Hindawi
Security and Communication Networks
Volume 2018, Article ID 3759626, 25 pages
https://doi.org/10.1155/2018/3759626

Review Article

Moving Target Defense Techniques: A Survey

Cheng Lei,[1,2] **Hong-Qi Zhang**,[1,2] **Jing-Lei Tan**,[1,2] **Yu-Chen Zhang**,[1,2] **and Xiao-Hu Liu**[1,2]

[1]*China National Digital Switching System Engineering & Technological Research Center, Zhengzhou, Henan Province 450001, China*
[2]*Henan Key Laboratory of Information Security, Zhengzhou, Henan Province 450001, China*

Correspondence should be addressed to Cheng Lei; leicheng12150@gmail.com

Cheng Lei and Hong-Qi Zhang contributed equally to this work.

Received 6 February 2018; Revised 30 May 2018; Accepted 15 July 2018; Published 22 July 2018

Academic Editor: Jesús Díaz-Verdejo

As an active defense technique to change asymmetry in cyberattack-defense confrontation, moving target defense research has become one of the hot spots. In order to gain better understanding of moving target defense, background knowledge and inspiration are expounded at first. Based on it, the concept of moving target defense is analyzed. Secondly, literature analysis method is adopted to explain the design principles and system architecture of moving target defense. In addition, some relevant key techniques are introduced from the aspects of strategy generation, shuffling implementation, and performance evaluation. After that, the applications of moving target defense in different network architectures are illustrated. Finally, existing problems and future trend in this field are elaborated so as to provide a basis for further study.

1. Introduction

With the continuous popularization and deepening of network applications, the early discrete and independent individuals have become highly correlative and interdependent among each other now. Internet of everything not only breeds the new normal of human society but also supports the efficient operation of critical infrastructure in important national fields [1]. In recent years, since the problems of sensitive information leakage, industrial systems collapsed, and financial services disrupted are more and more serious, the penetration and destruction of cyberspace pose a serious threat to all sectors of society. Chinese President Jinping Xi pointed out: "Without cybersecurity, there is no national security. Without informatization, there is no modernization." [2] Regarding the practical problems and potential threats suffered in current cyberspace, moving target defense (MTD), as one of the game-changing themes, provides a new idea to improve cyberspace security.

In the past few years, MTD developed rapidly and a large number of theories, methods, and technical research results have emerged. Previous review work mainly introduced the concept and the defensive strategies of MTD. They are rare to systematically summarize the inspiration, the exploration of MTD theory, the research of different defensive architecture, and the practical applications of MTD techniques.

2. Research Background of MTD

Due to the limitations of human's cognitive approaches, the defects in system design and engineering practice are inevitable, which forms network resource vulnerabilities. The core of network attack and defense is focused on the utilization of network resource vulnerability. The typical offensive and defensive modes are introduced in this section. Based on that, the root source of the asymmetric situation between offensive and defensive sides is analyzed.

2.1. Typical Offensive and Defensive Mode. From the attackers' prospect, network attacks are intended to analyze the targeted system and to find out the ubiquitous existence of vulnerability in network resource. After that, attackers take intrusive actions to bring about great loss to benign users. As widely accepted, cyber kill-chain is a multistage segmental type intrusive model proposed by Lockheed Martin cooperation [3] and now has been widely accepted. It describes the

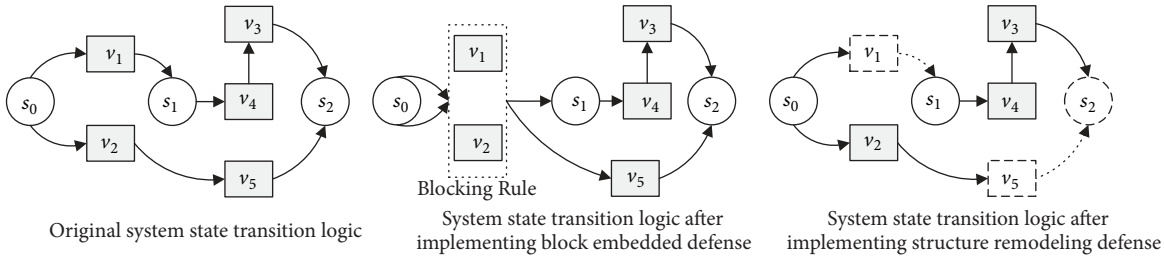


Figure 1: Illustration of existing defensive methods.

common intrusive behavioral pattern with two stages. Left-of-exploitation is to gather information and construct assault arsenal. Attackers explore and lock targeted system during reconnaissance. Besides, according to the analyzed network features, attackers formulate attack tools and methods. Right-of-exploitation of cyber kill-chain is mainly to carry out attack behavior and extend the damage scope. Then, the desired states of targeted systems can be achieved. Furthermore, attackers exploit similar vulnerabilities to enhance the offensive effectiveness.

From the perspective of defenders, existing defensive methods can be mainly divided into block embedded defense and structure remodeling defense. Illustrated in Figure 1, considering the defects of a system, block embedded defense method makes irregularity system state unreachable by using techniques such as access control and physical isolation. For instance, access control adds access control policy as the blocking rule to prevent unauthorized access. After that, attackers cannot access system to get the sensitive information they want. As a result, sensitive information in a system can be protected. Regardless of the universality of this kind of method, absolute obstruction, it becomes more and more difficult for the appearance of side channel attackers. Besides, the conflict and storage space explosion of policy sets emerge with the increase of the number of rules added. As for structure remodeling defense method, it mainly copes with the inherent flaw of targeted systems. Structure remodeling method makes network resource vulnerabilities unusable by inserting plug-ins and patching. For instance, upgrading system security can fix some vulnerabilities in the system. Attackers cannot launch attacks without exploiting corresponding vulnerabilities. As a result, the security of the system improves. Although this kind of method can invalidate attacks based on specific vulnerability exploitation, it is impractical to remodel the overall systems. What is more, it cannot ensure there is no bug in new system logic. Moreover, even partial logical remodeling is implemented, and cognitive limitations make it difficult to cover all critical vulnerabilities accurately.

According to the above analysis, existing defensive methods are difficult to resist continuous reconnaissance and long-term analysis in the attack phase of left-of-exploitation because of the stationarity of defensive mechanisms. Additionally, the cognitive limitations make it difficult to find all potential vulnerabilities exploited by attackers in right-of-exploitation phase.

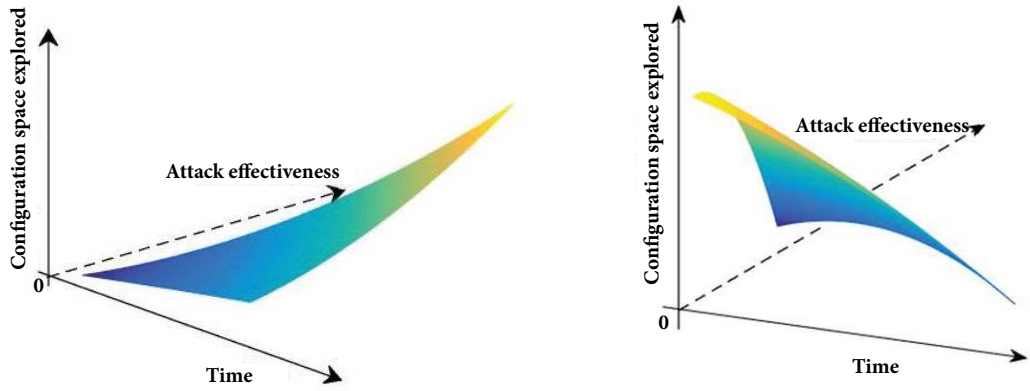
2.2. The Root Source of Offensive and Defensive Asymmetry Situation. While network security is faced with the harsh challenge of being easily attacked but hardly defended, the root source can be attributed to the following points [4]:

(1) The certainty of network composition provides the prerequisites for attackers to long-term analysis. Attackers can pinpoint the targeted system by collecting information in network. After that, more influential security threats are created by combining known and zero-day vulnerabilities. While defensive method based on prior knowledge is difficult to enumerate all possible attack methods or to explore all potential resource vulnerabilities. Therefore, the gap between attackers' comprehensive analysis to targeted systems and defenders' little knowledge to security threats results in the information advantage of attackers.

(2) The static nature of network structure provides the necessary interdependent environment for attackers to implement intrusion. Attackers implement control through installing backdoor plug-ins. The longer the network systems are used, the more time attackers have to exploit vulnerabilities. On the other hand, it is difficult for defenders to detect new types of security threats in real-time. In additions, the lag of patch issued provides sufficient time for attackers to implement intrusion. Consequently, long-term planning for attackers and real-time response requirements for defenders leads to the time advantage of attackers.

(3) The homogeneity of network elements provides the living space for attackers to expand the damage scope. Attackers can threaten the entire network system by only finding one flaw. Furthermore, the common vulnerability can be exploited in different systems in multiple attacks. On the other hand, defenders need to apply different defensive means comprehensively so as to achieve antivirus and patching vulnerabilities. Besides, in order to improve the security of targeted systems, defenders need to mitigate all vulnerabili ties. As a result, the commonly effective attack means and the comprehensive defensive approach lead to the cost advantage of attackers.

As the network system tends to be combined, automated, intelligent, and complicated, the sharp contrast between attackers adopting simple methods, small tools, and various vulnerabilities to launch effective intrusion with defenders implementing complex strategies, coordinated mechanisms, and comprehensive deployment further aggravates the asymmetry situation of the offensive and defensive side.



The trend of attack implementation over time under traditional defensive mechanism

The trend of attack implementation over time under moving target defense mechanism

FIGURE 2: The trend of attack implementation over time under different defensive mechanisms.

3. The Inspiration and Development of MTD Concept

How to break the asymmetric situation to achieve network security goal? In recent years, moving target defense, as one of the revolutionary techniques to change game rules, tries to create a conducive network environment for defense by increasing the attackers' apparent uncertainty.

3.1. Inspiration of Moving Target. The inspiration of "moving target" has long been applied in many fields such as biology, military, and cryptography development:

(1) In Biology. Creatures such as chameleons camouflage in the wild imitate the characteristics of surroundings. Mimicry octopus defends the enemy by imitating the appearance or behavior of other creatures [5]. Besides, dynamic immune mechanism and diverse structures of human immune system allow healthy human with the magnitude of 10^{14} cells carrying virus or bacteria with the magnitude of 10^{15}.

(2) In Military. Compared with fixed targets, moving targets in weapon shooting can greatly reduce the hit rate. In modern electronic warfare, frequency hopping communication can effectively enhance the anti-interference capability [6]. At the same time, both the Bagua Zhen, the Art of War in ancient military, and the guerrilla military thinking of Chairman Mao in modern warfare are to confuse opponents by constantly changing.

(3) In Cryptography. Cryptogram coding is the translation process from plaintext information codes to ciphertext information code according to appointment rules. Evolving cryptography is a combination of cryptology and evolutionary computation, which improves the adaptability of cryptography by borrowing biological evolution theory [7]. Besides, variable algorithm cryptography cluster is to resist cryptanalytic attacks effectively by improving the diversity and efficiency of cryptographic services.

Inspired by the fact that "moving " is better than "stationary", and "polymorphic" is better than "static", moving target defense changes the target of attackers from fixed one to a more dynamic one by actively changing network configurations over time [8]. Similarly, mimic cyber defense concept [9] and reconfigurable security computing [10] ideas also come into being.

3.2. Connotation and Extension of MTD Concept. The concept of moving target defense was proposed at first in the U.S. national cyber leap year summit in 2009 [11]. In 2012, the report of White House national security council explains the meaning of "moving target", which is systems that move in multiple dimensions to go against attacks and increase system resiliency. In 2014, moving target defense concept is defined as follows by Federal Cybersecurity Research and Development Program.

Definition 1. Moving target defense [12] enables us to create, analyze, evaluate, and deploy mechanisms and strategies that are diverse and that continually shift and change over time to increase complexity and cost for attackers, limit the exposure of vulnerabilities and opportunities for attack, and increase system resiliency.

Figure 2 describes the trend of attack benefit over time under traditional defense and moving target defense, respectively. Under traditional defense, due to the static and certainty nature of the network, the configuration space explored by attackers gradually expands as time goes by. At the same time, the exploited vulnerabilities and attack path constructed also increase continuously. What is more, with the homogeneity of network elements, attackers can exploit similar vulnerabilities and construct backdoors to extend the damage scope. Therefore, as color changes from blue to yellow on the left in Figure 2, the effectiveness of attack can achieve higher benefit with lower cost over time. On the other hand, since MTD mechanism increases the uncertainty of network composition, the availability of reconnaissance

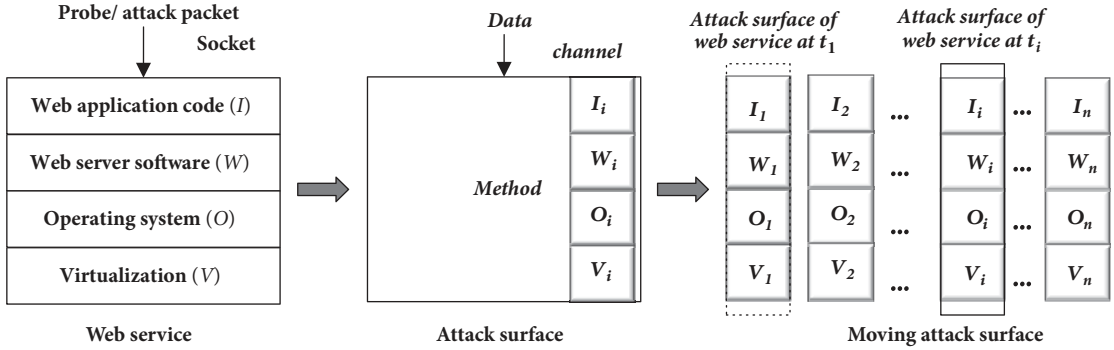


Figure 3: Attack surface and moving attack surface of web service.

and analysis results of attackers is reduced. Accordingly, the information asymmetry of attackers and defenders becomes less and less. Furthermore, even though attackers can exploit vulnerabilities and construct attack path successfully, attack time will be compressed with the time-varying of the system. Thereby, the time gap between attackers and defenders is getting smaller and smaller. At last, even if attackers can implement intrusive actions successfully, they are hard to achieve high benefit with low cost because of the diversity of system. As a result, as color changes from yellow to blue on the right in Figure 2, the attack effectiveness is less and less. What is more, the root is that MTD breaks the necessary conditions of cyber kill-chain implementation from the perspectives of information gathering and analysis, intrusive behavior implementation, and attack damage expansion, thus changing the asymmetric situation.

