack surface, we deploy honeypots in different nodes to balance the expanded attack surface and reduce the impact of the deception attack surface on the attack surface measurement. After deploying honeypots that sense and withstand attacks on the node, once the attacker scans and attacks the honeypot, the information on the attacker and his behavior will be immediately transmitted to the manager, where we can see the detailed data analysis. The attacker's goal is to scan the ports of the target host, identify the real service and launch the attack. The attacker may connect to the real host service or enter the honeypot. After a successful attack, the attacker can gain access to the service and steal important information.

We deploy the real attribute corresponding to the real service and the deceptive attribute corresponding to the virtual honeypot on the system. System services include file transfer service, remote connection service, web service, database services, etc. We open real services and virtual honeypots, and keep the corresponding ports open. Services are exposed to potential attacks, such as SSH may be subject to brute force, and web services may be attacked due to vulnerabilities. The real services and deceptive services deployed by the system, and their corresponding ports and possible attack forms are shown in the table 6. We can deceive attackers by deploying honeypot services or hiding real services. We first set all services to switch to deceptive services with a probability of 60%, so as to form a deception attack surface ($P_d = 0.6$). If the exposed falseness degree and the hidden truth degree of the system reach the standard, the probability of setting is reasonable. Otherwise, we need to readjust the probability. In this experiment, the standard setting used is that the exposed falseness degree is less than 0.3 and the hidden truth degree is greater than 0.5. In different scenarios, the standards for measuring the degree of deception can be appropriately adjusted based on the historical data, defenders' knowledge and experience. Signals are sent to attackers by marking different ports as real or deceptive. We can adopt four strategies from Strategy $s1$ to Strategy $s4$. The attacker can choose action A, action T, or action R according to the signal. Offensive and defensive confrontation is multi-stage. In each round of game confrontation, the defender can adjust the attributes and strategies, and the attacker can re-select the actions and the attributes to be attacked.

To demonstrate the effectiveness of our deception measurement method, we collected system services data and calculated the deception degree of the honeypot system according to the characteristics of the attributes. We calculated the deception degree of the system by using the exposed falseness degree and the hidden truth degree. In this experiment, the services in the system correspond to the attributes in the attack surface. Ten important features of services are extracted in each dimension. We assign the classification status of "0" or "1" to each feature component based on their existence, add the feature status values, and finally normalize them. The real mode and deceptive mode of each service are evaluated, and the evaluation results are listed in the table 7. The weight of each service is determined by its own access popularity or the proportion of each asset's value in the overall network. All results are normalized. According to the formula in Section 3 and probability P_d, the exposed falseness degree is 0.198, and the hidden truth degree is 0.833. Please note that 0 is the minimum and 1 is the maximum for both the exposed falseness degree and the hidden truth degree. The exposed falseness degree reflects the similarity between the real attributes and deceptive attributes in the deception system. A

Table 6: Types and functions of services

Name	Port	Type	Function	Attack Form
SSH honeypot	22	Remote connection service	Provide false SSH server	Brute force , weak password
Telnet honeypot	23	Remote connection service	Provide false Telnet server	Brute force , weak password
MYSQL honeypot	3306	Database service	Provide false MySQL sever	Injection attack, privilege escalation
REDIS honeypot	6379	Database service	Provide false REDIS sever	Unauthorized access, brute force
Gitlab honeypot	9093	Web service	Provide false login interface	Vulnerability, web attack
FTP honeypot	21	File transfer service	Provide false FTP server	Brute force, sniff
Oracle	1521	Database service	A database system	Injection attack, privilege escalation
MSSQL	1433	Database service	A database platform	Injection attack, privilege escalation
WebLogic	7001	Web service	Manage applications	Deserialization , weak password
SMTP	25	Mail service	Transmit mail information	Mail forgery