4. Theory and Architecture Design Study of MTD

4.1. Theory Study of MTD. The study of MTD theory is to elaborate the key elements which determine the effectiveness of MTD implementation. It defines the related concept and explains the relationship among those key elements. Based on the description method, theory study is summarized from the perspective of attack surface and attack graph in this section.

(1) MTD Theory Based on Attack Surface. Attack surface describes the set of vulnerabilities existing in network systems. With the unproven feature of system security, attack surface necessarily exists. Mandahata *et al.* [13] define the concept of attack surface as follows.

Definition 2 (attack surface (AS)). Given a system s, and its environment E_s, the attack surface of s is triple $\langle M^{E_s}, C^{E_s}, I^{E_s}\rangle$, where M^{E_s} is the set of methods to entry and exit, C^{E_s} is the set of channels, and I^{E_s} is the set of untrusted data items.

Based on it, Manadhata *et al.* [14] define attack surface shift so as to adjust to the dynamic, random, and diverse features of MTD. The definition is as follows.

Definition 3 (attack surface shift). Given a system s and its environment E_s, the old attack surface of s is R_o, and the new attack surface of s is R_n. The attack surface of s has shifted if there exists at least one resource $r \in (R_o \setminus R_n)$ or $(r \in R_o \cap R_n) \wedge (r_o \succ r_n)$.

In Definition 3, r_o refers to the contributions that r made to R_o, whereas r_n refers to those of R_n. In addition, $r_o \succ r_n$ means that r_o makes a larger contribution to the attack surface than r_n. To solve the problems of inconformity between assumption of invariance and continuous reachable in attack surface and MTD features, Huang *et al.* [15] proposed the concept of moving attack surface. It believes that the uncertainty and reachable of attack surface are the key features to implement MTD. The uncertainty refers to the continuously changing attributes and vulnerabilities in attack surface. The reachability means that attackers cannot determine whether the vulnerabilities being exploited are reachable within a certain period of time. For instance, web services typically include a mix of web application code (I), web server software (W), operating systems (O), and virtualization (V) technologies. The attack surface of web service is shown in Figure 3; it consists of probe/attack packet as the data, socket as the channel, and $\{I, W, O, V\}$ as the method. Since there are different IP configuration and various versions of web service, the attack surface of web service can change over time by adopting MTD. Zhuang *et al.* [16] define MTD system as a triple: $\langle \Gamma, G, P\rangle$. Γ refers to system configuration. It can be regarded as a set of configuration parameters. G refers to system performance goals and security goals. Performance goals are the goal set for accomplishing specific tasks, while security goals are the goal set to ensure the security of critical parts in the system. As for P, it refers to system strategy. It is necessary constraint condition to ensure system reaching some specific states. After analyzing, it believes that MTD should ensure the system performance at first. On that basis, it achieves its security goal by increasing the diversity of system configuration space and improving the uncertainty of configuration parameters. Furthermore, literature [17] proposed offensive and defensive theory in MTD. Analysis shows that attack cost under existing defensive mechanisms is the consumption of attackers to locate specific parameter value from the value range of configuration parameter and the consumption to

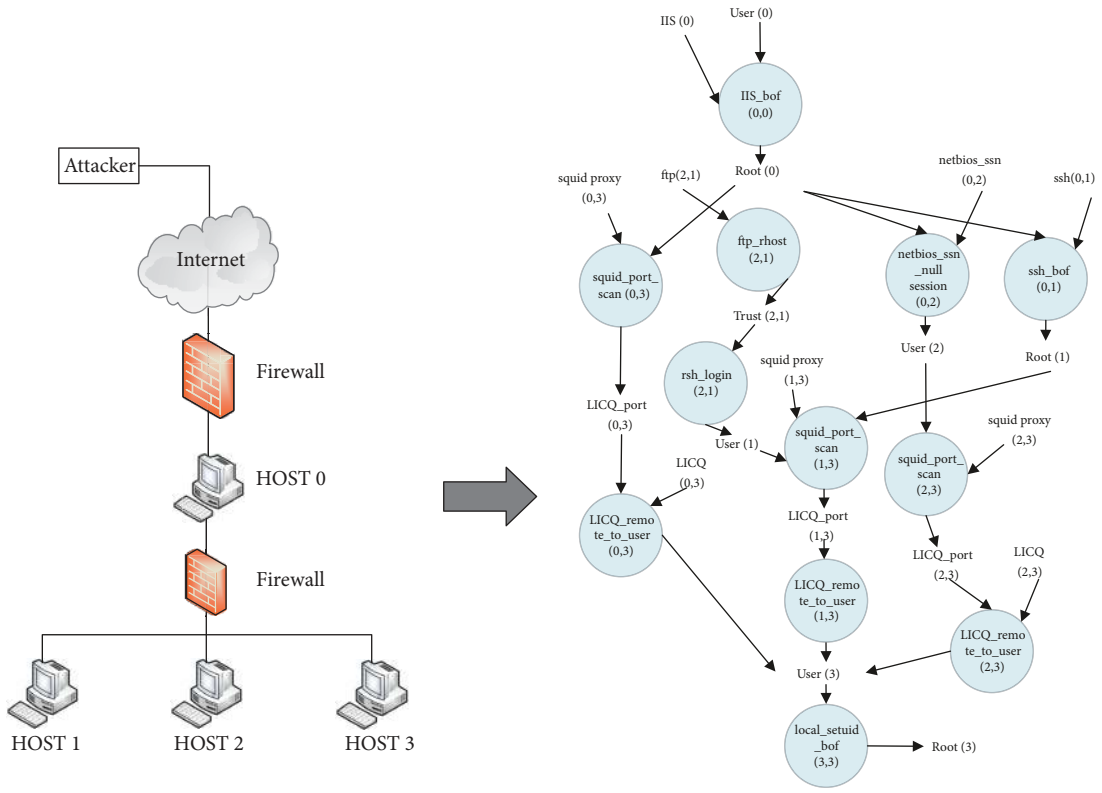


FIGURE 4: Network configuration and its attack graph.

convert system state into the set of preset states. However, the attack cost under MTD is the consumption of attackers to determine the set of valid configuration parameters from exploration space and the consumption in existing defensive mechanism. In other words, the key features of MTD are to expand the configuration space explored by attackers and to reduce the timeliness of information gathered and analyzed by attackers.

(2) MTD Theory Based on Attack Graph. Attack graph is a way to describe complex attack sequence that causes system state transition by considering vulnerability, attack goals, and node connectivity in targeted system simultaneously. Since the state attack graph causes inefficiency and combinatorial space explosion problems in the process of graph construction when the network scale enlarges, attribute attack graph is defined based on the "monotonicity" hypothesis proposed by Amman *et al.* [18].

Definition 4. Attribute attack graph can be presented as a triple: $AAG = (A_0 \cup A_d, T, E)$. A_0 refers to the set of initial nodes, which corresponds to the initial attributes of network and attackers occupies at first. A_d is the set of reachable nodes, which is corresponding to the attributes of network and attackers occupies after the intrusion implementation. T is atomic attack set, and E means directed edges.

As is shown in Figure 4, the network configuration is shown on the left side, and attack graph of this network is shown on the right side. This attack graph describes the attack path under the condition of the network configuration in [19]. Each node in this attack graph refers to one specific vulnerability exploited by the attacker in the network. For example, if the attacker gains the rights of HOST 3, it can gain the root privilege of HOST3 after exploiting *local_setuid_bof* in HOST3. Based on the given network configuration, vulnerabilities, and connectivity-limiting firewall policies, the success rate of different attack path in this attack graph can be calculated. In addition, MTD implementation can change the configuration of network attributes and main elements of network. Consequently, the attack graph and success rate of different attack path will be changed accordingly. Zhuang *et al.* [20–22] show that the implementation of MTD enables attackers to stall or fallback to one state node, which will lead to the explosion of attribute space because it is against monotonicity assumption. Therefore, attribute space is reduced effectively by setting network endpoint as a node in the attribute attack graph. What is more, Hong *et al.* [23] proposed a novel MTD analysis method based on hierarchical attack graph model. The hierarchical attack graph consists of two layers. On the other hand, aimed at the problem of hard to describe the interference of MTD implementation to the targeted system usability, Hamlet *et al.* [24] proposed a novel analysis method based on

dependency graph. After analyzing the effect of address space layout randomization implementation on the dependency relationship of the targeted system, it shows that it can find the shortest path in the dependency relationship so as to reduce the interference of MTD implementation to targeted system performance. In addition, in order to increase the difficulty of attack implementation, it can find out and block the shortest dependency relationship path exploited by attackers.

Consequently, MTD changes the type of vulnerabilities exposed and limits the exposure time in a diversified, random, and dynamic manner. Theory analysis of MTD based on attack surface explains the defense principle by analyzing the influence of MTD implementation on the set of vulnerabilities in the targeted system. However, theory analysis methods cannot expound the interference of MTD implementation to network system performance. Aimed at that problem, theory analysis methods based on attack graph are to state the interference of MTD implementation to the correlated relationship among vulnerabilities and network states.

4.2. Design Principle and System Architecture of MTD. The design principle of MTD illustrates the requirements that should be satisfied with, the conditions that should be met, and the capabilities MTD implementation needs. Hobson *et al.* [25] and Okhravi *et al.* [26, 27] proposed three requirements of MTD, which are coverage, unpredictability, and timeliness. Coverage refers to the ratio of vulnerability set transformed by MTD and vulnerability set exploited by attackers. It is prerequisite for effectively implementing MTD. The higher the coverage ratio is, the more specific the defense is. Unpredictability refers to the uncertainty degree of attackers' view to attack surface. The greater the unpredictability is, the harder for attackers to obtain and exploit vulnerabilities. It is the key factor in MTD implementation. Besides, timelines refer to the change frequency of MTD. Only when the triggering time is ahead of attack implementation time, MTD implementation is effective. It is the guarantee of effective implementation. On the other hand, due to the additional performance overhead caused by MTD implementation, the study of Zhaung *et al.* [16] shows that the first and foremost principle is to guarantee the availability of the targeted system. Based on it, the natural diversity and artificial diversity should be made good use so as to achieve dynamic and stochastic transformation. Green *et al.* [28] give more specific design principles in MTD design. The first principle is moving nature, where only benign users can pass the matching process. The second one is accessibility feature, which means that if and only if users pass authentication, can they get access to the network resource. The last one is the distinguishable feature, which means that MTD can distinguish trusted endpoint from untrusted ones.

By analyzing the design principle proposed above, it can be seen that the requirements of MTD proposed by Hobson and Okhravi *et al.* are mainly from the perspective of the defensive effectiveness of MTD. Since MTD is to improve the security of protected systems, the core elements affecting MTD implementing effectiveness can come down to coverage, unpredictability, and timeliness. Furthermore, the design principles proposed by Zhuang and Green *et al.* not only take MTD defensive effectiveness into account but also take system availability into consideration. Therefore, it is mainly from the perspective of both system availability and MTD defensive effectiveness. For the moving nature of MTD, it means that MTD system is unpredictability with the continuous transformation from the view of malicious attackers, while it is in a relative static state from the view of benign users. As a result, the targeted system is static only for benign users. For the accessibility feature, all users have to authenticate before getting access to the network resource. After authentication, only benign users can get access to the types of network resource permitted, while malicious attackers cannot bypass MTD to get access to network resource they want. Furthermore, the moving feature and accessibility feature of MTD system implementation, malicious attackers, and benign users can be distinguished. Consequently, network resource in the targeted system is available only for benign users. Based on the above studies, the basic principles in designing MTD should be obeyed as follows:

(1) Coverage: MTD should transform all vulnerabilities that might be exploited in a dynamic and random way. Specifically, it should cover vulnerabilities in critical resources.

(2) Unpredictability: network system is able to have sufficient heterogeneous and redundant components. Consequently, network elements should have enough transformation space.

(3) Timeliness: since not all vulnerabilities can be transformed, MTD should trigger in time transformation so as to compress intrusive actions.

(4) Superstability: when there are a variety of MTD mechanisms implemented in the targeted network system, the effectiveness of those mechanisms should be equivalent to a more complex one. At the same time, MTD should be able to synergize with existing defensive mechanisms.

(5) Functional equivalence: although network elements need to shift constantly during the implementation of MTD, the functions of the protected system should keep providing.

Based on the principals summarized above, different kinds of MTD architecture are summed. Since the definition of MTD in narrow perspective is to achieve proactive defense by constantly and randomly transforming network elements, proactive defense architecture adopted in mechanisms such as MT6D [29] and Address Space Randomization (ASR) [30] is shown on the left side of Figure 3. This architecture is mainly composed of two parts. The first part is strategy formulation, which is to preset security policy based on the result of expert experience or automated analysis. The other part is defensive implementation, which is to select transformation elements and set mutation period. However, since proactive defensive architecture has the disadvantage of blind defense and low defensive benefit, Zhuang *et al.* [31] proposed reactive defensive architecture. As is shown on the right side of Figure 3, reactive defensive architecture adds analysis engine and logic security model part to enhance the defense targeted. What is more, Zhou *et al.* [32] proposed MTD with evolvability by introducing evolutionary theory. It achieves evolvability by constructing a closed loop of defensive strategy generation, defensive mechanism implementation, and

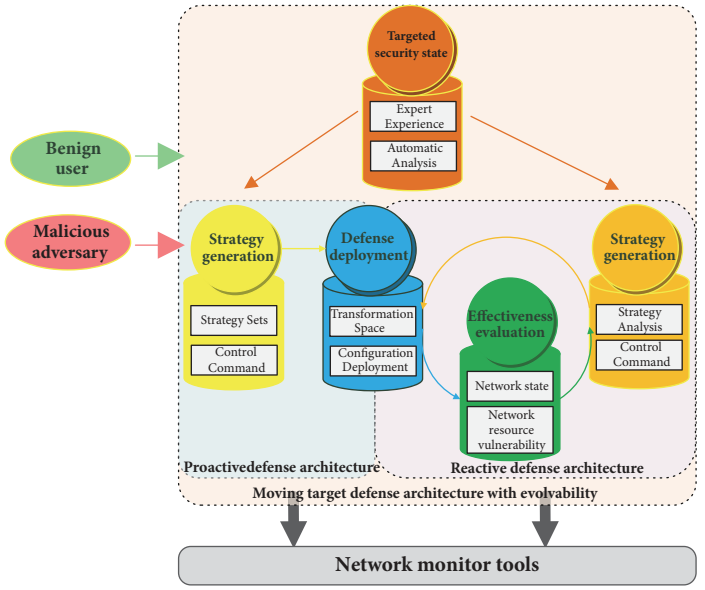


Figure 5: Different moving target defense architecture.

defensive effectiveness evaluation. What is more, in order to manage MTD, Carvalho *et al.* [33] proposed task awareness architecture based on elastic proxy. Consequently, MTD can be deployed to more dedicated or hybrid environment with different service requirements or operational constraints. At the same time, literature [34] proposed MTD based on man-machine collaboration, which is to cope with the deviation of data analysis in strategy formulation. By combing the result of expert experience with that of machine learning, it ensures the accuracy of strategy generation and the effectiveness of mechanism implementation.

As is shown in Figure 5, the architecture of MTD transits from proactive defense architecture to reactive defense architecture. Furthermore, it gradually develops into architecture with evolvability. In order to balance the proactiveness and pertinence during defense, defense architecture with evolvability combines the advantage of proactive defense and reactive defense. It adopts predefined transformation so as to achieve defense without detection. In the process of defense implementation, architecture with evolvability analyzes the state of the targeted system in real-time and formulates more targeted defensive strategy by adopting strategy generation, mechanism deployment, and effectiveness evaluation. Consequently, it improves the self-adaptive of MTD. Besides, as the application scenarios of MTD tend to be distributed in deployment, complex of targeted system structure, and complicated in operation and maintenance, command and control of MTD are also gradually transformed from manual or automatic way singly into the combination of expert experience and automation, thus improving the efficiency of defense deployment.

5. Key Techniques in MTD

According to the composition of MTD architecture, this section summarizes three key parts, which are strategy formulation, transformation mechanism, and effectiveness evaluation.

5.1. Research on the MTD Strategy Formulation. MTD strategy formulation is to generate the optimal defense strategy that meets the expected safety goals after analyzing existing network conditions and potential security threats. According to the different theories adopted, existing studies can be divided into three categories, shown in Table 1.

5.1.1. Strategy Formulation Based on Game Theory. Game Theory [35] is a kind of strategic selection method to achieve maximum benefit of each rational player in the pattern of mutual interest. The equilibrium of game is the steady state of rational players when no player can increase their own profit by changing their strategy unilaterally. Because the opposition, dependency, and noncooperative features in network confrontation are highly compatible with the feature of game theory, thus making the optimal strategy selection based on game theory becoming one of the hotspots to study.

(1) Strategy Selection under Complete Information Assumption. Since MTD implementation leads to the change of system state, Manadhata *et al.* [14] constructed two-player stochastic dynamic game model, and the impact of attack surface transformation on both offensive and defensive behaviors and system status is analyzed. Valizadeh *et al.* [36] regard the state transition of MTD as Markov decision process. The impact of different MTD strategies selected at the next moment is analyzed based on it. Since Markov chain does not take the impact of offensive and defensive behaviors on system state into consideration, Lei *et al.* [37] combine Markov decision process and dynamic game to describe confrontation process of MTD. On the other hand, since attackers with self-learning capability can infer defensive strategy after observing existing defensive behaviors, Feng *et al.* [38] combine Stackelberg

Table 1: Classification of MTD strategy formulation method.

Category	Classic Literatures	Description
Strategy formulation based on game theory	Manadhata *et al.* 2013 Melike *et al.* 2014 Valizadeh *et al.* 2016	The target opposition, dependency and non-cooperative features in offensive and defensive side is consistent with game model, thus formulating strategy on the basis of analyzing confrontation situations beforehand.
Strategy formulation based on machine learning control theory	Rowe *et al.* 2012 Coalbugh *et al.* 2013 Tozer *et al* 2015	This method uses machine learning to achieve the accurate perception of network status, and formulates feedback control strategy.
Strategy formulation based on evolution theory	Azab *et al.* 2011 Crouse *et al.* 2012 Bitam *et al.* 2016	Based on the diversity of genes, defensive transformation is achieved by gene mutation and recombination. In addition, configuration parameters are selected according to the environmental change.

game model with Markov decision process. An efficient iterative algorithm is designed to select optimal strategy under worst-case by abstracting strategy selection into minimum-maximum problem.

(2) Strategy Selection under Incomplete Information Assumption. Zhu *et al.* [39] describe confrontation as zero-sum and dynamic game with multistages. This model formulates defensive strategy by analyzing the increased performance overhead of targeted system and the impact on the attack implementation. Carter *et al.* [40] proposed leader-follower game model so as to describe the self-learning features of attackers under incomplete information condition. By analyzing the transformation impact on attackers and system performance consumption under static attack and self-adaptive attack conditions, respectively, it shows that maximizing platform difference increases the capability of resisting attacks effectively. In order to describe the feature of players in game following Bayes rule, Sengupta *et al.* [41] regard defenders as leaders and attackers as followers. Besides, Common Vulnerability Scoring System (CVSS) is used to quantify the set of vulnerabilities. Consequently, optimal defensive strategy is formulated. Feng *et al.* [42] proposed Bayesian Stackelberg dynamic game model to select strategy in generalized network confrontation with MTD. By comparing hidden transformation and transformation with feedback signal, it verifies that MTD can increase defense benefit by transmitting wrong signal deliberately to mislead attackers.

5.1.2. Strategy Formulation Based on Machine Learning Control Theory. Machine learning control theory [43] is a new discipline formed by the intersection of machine learning and control theory. Control theory is used in control system [44], in which controller with corrective function is adopted to control process variables to ensure the correctness of system operating. Machine learning [45] uses statistics and optimization methods to efficiently and accurately analyze the complex environment. Therefore, combining machine learning and control theory can solve the control optimization problem in the complex system. Due to the complex and distributed features of MTD deployment, machine learning control ensures the accuracy of MTD strategy selection.

Rowe *et al.* [46] proposed diversity transformation approach based on control theory. Security state assessment algorithm is adopted to analyze network security state at first. As a result, it determines triggering time. In the meantime, defense cost in different defense strategies is evaluated by quantifying implementation overhead. Therefore, it selects defensive strategy by ensuring both defensive effectiveness and low overhead. Adams *et al.* [47] compare open-loop with closed-loop defense systems; it verifies that the compensation feature of closed-loop system can reduce the input interference effectively. Besides, it shows that the multielements uncertainty will make the growth of complexity following necessary difference law. Therefore, to ensure system availability and improve system security, MTD should be unpredictable to the attackers and relatively static to legal users. In order to characterize the impact of attack implementation and network environment on MTD, Colbaugh *et al.* [48, 49] regard MTD strategy formulation as a hybrid dynamic system with the hidden mode. Strategy formulation achieves evolution by comprehensively analyzing network status changes and offensive and defensive dynamic sequences. Consequently, multiobjective reinforcement learning algorithm is designed to obtain the optimal security strategy so as to minimize attack surface and maximize configuration diversity. For the high computational complexity of optimal strategy generation, Zheng *et al.* [50] proposed a novel of a method to analyze different MTD strategies. By analyzing the defensive computational complexity in known special parameter domain and the entire parameter domain, it shows optimal defensive strategy in known special parameter domain convergences in polynomial complexity and defensive strategy formulation in remaining parameter domain convergences in an exponential complexity.

5.1.3. Strategy Formulation Based on Evolution Theory. Evolution theory [51] is a set of theories that explain the developmental variation among biological generations by genetic and observable phenomena. On the one hand, the unit of evolution is group, and genetic diversity is an important factor in evolution. On the other hand, natural selection is the major contributor to evolution. It affects the phenotype of species in its environment. Since a number of functional equivalence isomer constitutes the network executor of MTD, the components can be considered as different genomes on chromosomes. At the same time, offensive and defensive behaviors lead to the constant change of network environment. Therefore, MTD strategy selection can be considered as an evolutionary mechanism.

In order to solve "fouling" problem of chromosome pools in the genetic algorithm caused by fixed configuration, Crouse *et al.* [52] proposed strategy generation method to improve the diversity of genetic algorithm. It quantifies the weight of configuration parameter group according to the influence on system security and using time limit. Eventually, the diversity of configuration parameter is improved by updating chromosome pool, and the parameter group with the highest weight is selected as the optimal strategy. Due to the complex and changeable feature of MTD, selection of defense strategy must consider the dependency relationship and security of system components. John *et al.* [53] proposed strategy formulation architecture of MTD. It consists of configuration space exploration module based on evolutionary algorithm, transformation implementation module, and evaluation module based on expert experience. Two kinds of evolutionary algorithms are used to explore the set of configuration parameters achieving the same function and the same security goal, respectively. Consequently, the resilience of targeted system is increased by selecting optimal strategy. In terms of intelligent attacker evolving strategy, Winterrose *et al.* [54] proposed strategy selection method for diversified platform with time-varying. Malicious adversary evolution method is analyzed based on Holland genetic algorithm. As a result, investment bias measurement is adopted to measure the complexity and benefit of defense. Due to the deviations existing in strategy selection manually, Bitam *et al.* [55]

designed a kind of intelligent group algorithm so as to achieve self-learning to the network state.

After analyzing existing MTD strategy formulation studies, it can be concluded as follows: strategy formulation based on game theory formalizes objective opposition, the dependency among players, and the noncooperativeness during network confrontation process as game models. As a result, the prior decision-making is achieved on the basis of analyzing offensive and defensive confrontation situations. Moreover, since network confrontation with MTD can be divided into strategy formulation under complete information assumption and those under incomplete information assumption, how to construct game model effectively plays the key role in defensive strategy formulation process. As for strategy formulation based on machine learning control theory, it can achieve an accurate perception of network states by adopting machine learning. Besides, the defensive strategy can be formulated accurately by using control theory. Because defensive strategy can be formulated through machine learning control theory, the formulated strategy is local optimum based on current network states. However, global optimal defensive strategies are hard to formulate since this kind of method cannot deduct network confrontation with multistages and multistates. In addition, strategy formulation method based on evolution theory selects defensive strategy according to genetic evolution such as gene mutation and recombination. Due to genetic diversity feature, this kind of method can deduct the targeted system development. However, this method is difficult to formulate optimal defensive strategy under conditions of insufficient environmental variables. Consequently, MTD strategy formulation based on game theory becomes the mainstream method in strategy generation. Furthermore, on the one hand, in order to improve the accuracy of network system awareness, machine learning control theory can be adopted. On the other hand, evolution theory can be used so as to achieve targeted network system deduction.

5.2. Study of MTD Transformation Mechanisms. Transformation mechanism research is feasible MTD solutions for systems by shifting different elements to resist different security threats under various application scenarios. However, the classification at the network level is not clear in previous work. Moreover, with the emergence of collaborative transformation with multielements, transformation mechanisms can be divided into mechanisms in single-layer and in cross-layer. On that basis, the single-layer transformation mechanism is further divided into network-based transformation mechanism and node-based transformation mechanism, as shown in Table 2.

5.2.1. Single-Layer MTD Transformation Mechanism in Network. During network transmission, endpoint information, such as MAC address, IP address, port, protocol, and encryption algorithm, is used to identify the source and destination node. Forwarding path is forwarding links and routing nodes during end-to-end net-flow transmission. Therefore, endpoint information and forwarding path are two key elements in network transmission. On the other hand, as the organic part of attack surface, endpoint information and forwarding path become important network attributes to be protected. Hence, endpoint information shuffle and forwarding path migration in network-layer become the focus of current research.