Table 7: Feature values and weights of services

Service	Weight	Interaction		Function		Configuration	
		real	deceptive	real	deceptive	real	deceptive
SSH honeypot	0.05	0.60	0.30	0.70	0.50	1.00	0.80
Telnet honeypot	0.05	0.50	0.30	0.60	0.50	1.00	0.80
MYSQL honeypot	0.15	0.50	0.30	0.70	0.50	1.00	0.80
REDIS honeypot	0.15	0.70	0.40	0.80	0.50	1.00	0.80
Gitlab honeypot	0.15	0.80	0.60	0.50	0.40	1.00	0.70
FTP honeypot	0.05	0.40	0.30	0.50	0.40	1.00	0.90
Oracle	0.15	0.80	0.70	0.90	0.80	1.00	0.90
MSSQL	0.05	0.50	0.40	0.70	0.60	1.00	0.90
WebLogic	0.15	0.80	0.60	0.70	0.60	1.00	0.80
SMTP	0.05	0.40	0.20	0.50	0.40	1.00	0.50

smaller exposed falseness degree indicates the real attribute and the deceptive attribute have a shorter Euclidean distance between them, and thus it's more difficult to distinguish them. That is, the smaller the exposed falseness degree, the better the deception effect. The hidden truth degree reflects the concealment degree of the real target in the deception system. The greater the hidden truth degree, the better the deception effect. The calculation results show that the deception metrics process well representation.

After ensuring that the system has appropriate metrics, we use game to build a model framework to represent the attack and defense confrontation, further optimize the deception probability according to the strategy and payoff, and give the strategy to meet the Nash equilibrium. Because the cost of attackers testing deceptive attributes and real attributes is different, we consider the case that $c_t^h \neq c_t^n$. The maximum value of benefits and costs is 10. Refer to historical statistics and multiple indicators, the attack costs are $c_a = 3.0, c_a^l = 2.0$, $c_t^h = 0.5$ and $c_t^n = 1$, and the attacker gains a value with $g_a = 5.00$ if he successfully compromises a normal attribute. The cost of setting a honey attribute is $c_d = 1.00$, and the defender loss a value with $c_d^l = 5.0$ if the attacker compromises a normal attribute. The defender gains a value with $g_d^o = 6.0$ if he can observe an attack on a deceptive attribute. The defender gains a value with $g_d^r = 5.0$ if the attacker retreats. We then evaluate and analyze the equilibrium results on the basis of Table 5. Only strategy $(s1, w5)$ and $(s4, w5)$ are possible and the defender's payoff is $0.6g_d^r - 0.6c_d - 0.4c_d^l = 0.4$. Thus, the attacker is most likely to adopt the attack strategy $w5$ in order to achieve the attack target, while the defender should take the initiative to adopt the Strategy $s1$ or $s4$ in order to achieve the best defense effect. The information interaction in the process of offensive and defensive confrontation is complex, so the information transmission in the signal game can better reflect the interaction. For the defender, the strategy of information transmission and the prediction of attack strategy are very important. Through the game, we can derive the best strategy for information transfer and the prediction of attack strategy. According to the defender's strategy and payoff, we can further optimize the model by adjusting the deception probability P_d in the next stage. It is proved that our proposed method can model and evaluate the attack and defense scenarios, and provide the optimal defense strategy.

6 CONCLUSION

In this paper we propose a concept of deception attack surface to illustrate the deception-based moving target defense techniques. We also propose indicators that can measure the degree of deception in MTD systems. Based on this, we formulate the deception game model between an attacker and a defender. We also give the detailed example scenario to analyze the deception game's equilibrium. Experiments show that our method is effective. For future work, we may investigate the optimal strategies under a hybrid strategy to provide more effective deception techniques and strategies.

REFERENCES

[1] Md Ali Reza Al Amin, Sachin Shetty, Laurent Njilla, Deepak K Tosh, and Charles Kamhoua. 2021. Hidden markov model and cyber deception for the prevention of adversarial lateral movement. *IEEE Access* 9 (2021), 49662–49682.

[2] Ehab Al-Shaer, Qi Duan, and Jafar Haadi Jafarian. 2012. Random host mutation for moving target defense. In *International Conference on Security and Privacy in Communication Systems*. Springer, 310–327.

[3] Hooman Alavizadeh, Jin B Hong, Julian Jang-Jaccard, and Dong Seong Kim. 2018. Comprehensive security assessment of combined MTD techniques for the cloud. In *Proceedings of the 5th ACM Workshop on Moving Target Defense*. 11–20.