(1) Endpoint Information Shuffle Mechanism. Endpoint information shuffle mechanism can be divided into real endpoint information shuffle and virtual endpoint information shuffle. The research progress is introduced as follows:

(a) Real Endpoint Information Shuffle. Kewley *et al.* [56] proposed Dynamic Network Address Translation (DYNAT). Because malicious scanning accounts for about 95% time in attack, DYNAT prevents malicious scanning by replacing TCP/IP header information. At the same time, DYNAT assures service availability by assigning predefined key parameters to legitimate users. Aimed at worm attacks, Antonatos *et al.* [57] proposed Network Address Space Randomization (NASR). The method analyzes and discriminates the potential infected endpoints at first. After that, it changes endpoint information by using DHCP protocol. In order to solve synchronization problem in IP address mutation, Jia *et al.* [58, 59] achieve that goal by opening the endpoint information before and after current hopping period. In additions, Luo *et al.* [60] proposed keyed-hashing based self-synchronization mechanism to encode the port address so as to synchronize port transformation. Besides, in order to reduce the implementation cost, Badishi *et al.* [61] achieved lightweight port hopping by constructing first-in-first-out mutation channel based on the transmission rate. Aimed at the problems of limited hopping space in IPv4 and fixed hopping period, Dunlop *et al.* [29, 62] proposed moving target defense mechanism based IPv6 (MT6D). In order to enlarge the hopping space, IPv6 address space is adopted. Besides, MT6D uses pseudo-random number to set hopping period so as to improve the randomness. To solve the problem of high performance consumption in cloud platform, Debroy *et al.* [63] proposed a method which only triggers when the probability service nodes being attacked are high. Besides, it decreases the mutation distance by constructing heterogeneous equivalence in adjacent nodes. Thereby, it ensures the defense performance and the low cost during mutation at the same time.

(b) Virtual Endpoint Information Shuffle. Jafarian *et al.* [64] proposed Random Host Mutation (RHM). In order to prevent session interruption during address hopping, it achieves virtual transformation and real-time management of endpoint information by deploying hopping gateways. In terms of the bottleneck of routing management in traditional network, literature [65] proposed OpenFlow based Random Host Mutation (OF-RHM). On the other hand, MacFarland *et al.* [66] hide the link, IP, and port numbers of endpoint by setting up DNS hopping controller so as to prevent the leakage of MAC address. Skowyra *et al.* [67] proposed network identity elimination mechanism called PHARE. It prevents MAC address leakage by randomly transforming header when packets flow out of the endpoint. Moreover,

Table 2: Classification of MTD transformation mechanisms.

Category			Transformation element	Classic mechanisms	Cyber kill-chain stage being resisted
Single-layer transformation mechanism	Transformation mechanism to network	Endpoint information shuffle	MAC address, IP address, port, protocol	DYNAT (Kewley *et al.* 2001) MT6D (Dunlop *et al.* 2011) MOTAG (Jia *et al.* 2013)	Reconnaissance, delivery, command and control, actions on targets
		Forwarding path migration	Routing nodes, forwarding link	RRM (Duan *et al.* 2013) MANET (Lu *et al.* 2015)	
	Transformation mechanism to node	Platform environment diversity	Operating system, instruction set, store address, store data	SCIT (Bangalore *et al.* 2006) MEERKATS (Keromytis *et al.* 2012) Readactor (Homesecu *et al.* 2011)	Weaponization, exploitation, installation, actions on targets
		Software application heterogenous	Instruction layout, code encoding	NOMAD (Vikram *et al.* 2013)	
Cross-layer transformation mechanism			—	APOD (Atighetchi *et al.* 2004) A3 (Pal et al. 2011) SDNA (Yackoski *et al.* 2011) MCD (Wu *et al.* 2016)	Reconnaissance, weaponization, exploitation, installation, command and control, actions on targets

Sun *et al.* [68] proposed Decoy-Enhanced Seamless IP Randomization (DESIR) to increase the unpredictability. When unauthenticated nodes access the platform, DESIR uses honeypots to observe its behavior. When the user is judged as the attacker, DESIR prevents attack by changing endpoint information of node providing service and increasing the number of honeypots deployed. In order to prevent service interruption, DESIR separates the network identifier and transmission identifier of endpoint when it migrates services, thus ensuring the continuity of service provision by reserving the transmission identifier. Since fixed hopping period is hard to maximize the unpredictability, Jafarian *et al.* [69] proposed Spatial and Temporal Random Host Mutation (ST-RHM). ST-RHM performs transformation from spatial and temporal perspective. On this basis, literature [70] proposed a novel of address transformation having the capability of perceiving malicious scanning. It uses hypothesis test to analyze scanning behavior before implementing address mutation. Lei *et al.* [71] proposed Self-Adaptive Endpoint Hopping Technique (SEHT) by dividing scanning strategy into three categories. Based on the discrimination of scanning strategy by Sibson entropy, SEHT formulates optimal hopping endpoint information and hopping frequency. Based on it, literature [72] proposed a novel of method transforming attack surface in self-adaptive way. By regarding network as directed graph, it selects virtual endpoint information whose view distance is the largest to the current one. At the same time, this method enhances the availability and scalability of endpoint hopping by adopting heuristic algorithm in mutation deployment.

Although real endpoint information shuffle improves the uncertainty of endpoint by changing endpoint information such as IP, MAC, and port randomly, this method also aggravates network system overhead at the same time. Consequently, it is hard to achieve deployment in large-scale. On the other hand, virtual endpoint information shuffle establishes the mapping relationship between real endpoint information and virtual endpoint information. As a result, it only changes endpoint information form the external viewpoint. From the internal perspective, endpoint information stays the same. Therefore, virtual endpoint shuffle method has little impact on the workflow in endpoint. Besides, fixed shuffle period is hard to resist different kinds of scanning techniques or adapt various network situations; shuffle period varying method is one of the key problems to be studied. Hence, virtual endpoint shuffle method with varying shuffle period has become one of the hotspots in MTD study.

(2) Forwarding Path Migration Mechanism. Forwarding path migration randomly selects routing nodes to change forwarding paths under the premise of ensuring reachability. Existing forwarding path migration mechanisms can be mainly divided into two types.

(a) Forwarding Path Migration Based on Deterministic Multipath Selection. Dolev *et al.* [73] divide net-flow into n parts in one session, and only less than k parts can be allowed to transmit in the same path. Therefore, n-k threshold principle is used to resist passive eavesdropping. In order to improve the diversity of forwarding paths, Aseeri *et al.* [74] designed bidirectional multipath routing algorithm. By negotiating migrating paths between source and destination, forwarding path is changed randomly during net-flow transmission. In terms of quantum communication process being easily intercepted, Safavi-Naini *et al.* [75] proposed secure quantum communication mechanism based on multipath routing. By selecting a set of forwarding paths, the mechanism randomly chooses forwarding paths to transmit net-flow divided into N parts of one session. Consequently, it increases the uncertainty of net-flow transmission.

(b) Forwarding Path Migration Based on Routing Transformation. Duan *et al.* [76] proposed Random Route Mutation (RRM) so as to cope with interception problems caused by the static of the forwarding path. RRM computes a set of routers satisfying the constraints from the aspect of routing capacity, quality of service, and required overlap rate. What is more, Jafarian *et al.* [77] proposed dynamic forwarding path selection scheme based on game theory. This method selects the optimal combination of migration period and forwarding path by describing path migration as complete information static game. In addition, Gillan *et al.* [78] proposed an active defense mechanism based on agile virtualization infrastructure to address the hidden DDoS attacks. Problem of net-flow forwarding in virtual network can be formally described as directed graph. What is more, by describing the constraints of net-flow transmission using SMT, it increases the difficulty to launch DDoS attacks while ensuring the quality of service. Wang *et al.* [79] proposed link obfuscation mechanism called Linkbait to resist link flooding attacks. Linkbait consists of link filtering, link obfuscation, and zombie detection. Link filtering selects links that might be attacked after analyzing net-flow traffic distribution. Link obfuscation lures attackers by redirecting suspicious traffic to the decoy link. Consequently, Linkbait effectively resists link flooding attacks by dynamically migrating transmission links.

Since forwarding path migration mechanism based on deterministic multipath selection transmits the fragmented net-flow on different predefined forwarding paths, it increases the difficulty of passive eavesdropping by malicious adversaries. However, due to the limited number of nonintersecting forwarding paths in the network, multipath routing mechanisms have the disadvantage of narrow transformation space. On the other hand, forwarding path migration mechanism based on routing transformation generates and transforms forwarding paths randomly during net-flow transmission. Since it has the advantage of large transformation space, it has better effectiveness in preventing passive eavesdropping than forwarding path migration mechanism based on the deterministic multipath selection.

5.2.2. Single-Layer MTD Mechanism in Node. Platform environment and software applications are important components of the nodes in network. Platform environment is the runtime environment of hardware and software, which is the basis of operation. Software applications are the code set achieving a specific function.

(1) Platform Environment Diversification Mechanism. Platform environment diversification changes elements such as address space, instruction, and system attribute in the premise of ensuring reliable operation. The specific studies are as follows:

Address space randomization (ASR), instruction set randomization (ISR), and data randomization (DR) are three typical ways to achieve platform environment diversification. ASR is a kind of transformation technique that defends address attacks by randomizing target location in memorizer. Forrest [30] and Chew *et al.* [80] summarized three kinds of address randomization method, which is stack base address or global library function entry address and stack frame offset randomization, randomization of global variable location and the internal variable offset of the stack frame, and randomization of new stack frame location. ISR prevents code injection attacks by dynamically changing instruction set. Kc *et al.* [81] and Li *et al.* [82] proposed general instruction randomization method to Linux system and Windows system, respectively. Based on Kerckhoff principle, it uses cryptographic algorithms to encrypt vulnerable system instruction. Barrantes *et al.* [83] designed a novel of random instruction set simulator. It generates a unique set of instructions for each executing program so as to increase the difficulty of external injection attacks. At the same time, when there is external code injection, the targeted process prevents attacks in interrupting execution. DR improves the dynamic of runtime data by diversifying elements such as the storage location and data format. In order to resist absolute address attacks, Cowan *et al.* [84] proposed pointer protection mechanism by doing XOR operation before the pointer is stored in memorizer. Besides, when the pointer is loaded, it is decrypted so as to ensure its availability. To address the issue of relative address attacks, Bhatkar *et al.* [85] use different expressive method to different types of data so as to prevent nearby data type corruption caused by buffer overflow of single type data.

(2) Software Application Isomerization Mechanism. This kind of mechanism changes codes dynamically to enhance the heterogeneous of software applications under the premise of ensuring functional equivalence. Depending on the application software life cycle, it can be divided into two categories:

In terms of transformation mechanism adopted during software compilation and link, Roeder *et al.* [86] create multiple execution duplicates having an equivariant function but distinct difference in structure for each software. Because there are fewer vulnerabilities in different execution duplicates, attack surface uncertainty is improved by translating application copies periodically. Aimed at buffer overflow attacks, Franz *et al.* [87] proposed reverse execution stack architecture based on multivariable executive. Because two stacks run in the opposite direction, when buffer attack occurs, it can only affect one of the stacks. What is more, the anomalous phenomenon can be effectively detected. In order to achieve large-scale software diversification, Jacknson *et al.* [88] and Homesecu *et al.* [89] randomize machine code at compile time by constructing two kinds of mutually orthogonal encoders. This mechanism monitors the compilation behavior by adopting multivariate execution environment. In terms of the limited number of heterogeneous executables in real conditions, Donnell *et al.* [90] treat program running of the same version as the same color in graph. It maximizes the unpredictability by solving the optimal graph-coloring problem. Azab *et al.* [91] proposed bionics-based software diversification for complex applications. It segments complex software into smaller tasks. After that, functional equivalence isomer is adopted according to application requirements to transform attack surface effectively.

As for transforming mechanism implemented during software load and execution, Davi *et al.* [92] proposed a novel of safe and efficient instruction-level randomization mechanism called XIFER so as to solve the problem of limited ALSR space. This method changes the relationship between code and data by randomizing executable and library files. Gupta *et al.* [93] proposed fine-grained randomization defense method called Marlin to address return-oriented programming (ROP) attacks. Marlin decomposes binary code of application into code blocks. After that, randomization algorithm is used to shuffle the code blocks. Consequently, it increases the entropy of code blocks by increasing the randomness. As for web bots, Vikram *et al.* [94] proposed NOMAD, which prevents web crawlers attacks and fake requests by randomizing the name/id of the HTML component in HTTP page. At the same time, it can increase the unpredictability by inserting fake components. Christodorescu *et al.* [95] proposed end-to-end software diversification method. It consists of static module and operating module. Static module identifies the code bit invoked by subroutines before program running. What is more, it rewrites the running components in the way of keeping semantic. Operating module guides the change of execution environment by formulating diversified strategies. As a result, it enhances software diversity by multilevel transformation.

5.2.3. Cross-Layer MTD Mechanism. Cross-layer MTD mechanism enhances defensive capability through transforming multielements collaboratively in different layers. The details are shown in Table 3.

Taython *et al.* [96, 97] proposed Net Maneuver Commander (NMC). It collects threat intelligence by deploying honeypots and predicts possible security threats after analyzing historical data. According to the algorithm diversity, geographic destination, and moving intervals, randomization algorithm is used to maneuver elements such as forwarding path, port, operating platform, and data so as to improve the flexibility of the targeted system.

Atighetchi *et al.* [98, 99] applied Applications that Participate in their Own Defense (APOD) under the funding of DARPA. At first, APOD locates the area where the attack might implement by analyzing and identifying potential security threats. Next, security strategy is formulated and decomposed into subsecurity policies sending to specific application components. Finally, system security is enhanced by collaboratively adopting mechanisms such as endpoint information hopping, network intrusion detection, and container separation.

Table 3: Cross-layer moving target defense transformation mechanism.

Name	Mechanism Characteristics					Related Technologies
	Redundant	Heterogeneous	Random	Timeliness	Deception	
NMC	√		√	√	√	Operating environment randomization, forwarding path transformation, network intrusion detection, virtual machine isolation
APOD			√	√	√	End information transformation, network intrusion detection, container separation
A3	√	√	√		√	Container-based isolation, backup, software randomization
TALENT		√	√	√		Instruction randomization, security state awareness
SDNA			√	√	√	IPv6, IP address mutation, authentication, honeypot

Pal *et al.* [100–102] proposed Advanced Adaptive Application (A3). A3 achieves proactive defense by adopting dedicated isolation container, defensive buffer, and modify and replay. To ensure the interaction of application processes in a controllable way, dedicated isolation container isolates each application process. Defensive buffer identifies potential malicious process through implementing interception, observation, analysis, processing, and conversion to key process. After that, modify and replay rolls back the key service with high security risk to its security state in the past.

Okhravi *et al.* [40, 103] proposed trusted dynamic logical heterogeneity system for key services (TALENT). TALENT uses container and portable checkpoint compiler to generate virtual runtime environment. It achieves platform heterogenous by implementing instruction randomization and operating system diversity. Besides, it maintains the continuity of critical services through preserving application state during migration. Therefore, it improves the heterogenous of the trusted platform under the promise of good quality of service.

Yackoski *et al.* [104, 105] proposed Self Shielding Dynamic Network Architecture (SDNA). SDNA consists of management node, honeypot, and key service virtual machine. In order to prevent eavesdropping, SDNA uses endpoint information hopping based on IPv6. Management node is deployed in each subnet to convert real IP into virtual IP of egress net-flows. Meanwhile, SDNA dynamically shuffles the key service nodes by using virtual machine cleaning technique.

5.3. Research of Effectiveness Evaluation Methods MTD Performance. To analyze the cost and benefit of MTD implementation, effectiveness evaluation detects the change of endpoint's state and the availability rate of vulnerabilities before and after defense. Defense cost refers to the impact of defense strategy on system availability and the consumption of system. Defense benefit refers to the impact of attackers intrusion after MTD implementation. Existing effectiveness evaluation research can be divided into four categories according to the method adopted.

(1) Empirical Evaluation Based on Offensive and Defensive Experiments. In order to verify the validity of stochastic polymorphic defense, Evans *et al.* [106] proposed effectiveness analysis method based on experiments of network confrontation. Since diversity is one of the important features of MTD, it mainly analyzes the effectiveness of diversity defense mechanisms such as address space randomization and data randomization in node. By experimenting diversity defense mechanisms preventing attack cases like circumvention attacks, it analyzes the capability of diversity defense mechanisms turning code injection or memory corruption attacks into denial of service attacks. As a result of deputy attacks and entropy reduction attacks, it can be seen that diversity defensive mechanisms are ineffective against the type of evasive attack and agent attack. In addition, it can resist scanning and brute force attacks to a certain extent. In order to conduct experiments in a more extensive way, Leeuwen *et al.* [107] proposed an architecture to measure the effectiveness of different layers of MTD mechanisms by implementing typical attacks in each phase of cyber kill-chain. It selects an appropriate experimental environment so as to ensure the fidelity of evaluation results. In order to measure the defensive benefit of MTD mechanisms, the desirable attributes, such as system weakness, system protection, and security threat knowledge, are adopted [108]. The result shows that different mechanisms in different layer can resist different phase in cyber kill-chain. What is more, in order to measure the operational cost of MTD implementation, literature [109] designed application performance monitoring (APM) and network performance monitoring (NPM). By adopting APM and NPM, the operational costs of MTD are mainly analyzed from the aspects of operating expenses, the use rate of network or endpoint node resource, and the overhead change with the change in network size. After analyzing both MTD defensive impacts on network services like DHCP and DNS and the impact of MTD approaches to traditional security methods, it can be concluded that the operational cost of MTD implementation is within affordable overhead.

(2) Empirical Evaluation Based on Simulated Experiments. By simulating different attack scenarios through OMNet ++ components, Zheng *et al.* [110] evaluated the effectiveness of IP address mutation and endpoint diversification mechanisms. After conducting 576 simulation tests, it can be seen that IP address mutation frequency can compress the attack time, and the diversity of the endpoints can increase the difficulty of attack path construction. Furthermore, since there exists optimal IP address change rate, any increase beyond the optimal value will not decrease the attack success rate. Zhuang *et al.* [20] use NeSSi2 to simulate network construction. Based on it, the attack component, defense component, and ground truth component are implemented as NeSSi2 components along with three endpoints. By simulating periodically network attack, the impact of endpoint configuration parameters change on attack success rate is analyzed. As a result, it can be concluded that with the increase of attack path length and the more protection an MTD mechanism provides, MTD implementation is much more effectiveness.

(3) Abstract Evaluation Based on Mathematical Model. Han Y *et al.* [111] proposed effectiveness analysis method based on the cyber epidemic dynamics. This method abstracts three kinds of MTD mechanisms into different cyber epidemic dynamic models, which changes either the structure or capabilities of offensive and defensive sides. By analyzing those three kinds of strategies, it shows that the conditions should be satisfied so as to achieve the optimal benefit with minimum cost. Carroll *et al.* [112] proposed address mutation evaluation method based on the *Urn* model. By comparative analysis no address mutation and perfect address mutation, it shows that the defensive efficiency is related to the space of address mutated, mutation frequency, and malicious scanning frequency. Prakash *et al.* [113] proposed an empirical analysis method based on the game theory. The method verifies that reactive MTD has higher defense benefit than proactive ones after analyzing defensive implementation under 72 kinds of network configuration, offensive and defensive costs, and benefits conditions. What is more, it shows that the capability

of reactive defense mechanism depends on the detection ability. As for the real-time characteristics of MTD, Bopche *et al.* [114] proposed dynamic network attack surface analysis method based on graph similarity measurement. It evaluates the defensive benefits by designing distance metric algorithm and distance metric algorithm.

(4) Effectiveness Evaluation Based on Mixed Analysis. Xu *et al.* [115] proposed three-layer effectiveness analysis architecture. The bottom layer is program-level state machine, which is used to capture the running status of each process. The middle layer is system-level state machine that simulates interaction among different programs. The top layer is the task-level evaluation state machine, which analyzes the defense effectiveness of different MTD combinations. Zaffarano *et al.* [116] proposed analysis method based on activity templates. It compares the effectiveness of MTD under task template and attack template to analyze the cost and benefit of MTD implementation. Cheng *et al.* [117] proposed evaluation architecture based on change-point detection. To establish the relationship between vulnerability changes and node security state transitions, hierarchical network resource is defined. Meanwhile, in order to unify the quantization result, standardized metric algorithm is designed based on CVSS. Furthermore, in order to evaluate MTD effectiveness in a more comprehensive and efficient way, how to ensure the consistency of data process and transformation during different evaluation methods above is one of the key problems to be solved. On the other hand, since the attributes of defensive cost and benefit are hard to collect and metric accurately in evaluation based on mixed analysis, automated measure architectures are proposed. In order to quantify the impact of network configuration and endpoint execution change in tests, cyber quantification framework (CQF) is designed [118]. CQF consists of experimentation configuration module, the cyber metrics processor, and the asset assignment engine. The experimentation configuration module is to specify and execute experimental configurations by providing low-level API and high-level web interface. The cyber metrics processor is to collect raw data and select computing methods during experimentation. Besides, the asset assignment engine is to provide functionality for more sophisticated analysis and comparison. Eskridge *et al.* [119] proposed a common infrastructure supporting MTD assessment called virtual infrastructure for network emulation (VINE) so as to measure the impact of MTD performance on network in more complicated offensive and defensive confrontation cases. VINE realizes the complete construction of MTD mechanism under the condition of complex network environment through developing construction, deployment, execution, and monitoring components. At the same time, VINE has the feature of scalability, high-fidelity; it can collect, measure, and evaluate the MTD performance in a more comprehensive way through changing application scenarios and the attackers' capability. Furthermore, in order to enhance the automation level of raw data collection and measurement, multitasks model automatic construction architecture is designed [120]. Based on the automatically constructed ontological system model, the measurement of resource impact on network or endpoints associated with MTD and the validation of attack vectors during execution are taken automatically. In addition, by supporting test platforms such as VINE, the applicability of proposed architecture is improved.

Shown in Table 4, after analyzing different types of MTD effectiveness evaluation methods above, it can be concluded as follows.

Although effectiveness evaluation method based on offensive and defensive experiments and effectiveness evaluation based on simulated experiments have high accuracy, the application scope of different experimental conditions is various. What is more, the empirical analysis methods based on offensive and defensive experiments are limited by the number of samples. The analysis methods based on simulated experiments are hard to simulate complex application scenarios. As a result, those two evaluation methods above are hard to evaluate different types of MTD effectiveness extensively. Furthermore, those two evaluation methods are impossible to compare evaluation results of MTD performance among different application environments. In terms of abstract evaluation based on mathematical model, the MTD performance effectiveness under different conditions can be comparatively analyzed. Besides, evaluation method based on mathematical model can greatly improve the evaluation efficiency. However, due to the fact that the features of multisteps of process, parallel tasks in network systems, and complex structure of network system are hard to characterize accurately with mathematical models, there is a deviation between evaluation results and the actual situation of MTD performance. In order to improve the efficiency of the evaluation process and enhance the accuracy of evaluation results at the same time, effectiveness evaluation method based on mixed analysis integrates the advantages of different evaluation methods above. Therefore, it has become the mainstream of existing MTD evaluation studies.

6. Application of MTD in Different Conditions

With the continuous development and maturity of MTD, it can be applied to various fields under different network architectures. This section selected some typical defense system to illustrate its operational mechanism.

(1) Application in Traditional Network. Jia *et al.* [121, 122] proposed an MTD architecture in order to protect cloud computing platform called MOTAG. MOTAG takes advantage of the redundancy of cloud platform resources. It improves the platform's dynamic feature by adopting covert double-layer proxy mechanism. As shown in Figure 6, MOTAG consists of an authentication server, proxy nodes, filter ring, and application server. Authentication server is used to authenticate the client requested to get access. If the client passes the authentication, the proxy node connects clients to the application server via filter ring. Filter ring will detect if there are abnormal events in the proxy node. If that is true, filter ring will send warning alarm to the authentication server through private channel, and the client-proxy connecting relationship is shuffled by greedy algorithm so as to locate

Table 4: Moving target defense effectiveness evaluation method classification.

Method	Classic literatures	Advantages	Disadvantages
Empirical evaluation based on offensive and defensive experiments	Evans *et al.* 2011 Leeuwen *et al.* 2016	It can be used to test defense mechanism responding to different attacks.	It is limited to the number of samples. Thus difficulty in evaluating MTD effectiveness in wide range.
Empirical evaluation based on simulated experiments	Zhuang *et al.* 2012 Zheng *et al.* 2016	It can analyze MTD effectiveness in different application scenarios and compares in vertical.	It evaluates MTD effectiveness based on prior knowledge, which is not capable of assessing MTD to unknown attacks.
Abstract evaluation based on mathematical model	Clark *et al.* 2013 Han *et al.* 2014 Carroll *et al.* 2014	It has the advantage of universality, thus being able to compare different MTD mechanisms.	Since it is difficult to abstract complicated environment and tasks in parallel, it may cause the deviation between constructed model and the actual one.
Effectiveness evaluation based on mixed analysis	Okhravi *et al.* 2014 Zaffarano *et al.* 2015 Eskridge *et al.* 2015 Lei *et al.* 2016	It combines the advantages of the above analysis methods.	It needs to ensure the consistency of data process and transformation using different methods.

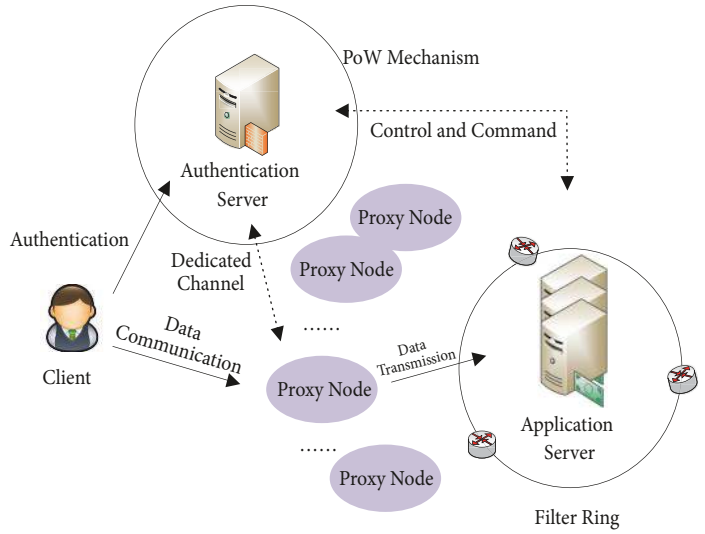


Figure 6: MOTAG system architecture.

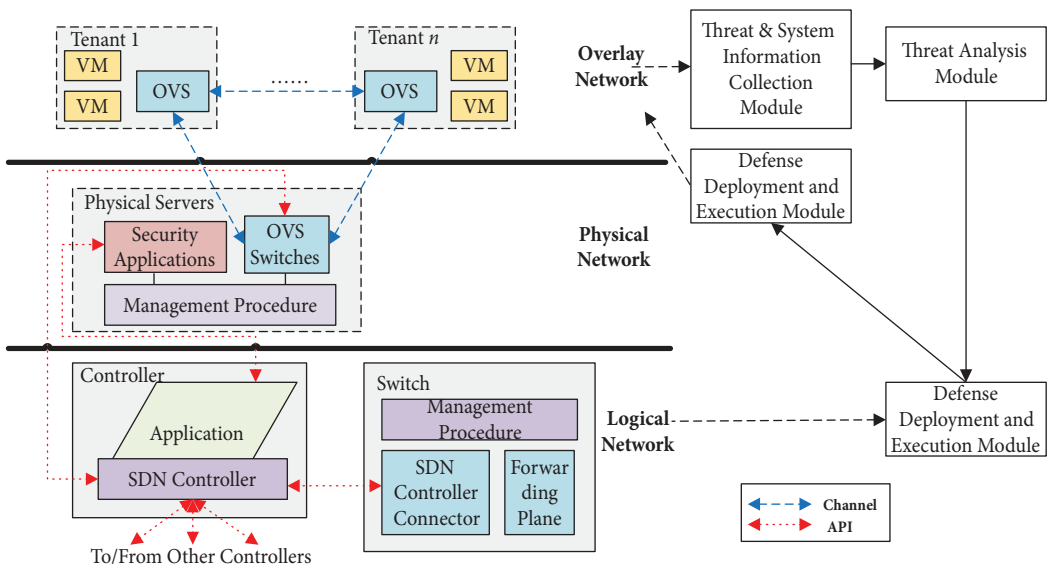


Figure 7: Extensible MTD architecture in SDN.

the suspicious client. In addition, MOTAG employs a Proof-of-Work mechanism to prevent flood attacks to the authentication server. Furthermore, in order to deploy in conditions not capable of authenticating client users, it also protects key services in cloud platform by constructing cloud replication mutation mechanism [123].