[4] Hooman Alavizadeh, Jin B Hong, Dong Seong Kim, and Julian Jang-Jaccard. 2021. Evaluating the effectiveness of shuffle and redundancy mtd techniques in the cloud. *Computers & Security* 102 (2021), 102091.

[5] Massimiliano Albanese, Ermanno Battista, and Sushil Jajodia. 2016. Deceiving attackers by creating a virtual attack surface. In *Cyber Deception*. Springer, 167–199.

[6] Kevin Borders, Laura Falk, and Atul Prakash. 2007. OpenFire: Using deception to reduce network attacks. In *2007 Third International Conference on Security and Privacy in Communications Networks and the Workshops-SecureComm 2007*. IEEE, 224–233.

[7] Thomas E Carroll, Michael Crouse, Errin W Fulp, and Kenneth S Berenhaut. 2014. Analysis of network address shuffling as a moving target defense. In *2014 IEEE international conference on communications(ICC)*. IEEE, 701–706.

[8] Thomas E Carroll and Daniel Grosu. 2011. A game theoretic investigation of deception in network security. *Security and Communication Networks* 4, 10 (2011), 1162–1172.

[9] Ankur Chowdhary, Sailik Sengupta, Adel Alshamrani, Dijiang Huang, and Abdulhakim Sabur. 2019. Adaptive MTD security using Markov game modeling. In *2019 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 577–581.

[10] Fred Cohen et al. 2010. Moving target defenses with and without cover deception.

[11] Warren Connell, Daniel A Menascé, and Massimiliano Albanese. 2017. Performance modeling of moving target defenses. In *Proceedings of the 2017 Workshop on Moving Target Defense*. 53–63.

[12] Basirudin Djamaluddin, Ahmed Alnazeer, and Farag Azzedin. 2018. Web deception towards moving target defense. In *2018 International Carnahan Conference on Security Technology (ICCST)*. IEEE, 1–5.

[13] Fei Fang. 2021. Game Theoretic Models for Cyber Deception. In *Proceedings of the 8th ACM Workshop on Moving Target Defense*. 23–24.

[14] Xiaotao Feng, Zizhan Zheng, Derya Cansever, Ananthram Swami, and Prasant Mohapatra. 2017. A signaling game model for moving target defense. In *IEEE INFOCOM 2017-IEEE conference on computer communications*. IEEE, 1–9.

[15] Chungang Gao, Yongjie Wang, Xinli Xiong, and Wendian Zhao. 2021. Mtdcd: an mtd enhanced cyber deception defense system. In *2021 IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, Vol. 4. IEEE, 1412–1417.

[16] Jason R Hamlet and Christopher C Lamb. 2016. Dependency graph analysis and moving target defense selection. In *Proceedings of the 2016 ACM Workshop on Moving Target Defense*. 105–116.

[17] Xiao Han, Nizar Kheir, and Davide Balzarotti. 2017. Evaluation of deception-based web attacks detection. In *Proceedings of the 2017 Workshop on Moving Target Defense*. 65–73.

[18] Jin B Hong and Dong Seong Kim. 2015. Assessing the effectiveness of moving target defenses using security models. *IEEE Transactions on Dependable and Secure Computing* 13, 2 (2015), 163–177.

[19] Jafar Haadi H Jafarian, Ehab Al-Shaer, and Qi Duan. 2014. Spatio-temporal address mutation for proactive cyber agility against sophisticated attackers. In *Proceedings of the First ACM Workshop on Moving Target Defense*. 69–78.

[20] Sushil Jajodia, Anup K Ghosh, Vipin Swarup, Cliff Wang, and X Sean Wang. 2011. *Moving target defense: creating asymmetric uncertainty for cyber threats*. Vol. 54. Springer Science & Business Media.

[21] Hai Jin, Zhi Li, Deqing Zou, and Bin Yuan. 2019. Dseom: A framework for dynamic security evaluation and optimization of mtd in container-based cloud. *IEEE Transactions on Dependable and Secure Computing* 18, 3 (2019), 1125–1136.

[22] Duohe Ma, Liming Wang, Cheng Lei, Zhen Xu, Hongqi Zhang, and Meng Li. 2017. Quantitative security assessment method based on entropy for moving target defense. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*. 920–922.