(2) Application in Software Defined Network. To solve the problem of policy conflict of cloud platform transformation in SDN, Chowdhary *et al.* [124, 125] proposed a novel of MTD architecture. It takes advantage of the centralized control of the controller to improve the effectiveness of defense by detecting and solving policy conflicts. As shown in Figure 7, this system perceives threats and implements transformation in real-time by constructing security threat detection, strategy analysis, and defense implementation closed-loop. The security threat and system states information collection module sends the collected vulnerability information to security threat analysis module, in which attack graph is adopted to process and analyze the potential attacks. After that, candidate strategies are generated based on connectivity. Defensive deployment and execution module chooses one of the strategies and sends it to policy conflict detection and resolution module to check. If there is no policy conflict, the chosen defensive strategy is delivered to the defensive deployment and the execution module to implement.

(3) Application in Ipv6 Network. Heydari *et al.* [126, 127] proposed Mobile IPv6 based moving target defense (MI-MTD). In terms of the problem that web services are easily blocked, MI-MTD changes the address and user groups by taking advantage of the large address space and coexistence of multiple addresses feature in IPv6. As shown in Figure 8, MI-MTD is mainly composed of three parts. Access authentication module is to verify the user identity to access through

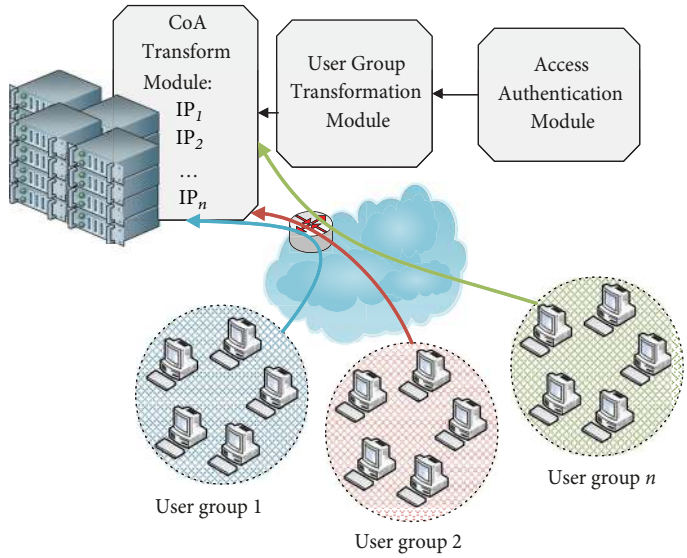


Figure 8: MI-MTD system structure.

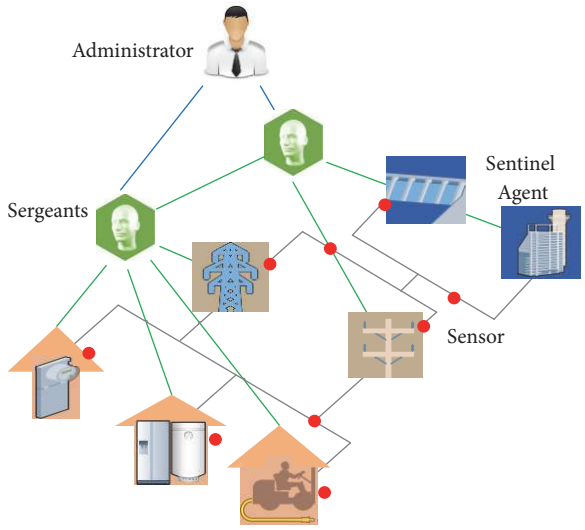


Figure 9: ABCD system structure.

IPsec. When the user is authenticated, it will be assigned to a group with specific Care of Address, in which is to get access to the server. Meanwhile, MI-MTD builds mapping relationship between Home of Address and CoA so as to ensure session continuity.

(4) Application in Internet of Things. Fink *et al.* [128] proposed Ant-Based Cyber Defense (ABCD). In order to cope with the complex structure of the smart grid and the high cost of defensive transformation, sensor (Digital Ant) with features of comprehensively sensing and real-time transmission is utilized. As shown in Figure 9, ABCD consists of sergeants, sentinel agents, and sensors. Among them, sergeants formulate defense strategy by receiving the feedback of high-level enclave agent, which interacts with the administrator. Sentinel agents are adopted to perceive security situation of the protected system. Based on it, they send the integrated state information to sergeants. Sensors are used to observe and collect different indicators of protected system. Therefore, by constructing sergeants-sentinel agent-sensor layered architecture, ABCD can perceive the overall awareness of smart grid security status and implement real-time defense.

7. Conclusion and Future Direction

Although the research achievements are considerable in various fields of MTD application, there are still many problems remaining unsolved. At the same time, the continuous emergence of new techniques and the interdisciplinarity of different disciplines also provide new directions in the development and idea of MTD study.

(1) The Evolution of MTD Architecture and Mode. Existing MTD architecture has already evolved from proactive defense architecture into reactive defense architecture. At the same time, different mechanisms have been combined so as to enhance defensive effectiveness. Although existing research of MTD has been made some progress, there are still two problems urgently to be addressed in this field:

(i) As the attack means become more and more convert and intelligent, attackers already have the capability of self-learning and evading defensive mechanisms. They can break defensive border and establish fortified point in the protected network. Aimed at the bottleneck in denial type MTD mechanism, deceptive type MTD has brought a new idea [129]. It provides misleading information by actively exposing real and fake intelligence of the protected system. Furthermore, attack behavior will develop towards the direction which is in favor of defenders. As a result, the following study of MTD architecture can try to increase the deceitfulness in the defensive mechanism so as to achieve more effective defense.

(ii) On the other hand, the combination of the defensive mechanism will greatly increase the complexity of defense system and the cost of defense implementation. In recent years, network service chain has brought enlightenment for solving high cost of the combined defense mode. Customized defense chain [130] can be formulated not only based on the targeted system importance and security state but also based on the security goal and its environment. Therefore, it can deploy more effective defensive combination with lower cost.

(2) The Key Techniques in Moving Target Defense. In the study of defensive strategy formulation, the key factor is the construction of offensive and defensive model [37]. Although existing research can describe network confrontation under known attack condition accurately, there are still problems under unknown attack condition:

(i) Whether the features chosen can effectively describe the unknown type attack behavior is not known, which will dramatically affect the effectiveness of defensive strategy selected. Therefore, the correctness of description to unknown attacks and the accuracy of model construction are urgent problems to be solved.

In terms of defensive mechanism study, the key factors are coverage of transformation elements, unpredictability of transformation elements, and the timeliness of transformation frequency [26]. If the intersection between the selected set of transformation elements and the exploited set of attackers is empty, or the space of transformation elements is limited, the implementation of MTD will lose efficacy soon. Consequently, transformation elements in existing studies gradually evolve from single element to multielements. What is more, it also extends from a single-layer to cross-layer transformation [99]. On the other hand, the triggering moment of defense also greatly affects the effectiveness of mechanism implementation. However, there are still some shortcomings in the study of defensive mechanism:

(i) As for the coverage of transformation elements, there are few studies to explore "collaborative" problem among multielements and "getting rid of coherence" cross different layers. The so-called "collaborative" problem refers to the collision problem caused by multielements transformation, which may result in defensive failure. "Getting rid of coherence" problem refers to pattern problems, in which attackers may find transformation pattern in other layers when analyzing transformation pattern in some layers.

(ii) For the unpredictability of transformation element space, existing functional equivalence isomer such as diversified compiler and multiversion software may have some common vulnerabilities. Attackers can find out such vulnerabilities in common so as to bypass MTD mechanisms [131]. Therefore, how to increase the heterogeneity degree of different functional equivalence isomers is one of the directions in future studies.

(iii) For transformation triggering moment of MTD, although defensive mechanism based on stochastic mutation period can increase the difficulty of attackers to exhaustively guess the exact mutation period, it is hard to effectively balance the defensive benefit and cost. As a result, how to adjust the transformation period dynamically to minimize the defensive cost under the condition of ensuring the effectiveness of defense implementation remains a pressing issue to be solved.

In terms of effectiveness evaluation study, the key factors lie in the reliability of measurement process and the comparability of results among different defensive mechanisms [115]. Although research of effective evaluation has gradually changed from the singly method to hybrid analysis method, there are still two important aspects which should be strengthened:

(i) Existing assessment indicators and evaluation methods are mainly to quantify and evaluate the potential benefit in network confrontation process. With the means of attack becoming more subtle, existing assessment methods are not capable of measuring and evaluating the covertness factors, especially in the offensive side during network confrontation.

(ii) Since the deployment environment of MTD mechanisms is different, the quantitative criteria of offensive and defensive using vulnerabilities need to be unified, which is one of the challenges in current research [116].

(3) Application of Moving Target Defense Mechanism. The applications of MTD should consider factors such as the manageability of mechanism, the low-performance consumption, and scalability of deployment. Although existing system application study improves the efficiency by adopting the automated configuration and solves deployment optimization problems by using satisfiable module theory, multiobjective optimization, there are still some problems as follows [65]:

(i) Automated configuration can greatly improve the efficiency of MTD deployment. However, with the increase of complexity and enlargement of defensive system, configuration failure may occur due to policy conflict during the process of automated configuration. Therefore, the correctness and effectiveness of automated configuration method should be the focus in the following research.

(ii) Many solutions to coping with deployment optimization are N-P problems. Approximate solution by loosening the constraints or using heuristic method may not be used for practical applications. Therefore, how to design more efficient solution algorithms for different problems to ensure the accuracy and effectiveness in deployment optimization still needs to be solved.

As an active defensive means changing game rules in network confrontation, MTD is trying to reverse the attack and defense asymmetry by continuously innovating defensive architecture, deepening transformation mechanism study, and optimizing the application and deployment of MTD. Literature analysis method is used to summarize typical works of MTD literature in the last decade. The inspiration of MTD is explained. What is more, the related theory study and key techniques are summarized. Finally, challenges and future directions in this field are discussed to provide a reference for further research.

Conflicts of Interest

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

References

[1] H. Zhang, W. Han, X. Lai, D. Lin, J. Ma, and J. Li, "Survey on cyberspace security," *Sciene Sinica Informationis*, vol. 46, no. 2, pp. 125–164, 2016.

[2] X. JinPing, "Overall layout and planning all parties to strive to innovate and develop China into a strong cyberpower," *People's Daily*, pp. 2–28, 2014.

[3] M. Conti, T. Dargahi, and A. Dehghantanha, "Cyber threat intelligence: challenges and opportunities," in *Cyber Threat Intelligence*, vol. 70 of *Advances in Information Security*, pp. 1–6, Springer International Publishing, Cham, 2018.

[4] S. Jajodia et al., *Moving Target Defense: Creating Asymmetric Uncertainty for Cyber Threats*, Springer Science & Business Media, 2011.

[5] D. Kramer and W. Karl, "Realizing a proactive, self-optimizing system behavior within adaptive, heterogeneous many-core architectures," in *Proceedings of the 2012 IEEE 6th International Conference on Self-Adaptive and Self-Organizing Systems, SASO 2012*, pp. 39–48, France, September 2012.

[6] R. Nee and R. Prasad, *OFDM for Wireless Multimedia Communications*, Artech House, Inc, 2000.

[7] H. Zhang G, L. C. Li, and M. Tang, "Capability of evolutionary cryptosystems against differential cryptanalysis," *SCIENTIA SINICA Informationis*, vol. 43, no. 4, pp. 545–554, 2013.

[8] H. Okhravi, W. Streilein, and K. S. Bauer, *Moving Target Techniques: Leveraging Uncertainty for Cyber Defense*, MIT Lincoln Laboratory Lexington United States, 2015.

[9] W. Jiangxing, "Research on cyber mimic defense," *Journal of Cyber Security*, vol. 4, pp. 1–10, 2016.

[10] W. Xiao, X.-Y. Chen, and Y.-B. Bao, "Review of research on reconfigurable information security system," *Tien Tzu Hsueh Pao/Acta Electronica Sinica*, vol. 45, no. 5, pp. 1240–1248, 2017.

[11] "National cyber leap year summit 2009 co-chirs' report [EB/OL]," https://www.nitrd.gov/nitrdgroups/index.php?title=National_Cyber_Leap_Year.

[12] "Cybersecurity game-change research & development recommendations [EB/OL]," http://www.nitrd.gov/pubs/CSIA_IWG_%20Cybersecurity_%20GameChange_RD_%20Recommendations_20100513.pdf.

[13] P. K. Manadhata and J. M. Wing, "An attack surface metric," *IEEE Transactions on Software Engineering*, vol. 37, no. 3, pp. 371–386, 2011.

[14] P. K. Manadhata, "Game theoretic approaches to attack surface shifting," in *Moving Target Defense II*, vol. 100 of *Advances in Information Security*, pp. 1–13, Springer, New York, NY, USA, 2013.

[15] Y. Huang and A. K. Ghosh, "Introducing diversity and uncertainty to create moving attack surfaces for web services," in *Moving Target Defense*, vol. 54 of *Advances in Information Security*, pp. 131–151, Springer, New York, NY, USA, 2011.

[16] R. Zhuang, S. A. DeLoach, and X. Ou, "Towards a theory of moving target defense," in *Proceedings of the 1st ACM Workshop on Moving Target Defense (MTD '14)—Co-located with 21st ACM Conference on Computer and Communications Security (CCS '14)*, pp. 31–40, Scottsdale, Ariz, USA, November 2014.

[17] R. Zhuang, A. G. Bardas, S. A. DeLoach, and X. Ou, "A theory of cyber attacks: a step towards analyzing mtd systems," in *Proceedings of the 2nd ACM Workshop on Moving Target Defense, MTD 2015*, pp. 11–20, USA, 2015.

[18] P. Ammann, D. Wijesekera, and S. Kaushik, "Scalable, graph-based network vulnerability analysis," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pp. 217–224, ACM, Washington, DC, USA, November 2002.

[19] N. Ghosh and S. Ghosh, "An approach for security assessment of network configurations using attack graph," in *Proceedings of the 2009 First International Conference on Networks & Communications*, pp. 283–288, Chennai, India, December 2009.

[20] R. Zhuang, S. Zhang, and A. S. DeLoach, "Simulation-based approaches to studying effectiveness of moving-target network defense," in *Proceedings of the National symposium on moving target research*, pp. 1–12, 2012.

[21] S. A. DeLoach, X. Ou, R. Zhuang, and S. Zhang, "Model-driven, moving-target defense for enterprise network security," in *Models@run.time*, vol. 8378 of *Lecture Notes in Computer Science*, pp. 137–161, Springer International Publishing, Cham, 2014.

[22] R. Zhuang, S. A. DeLoach, and X. Ou, "A model for analyzing the effect of moving target defenses on enterprise networks," in *Proceedings of the 9th Annual Cyber and Information Security Research Conference (CISRC '14)*, pp. 73–76, April 2014.

[23] J. B. Hong and D. S. Kim, "Assessing the effectiveness of moving target defenses using security models," *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 2, pp. 163–177, 2016.

[24] J. R. Hamlet and C. C. Lamb, "Dependency graph analysis and moving target defense selection," in *Proceedings of the 2016 ACM Workshop on Moving Target Defense, MTD 2016*, pp. 105–116, Austria.

[25] T. Hobson, H. Okhravi, D. Bigelow, R. Rudd, and W. Streilein, "On the challenges of effective movement," in *Proceedings of the the First ACM Workshop*, pp. 41–50, Scottsdale, Arizona, USA, November 2014.

[26] H. Okhravi, T. Hobson, D. Bigelow, and W. Streilein, "Finding focus in the blur of moving-target techniques," *IEEE Security & Privacy*, vol. 12, no. 2, pp. 16–26, 2014.

[27] H. Okhravi and H. Shrobe, *Moving Target Techniques: Cyber Resilience Throught Randomization, Diversity, and Dynamism*, Massachusetts Inst. of Tech. Lexington United States, 2017.

[28] M. Green, D. C. MacFarland, D. R. Smestad, and C. A. Shue, "Characterizing network-based moving target defenses," in *Proceedings of the 2nd ACM Workshop on Moving Target Defense, MTD 2015*, pp. 31–35, USA, 2015.

[29] M. Dunlop, S. Groat, W. Urbanski, R. Marchany, and J. Tront, "MT6D: a moving target IPv6 defense," in *Proceedings of the Military Communications Conference (MILCOM '11)*, pp. 1321–1326, IEEE, Baltimore, Md, USA, November 2011.

[30] S. Forrest, A. Somayaji, and D. H. Ackley, "Building diverse computer systems," in *Proceedings of the 1997 6th Workshop on Hot Topics in Operating Systems, HOTOS*, pp. 67–72, May 1997.

[31] R. Zhuang, S. Zhang, A. Bardas, S. A. DeLoach, X. Ou, and A. Singhal, "Investigating the application of moving target defenses to network security," in *Proceedings of the 2013 6th International Symposium on Resilient Control Systems, ISRCS 2013*, pp. 162–169, San Francisco, Calif, USA, August 2013.

[32] H. Zhou, C. Wu, M. Jiang et al., "Evolving defense mechanism for future network security," *IEEE Communications Magazine*, vol. 53, no. 4, pp. 45–51, 2015.

[33] M. Carvalho, T. C. Eskridge, K. Ferguson-Walter, and N. Paltzer, "MIRA: a support infrastructure for cyber command and control operations," in *Proceedings of the Resilience Week, RSW 2015*, pp. 102–107, USA, August 2015.

[34] M. Carvalho, J. M. Bradshaw, L. Bunch et al., "Command and control requirements for moving-target defense," *IEEE Intelligent Systems*, vol. 27, no. 3, pp. 79–85, 2012.

[35] X. Liang and Y. Xiao, "Game theory for network security," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 472–486, 2013.

[36] H. Maleki, S. Valizadeh, W. Koch et al., "Markov modeling of moving target defense games," *Journal of Cryptology*, pp. 47–83, 2016.

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

[38] X. Feng, Z. Zheng, P. Mohapatra, and D. Cansever, "A stackelberg game and markov modeling of moving target defense," in *Proceedings of the International Conference on Decision and Game Theory for Security*, vol. 10575, pp. 315–335, Springer International Publishing.

[39] Q. Zhu and T. Başar, "Game-theoretic approach to feedback-driven multi-stage moving target defense," in *Proceedings of the International Conference on Decision and Game Theory for Security*, vol. 8252, pp. 246–263, Springer International Publishing, 2013.

[40] M. K. Carter, F. J. Riordan, and H. Okhravi, "A game theoretic approach to strategy determination for dynamic platform defenses," in *Proceedings of the First ACM Workshop on Moving Target Defense*, pp. 21–30, 2014.

[41] S. Sengupta, G. S. Vadlamudi, S. Kambhampati et al., "A game theoretic approach to strategy generation for moving target defense in web applications," in *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, pp. 178–186, 2017.

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

[43] P. J. Fleming and R. C. Purshouse, "Evolutionary algorithms in control systems engineering: a survey," *Control Engineering Practice*, vol. 10, no. 11, pp. 1223–1241, 2002.

[44] J. Zabczyk, *Mathematical Control Theory: An Introduction*, Springer Science & Business Media, Boston, Mass, USA, 2009.

[45] E. Alpaydin, *Introduction to Machine Learning*, MIT press, 2014.

[46] J. Rowe, N. K. Levitt, and T. Demir, "Artificial diversity as maneuvers in a control theoretic moving target defense," in *Proceedings of the National Symposium on Moving Target Research*, 2012.

[47] D. M. Adams, D. S. Hitefield, and B. Hoy, "Application of cybernetics and control theory for a new paradigm in cybersecurity," *Cryptography and Security*, 2013, arXiv:1311.0257.

[48] R. Colbaugh and K. Glass, "Predictability-oriented defense against adaptive adversaries," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC '12)*, pp. 2721–2727, October 2012.

[49] R. Colbaugh and K. Glass, *Predictive Moving Target Defense*, United States,, Sandia National Laboratories (SNL-NM), Albuquerque, NM, USA, 2012.

[50] R. Zheng, W. Lu, and S. Xu, "Preventive and reactive cyber defense dynamics is globally stable," *IEEE Transactions on Network Science and Engineering*, 2017.

[51] E. Mayr and W. B. Provine, *The Evolutionary Synthesis: Perspectives on the Unification of Biology*, Harvard University Press, 1998.

[52] M. Crouse, W. E. Fulp, and D. Canas, "Improving the diversity defense of genetic algorithm-based moving target approaches," in *Proceedings of the National Symposium on Moving Target Research*, 2012.

[53] D. J. John, R. W. Smith, W. H. Turkett, D. A. Cañas, and E. W. Fulp, "Evolutionary based moving target cyber defense," in *Proceedings of the 16th Genetic and Evolutionary Computation Conference, GECCO 2014*, pp. 1261–1268, Canada, July 2014.

[54] L. M. Winterrose and K. M. Carter, "Strategic evolution of adversaries against temporal platform diversity active cyber defenses," in *Proceedings of the 2014 Symposium on Agent Directed Simulation*, 9 pages, 2014.

[55] S. Bitam, S. Zeadally, and A. Mellouk, "Bio-inspired cybersecurity for wireless sensor networks," *IEEE Communications Magazine*, vol. 54, no. 6, pp. 68–74, 2016.

[56] D. Kewley, R. Fink, J. Lowry, and M. Dean, "Dynamic approaches to thwart adversary intelligence gathering," in *Proceedings of the DARPA Information Survivability Conference and Exposition II, DISCEX 2001*, pp. 176–185, Anaheim, Calif, USA, June 2001.

[57] S. Antonatos, P. Akritidis, E. P. Markatos, and K. G. Anagnostakis, "Defending against hitlist worms using network address space randomization," *Computer Networks*, vol. 51, no. 12, pp. 3471–3490, 2007.

[58] L.-Y. Shi, C.-F. Jia, and S.-W. Lu, "Research on end hopping for active network confrontation," *Tongxin Xuebao/Journal on Communication*, vol. 29, no. 2, pp. 106–110, 2008.

[59] K. Lin, C.-F. Jia, and L.-Y. Shi, "Improvement of distributed timestamp synchronization," *Tongxin Xuebao/Journal on Communication*, vol. 33, no. 10, pp. 110–116, 2012.

[60] Y.-B. Luo, B.-S. Wang, X.-F. Wang, and B.-F. Zhang, "A keyed-hashing based self-synchronization mechanism for port address hopping communication," *Frontiers of Information Technology and Electronic Engineering*, vol. 18, no. 5, pp. 719–728, 2017.

[61] G. Badishi, A. Herzberg, and I. Keidar, "Keeping denial-of-service attackers in the dark," *IEEE Transactions on Dependable and Secure Computing*, vol. 4, no. 3, pp. 191–204, 2007.

[62] M. Dunlop, S. Groat, W. Urbanski, R. Marchany, and J. Tront, "The blind Man's bluff approach to security using IPv6," *IEEE Security & Privacy*, vol. 10, no. 4, pp. 35–43, 2012.

[63] S. Debroy, P. Calyam, M. Nguyen, A. Stage, and V. Georgiev, "Frequency-minimal moving target defense using software-defined networking," in *Proceedings of the 2016 International Conference on Computing, Networking and Communications (ICNC)*, pp. 1–6, Kauai, HI, USA, Feburary 2016.

[64] E Al-Shaer, Q. Duan, and J. H. Jafarian, "Random host mutation for moving target defense," in *Proceeding of the SecureComm*, pp. 310–327, 2012.

[65] J. H. Jafarian, E. Al-Shaer, and Q. Duan, "Openflow random host mutation: transparent moving target defense using software defined networking," in *Proceedings of the 1st Workshop on Hot Topics in Software Defined Networks (HotSDN '12)*, pp. 127–132, ACM, Helsinki, Finland, August 2012.

[66] D. C. MacFarland and C. A. Shue, "The SDN shuffle: creating a moving-target defense using host-based software-defined networking," in *Proceedings of the 2nd ACM Workshop on Moving Target Defense, MTD 2015*, pp. 37–41, USA, 2015.

[67] R. Skowyra, K. Bauer, V. Dedhia, and H. Okhravi, "Have No PHEAR: networks without identifiers," in *Proceedings of the 2016 ACM Workshop on Moving Target Defense, MTD 2016*, pp. 3–14, Austria, 2016.

[68] J. Sun and K. Sun, "DESIR: decoy-enhanced seamless IP randomization," in *Proceedings of the 35th Annual IEEE International Conference on Computer Communications, IEEE INFOCOM 2016*, pp. 1–9, April 2016.

[69] J. H. H. Jafarian, E. Al-Shaer, and Q. Duan, "Spatio-temporal address mutation for proactive cyber agility against sophisticated attackers," in *Proceedings of the 1st ACM Workshop on Moving Target Defense (MTD '14)*, pp. 69–78, Scottsdale, AZ, USA, November 2014.

[70] J. H. Jafarian, E. Al-Shaer, and Q. Duan, "Adversary-aware IP address randomization for proactive agility against sophisticated attackers," in *Proceedings of the 34th IEEE Annual Conference on Computer Communications and Networks, IEEE INFOCOM 2015*, pp. 738–746, Hong Kong, May 2015.

[71] C. Lei, H.-Q. Zhang, D.-H. Ma, and Y.-J. Yang, "Network moving target defense technique based on self-adaptive end-point hopping," *Arabian Journal for Science and Engineering*, vol. 42, no. 8, pp. 3249–3262, 2017.

[72] L. Cheng, M. Duo-He, Z. HongQi, Y. YingJie, and W. Li-Ming, "Moving target defense technique based on network attack surface self-adaptive mutation," *Chinese Journal of Computers*, vol. 40, no. 130, 2017.

[73] S. Dolev and S. T. David, "SDN-based private interconnection," in *Proceedings of the 2014 13th IEEE International Symposium on Network Computing and Applications, NCA 2014*, pp. 129–136, USA, August 2014.

[74] A. Aseeri, N. Netjinda, and R. Hewett, "Alleviating eavesdropping attacks in software-defined networking data plane," in *Proceedings of the 12th Annual Cyber and Information Security Research Conference, CISRC 2017*, USA, April 2017.

[75] R. Safavi-Naini, A. Poostindouz, and V. Lisy, "Path hopping," in *Proceedings of the the 2017 Workshop*, pp. 111–114, Dallas, TX, USA, October 2017.

[76] Q. Duan, E. Al-Shaer, and H. Jafarian, "Efficient random route mutation considering flow and network constraints," in *Proceedings of the 1st IEEE International Conference on Communications and Network Security (CNS '13)*, pp. 260–268, October 2013.