[23] Hoda Maleki, Saeed Valizadeh, William Koch, Azer Bestavros, and Marten Van Dijk. 2016. Markov modeling of moving target defense games. In *Proceedings of the 2016 ACM workshop on moving target defense*. 81–92.

[24] Pratyusa K Manadhata. 2013. Game theoretic approaches to attack surface shifting. In *Moving Target Defense II*. Springer, 1–13.

[25] Tuan Anh Nguyen, Minjune Kim, Jangse Lee, Dugki Min, Jae-Woo Lee, and Dongseong Kim. 2022. Performability evaluation of switch-over Moving Target Defence mechanisms in a Software Defined Networking using stochastic reward nets. *Journal of Network and Computer Applications* 199 (2022), 103267.

[26] Richard Poschinger, Nils Rodday, Raphael Labaca-Castro, and Gabi Dreo Rodosek. 2020. OpenMTD: A Framework for Efficient Network-Level MTD Evaluation. In *Proceedings of the 7th ACM Workshop on Moving Target Defense*. 31–41.

[27] L. C. Ricord. 1996. The Cuckoo's Egg: Tracking a Spy through the Maze of Computer Espionage. *Topics in Health Information Management* 43, 8 (1996), 75–76.

[28] Joshua Taylor, Kara Zaffarano, Ben Koller, Charlie Bancroft, and Jason Syversen. 2016. Automated effectiveness evaluation of moving target defenses: Metrics for missions and attacks. In *Proceedings of the 2016 ACM Workshop on Moving Target Defense*. 129–134.

[29] Matheus Torquato, Paulo Maciel, and Marco Vieira. 2021. PyMTDEvaluator: A Tool for Time-Based Moving Target Defense Evaluation: Tool description paper. In *2021 IEEE 32nd International Symposium on Software Reliability Engineering (ISSRE)*. IEEE, 357–366.

[30] Vincent E Urias, William MS Stout, and Caleb Loverro. 2015. Computer network deception as a moving target defense. In *2015 International Carnahan Conference on Security Technology (ICCST)*. IEEE, 1–6.

[31] C. Wang and Z. Lu. 2018. Cyber Deception: Overview and the Road Ahead. *IEEE Security Privacy* 16, 2 (2018), 80–85.

[32] Dayong Ye, Tianqing Zhu, Sheng Shen, and Wanlei Zhou. 2020. A differentially private game theoretic approach for deceiving cyber adversaries. *IEEE Transactions on Information Forensics and Security* 16 (2020), 569–584.

[33] Kara Zaffarano, Joshua Taylor, and Samuel Hamilton. 2015. A Quantitative Framework for Moving Target Defense Effectiveness Evaluation. In *Acm Workshop*. 3–10.

[34] Mengyuan Zhang, Lingyu Wang, Sushil Jajodia, and Anoop Singhal. 2018. Network attack surface: Lifting the concept of attack surface to the network level for evaluating networks' resilience against zero-day attacks. *IEEE Transactions on Dependable and Secure Computing* 18, 1 (2018), 310–324.

[35] Yaqin Zhang, Duohe Ma, Xiaoyan Sun, Kai Chen, and Feng Liu. 2020. WGT: Thwarting Web Attacks Through Web Gene Tree-based Moving Target Defense. In *2020 IEEE International Conference on Web Services (ICWS)*. IEEE, 364–371.

[36] Yaqin Zhang, Duohe Ma, Xiaoyan Sun, Kai Chen, and Feng Liu. 2020. What You See Is Not What You Get: Towards Deception-Based Data Moving Target Defense. In *2020 IEEE 39th International Performance Computing and Communications Conference (IPCCC)*. IEEE, 1–8.

[37] Quanyan Zhu and Tamer Başar. 2013. Game-theoretic approach to feedback-driven multi-stage moving target defense. In *International conference on decision and game theory for security*. Springer, 246–263.

[38] Rui Zhuang, Scott A. Deloach, and Xinming Ou. 2014. Towards a Theory of Moving Target Defense. *ACM* (2014).

[39] Rui Zhuang, Su Zhang, Scott A DeLoach, Xinming Ou, Anoop Singhal, et al. 2012. Simulation-based approaches to studying effectiveness of moving-target network defense. In *National symposium on moving target research*, Vol. 246.