[77] J. Jafarian, E. Al-Shaer, and Q. Duan, "Formal approach for route agility against persistent attackers," in *Computer Security—ESORICS 2013*, J. Crampton, S. Jajodia, and K. Mayes, Eds., vol. 8134 of *Lecture Notes in Computer Science*, pp. 237–254, Springer, Berlin, Germany, 2013.

[78] F. Gillani, E. Al-Shaer, S. Lo, Q. Duan, M. Ammar, and E. Zegura, "Agile virtualized infrastructure to proactively defend against cyber attacks," in *Proceedings of the 34th IEEE Annual Conference on Computer Communications and Networks, IEEE INFOCOM 2015*, pp. 729–737, May 2015.

[79] Q. Wang, F. Xiao, M. Zhou et al., "Mitigating link-flooding attacks with active link obfuscation," *Networking and Internet Architecture*, 2017, arXiv:1703.09521.

[80] M. Chew and D. Song, "Mitigating buffer overflows by operating system randomization," Technical Report CMU-CS-02-197, 2002.

[81] G. S. Kc, A. D. Keromytis, and V. Prevelakis, "Countering code-injection attacks with instruction-set randomization," in *Proceedings of the 10th ACM Conference on Computer and Communications Security, CCS 2003*, pp. 272–280, USA, October 2003.

[82] L. Li, J. Just, and R. Sekar, "Address-space randomization for windows systems," in *Proceedings of the 2006 22nd Annual Computer Security Applications Conference (ACSAC'06)*, pp. 329–338, Miami Beach, FL, USA, December 2006.

[83] E. G. Barrantes, T. S. Palmer, D. H. Ackley, D. Stefanović, S. Forrest, and D. D. Zovi, "Randomized instruction set emulation to disrupt binary code injection attacks," in *Proceedings of the 10th ACM Conference on Computer and Communications Security, CCS 2003*, pp. 281–289, USA, October 2003.

[84] C. Cowan, S. Beattie, and J. Johansen, "Pointguard TM: protecting pointers from buffer overflow vulnerabilities," in *Proceedings of the 12th conference on USENIX Security Symposium*, vol. 12, pp. 91–104, 2003.

[85] S. Bhatkar and R. Sekar, "Data space randomization," in *Proceedings of the DIMVA*, pp. 1–22, 2008.

[86] T. Roeder and F. B. Schneider, "Proactive obfuscation," *ACM Transactions on Computer Systems*, vol. 28, no. 2, 2010.

[87] B. Salamat, A. Gal, and M. Franz, "Reverse stack execution in a multi-variant execution environment," in *Proceedings of the Workshop on Compiler and Architectural Techniques for Application Reliability and Security*, pp. 1–7, 2008.

[88] T. Jackson, C. Wimmer, and M. Franz, "Multi-variant program execution for vulnerability detection and analysis," in *Proceedings of the the Sixth Annual Workshop*, pp. 1–38, Oak Ridge, Tennessee, April 2010.

[89] T. Jackson, A. Homescu, S. Crane, P. Larsen, S. Brunthaler, and M. Franz, "Diversifying the software stack using randomized NOP insertion," in *Moving Target Defense II*, vol. 100 of *Advances in Information Security*, pp. 151–173, Springer, New York, NY, USA, 2013.

[90] A. J. O'Donnell and H. Sethu, "On achieving software diversity for improved network security using distributed coloring algorithms," in *Proceedings of the 11th ACM Conference on Computer and Communications Security, CCS 2004*, pp. 121–131, USA, October 2004.

[91] M. Azab, R. Hassan, and M. Eltoweissy, "ChameleonSoft: a moving target defense system," in *Proceedings of the 7th International Conference on Collaborative Computing: Networking, Applications and Worksharing, ColiaborateCom 2011*, pp. 241–250, USA, October 2011.

[92] L. V. Davi, A. Dmitrienko, S. Nürnberger, and A.-R. Sadeghi, "Gadge me if you can: secure and efficient ad-hoc instruction-level randomization for x86 and ARM," in *Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security, ASIA CCS 2013*, pp. 299–310, China, May 2013.

[93] A. Gupta, S. Kerr, M. S. Kirkpatrick, and E. Bertino, "Marlin: a fine grained randomization approach to defend against ROP attacks," in *Proceedings of the International Conference on Network and System Security*, vol. 7873, pp. 293–306, Springer Berlin Heidelberg, 2013.

[94] S. Vikram, C. Yang, and G. Gu, "NOMAD: towards non-intrusive moving-target defense against web bots," in *Proceedings of the 1st IEEE International Conference on Communications and Network Security, CNS 2013*, pp. 55–63, USA, October 2013.

[95] M. Christodorescu, M. Fredrikson, S. Jha, and J. Giffin, "End-to-end software diversification of internet services," in *Moving Target Defense*, vol. 54 of *Advances in Information Security*, pp. 117–130, Springer, New York, NY, USA, 2011.

[96] P. Beraud, A. Cruz, S. Hassell, and S. Meadows, "Using cyber maneuver to improve network resiliency," in *Proceedings of the 2011 IEEE Military Communications Conference, MILCOM 2011*, pp. 1121–1126, USA, November 2011.

[97] P. Beraud, A. Cruz, S. Hassell, J. Sandoval, and J. J. Wiley, "Cyber defense network maneuver commander," in *Proceedings of the 44th Annual 2010 IEEE International Carnahan Conference on Security Technology, ICCST 2010*, pp. 112–120, USA, October 2010.

[98] M. Atighetchi, P. Pal, F. Webber, R. Schantz, C. Jones, and J. Loyall, "Adaptive cyberdefense for survival and intrusion tolerance," *IEEE Internet Computing*, vol. 8, no. 6, pp. 25–33, 2004.

[99] W. Nelson, W. Farrell, M. Atighetchi et al., "APOD experiment 2: final report," *BBN Technologies LLC, Technical Memorandum*, vol. 1326, 2002.

[100] P. Pal, R. Schantz, A. Paulos, J. Regehr, and M. Hibler, "Advanced adaptive application (A3) environment: initial experience," in *Proceedings of the Middleware 2011 Industry Track: Part of the ACM/IFIP/USENIX International Middleware Conference, Middleware'11*, 5 pages, Portugal, December 2011.

[101] A. Paulos, P. Pal, R. Schantz, and B. Benyo, "Moving target defense (MTD) in an adaptive execution environment," in *Proceedings of the the Eighth Annual Cyber Security and Information Intelligence Research Workshop*, Oak Ridge, Tenn, USA, January 2013.

[102] P. Pal, R. Schantz, A. Paulos, and B. Benyo, "Managed execution environment as a moving-target defense infrastructure," *IEEE Security & Privacy*, vol. 12, no. 2, pp. 51–59, 2014.

[103] H. Okhravi, I. E. Robinson, S. Yannalfo et al., "Talent: dynamic platform heterogeneity for cyber survivability of mission critical applications," in *Proceedings of the Secure and Resilient Cyber Architecture Conference (SRCA'10)*, 2010.

[104] J. Yackoski, P. Xie, H. Bullen, J. Li, and K. Sun, "A self-shielding dynamic network architecture," in *Proceedings of the 2011 IEEE Military Communications Conference, MILCOM 2011*, pp. 1381–1386, USA, November 2011.

[105] J. Yackoski, H. Bullen, X. Yu, and J. Li, "Applying self-shielding dynamics to the network architecture," in *Moving Target Defense II*, vol. 100 of *Advances in Information Security*, pp. 97–115, Springer, New York, NY, USA, 2013.

[106] D. Evans, A. Nguyen-Tuong, and J. Knight, "Effectiveness of moving target defenses," in *Moving Target Defense*, vol. 54 of *Advances in Information Security*, pp. 29–48, Springer, New York, NY, USA, 2011.

[107] B. Van Leeuwen, W. Stout, and V. Urias, "MTD assessment framework with cyber attack modeling," in *Proceedings of the 50th Annual IEEE International Carnahan Conference on Security Technology, ICCST 2016*, USA, October 2016.

[108] M. Torgerson, *Security Metrics for Communication Systems*, 12th ICCRTS, Newport, RI, USA, 2007.

[109] B. Van Leeuwen, W. M. S. Stout, and V. Urias, "Operational cost of deploying moving target defenses defensive work factors," in *Proceedings of the 34th Annual IEEE Military Communications Conference, MILCOM 2015*, pp. 966–971, USA, October 2015.

[110] J. Zheng and A. S. Namin, "The impact of address changes and host diversity on the effectiveness of moving target defense strategy," in *Proceedings of the 2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC)*, pp. 553–558, Atlanta, GA, USA, June 2016.

[111] Y. Han, W. Lu, and S. Xu, "Characterizing the power of moving target defense via cyber epidemic dynamics," in *Proceedings of the the 2014 Symposium and Bootcamp*, pp. 1–12, Raleigh, North Carolina, April 2014.

[112] T. E. Carroll, M. Crouse, E. W. Fulp, and K. S. Berenhaut, "Analysis of network address shuffling as a moving target defense," in *Proceedings of the 1st IEEE International Conference on Communications (ICC '14)*, pp. 701–706, IEEE, Sydney, Australia, June 2014.

[113] A. Prakash and M. P. Wellman, "Empirical game-theoretic analysis for moving target defense," in *Proceedings of the 2nd ACM Workshop on Moving Target Defense (MTD '15)*, pp. 57–65, 2015.

[114] G. S. Bopche and B. M. Mehtre, "Graph similarity metrics for assessing temporal changes in attack surface of dynamic networks," *Computers & Security*, vol. 64, pp. 16–43, 2017.

[115] J. Xu, P. Guo, M. Zhao, R. F. Erbacher, M. Zhu, and P. Liu, "Comparing different moving target defense techniques," in *Proceedings of the 1st ACM Workshop on Moving Target Defense (MTD '14)*, pp. 97–107, ACM, Scottsdale, Ariz, USA, 2014.

[116] K. Zaffarano, J. Taylor, and S. Hamilton, "A quantitative framework for moving target defense effectiveness evaluation," in *Proceedings of the 2nd ACM Workshop on Moving Target Defense, MTD 2015*, pp. 3–10, USA.

[117] C. Lei, D.-h. Ma, H.-q. Zhang, and L.-m. Wang, "Moving target network defense effectiveness evaluation based on change-point detection," *Mathematical Problems in Engineering*, vol. 2016, Article ID 6391502, 11 pages, 2016.

[118] J. Taylor, K. Zaffarano, B. Koller, C. Bancroft, and J. Syversen, "Automated effectiveness evaluation of moving target defenses: metrics for missions and attacks," in *Proceedings of the 2016 ACM Workshop on Moving Target Defense, MTD 2016*, pp. 129–134, Austria.

[119] T. C. Eskridge, M. Carvalho, E. Stoner, T. Toggweiler, and A. Granados, "VINE: a cyber emulation environment for MTD experimentation," in *Proceedings of the 2nd ACM Workshop on Moving Target Defense, MTD 2015*, pp. 43–47, USA.

[120] M. Atighetchi, B. Simidchieva, M. Carvalho, and D. Last, "Experimentation support for cyber security evaluations," in *Proceedings of the 11th Annual Cyber and Information Security Research Conference, CISRC 2016*, USA, April 2016.

[121] Q. Jia, K. Sun, and A. Stavrou, "MOTAG: moving target defense against internet denial of service attacks," in *Proceedings of the 2013 IEEE 2013 22nd International Conference on Computer Communication and Networks, ICCCN 2013*, Bahamas, Caribbean, August 2013.

[122] H. Wang, Q. Jia, D. Fleck, W. Powell, F. Li, and A. Stavrou, "A moving target DDoS defense mechanism," *Computer Communications*, vol. 46, pp. 10–21, 2014.

[123] Q. Jia, H. Wang, D. Fleck, F. Li, A. Stavrou, and W. Powell, "Catch me if you can: a cloud-enabled DDoS defense," in *Proceedings of the 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2014*, pp. 264–275, USA, June 2014.

[124] A. Chowdhary, S. Pisharody, and D. Huang, "SDN based scalable MTD solution in cloud network," in *Proceedings of the 2016 ACM Workshop on Moving Target Defense, MTD 2016*, pp. 27–36, Austria.

[125] S. Pisharody, J. Natarajan, A. Chowdhary, A. Alshalan, and D. Huang, "Brew: a security policy analysis framework for distributed SDN-based cloud environments," *IEEE Transactions on Dependable and Secure Computing*, 2017.

[126] V. Heydari, S.-I. Kim, and S.-M. Yoo, "Anti-censorship framework using mobile IPv6 based moving target defense," in *Proceedings of the 11th Annual Cyber and Information Security Research Conference, CISRC 2016*, USA, April 2016.

[127] V. Heydari, S.-I. Kim, and S.-M. Yoo, "Scalable anti-censorship framework using moving target defense for web servers," *IEEE*

Transactions on Information Forensics and Security, vol. 12, no. 5, pp. 1113–1124, 2017.

[128] G. A. Fink and C. S. Oehmen, "Final report for bio-inspired approaches to moving-target defense strategies," Tech. Rep. PNNL-21854, Pacific Northwest National Laboratory (PNNL), Richland, WA, USA, 2012.

[129] M. H. Almeshekah and E. H. Spafford, "Cyber security deception," *Cyber Deception: Building the Scientific Foundation*, pp. 23–50, 2016.

[130] A. Shameli Sendi, Y. Jarraya, M. Pourzandi, and M. Cheriet, "Efficient provisioning of security service function chaining using network security defense patterns," *IEEE Transactions on Services Computing*, pp. 1-1, 2017.

[131] K. Z. Snow, F. Monrose, L. Davi, A. Dmitrienko, C. Liebchen, and A.-R. Sadeghi, "Just-in-time code reuse: on the effectiveness of fine-grained address space layout randomization," in *Proceedings of the 34th IEEE Symposium on Security and Privacy, SP 2013*, pp. 574–588, USA, May 2013.