Cost-effective Defense Timing Selection for Moving Target Defense in Satellite Computing Systems

Lin Zhang[1,2,3], Yunchuan Guo[1,2,3], Siyuan Leng[1,2,3], Xiaogang Cao[1,2,3], Fenghua Li[1,2,3], and Liang Fang[1,3(✉)]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
{zhanglin1716, guoyunchuan, lengsiyuan, caoxiaogang, lifenghua, fangliang}@iie.ac.cn
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3] Key Laboratory of Cyberspace Security Defense, Beijing, China

Abstract. Satellite computing system (SCS), with its huge economic value, is suffering from increasing attacks. Moving Target Defense (MTD) can create the asymmetric situation between attacks and defenses by changing the attack surface. As SCS's limited defense resources, current MTD defense timing selection methods are not suitable for SCS. This paper proposes a Markov Game based Defense Timing Selection (MGDTS) approach for MTD in SCS. MGDTS formulates attack-defense adversarial relationship as a Markov game with incomplete information, and explicit costs are used to define the resource consumption of a defender. For defense timing decision, MGDTS uses a Markov decision process to construct the defense timing decision equation, and a real-time dynamic programming to solve the equation. Experimental results show that compared with other MTDs, MGDTS can improve the security of MTD while reducing its costs.

Keywords: Satellite computing system · Moving target defense · Defense timing selection · Network scanning.

1 Introduction

Satellite Computing System (SCS), which virtualizes the computing resources of high-performance satellites, offers large-scale computing and communication services to space users (e.g., remote sensing satellites). Large aerospace companies are trying their best to develop SCSs. For example, Starlink constellation has more than 30000 Linux nodes (and more than 6000 microcontrollers) in space now and provides extra computing power [24]. Undoubtedly, SCSs own huge economic value. According to The Wall Street Journal, Starlink brought in $1.4 billion in revenue in 2022 [12]. However, due to its huge economic value, SCS is suffering from increasing attacks. For example, Viasat's KA-SAT satellite network was attacked during the Russia-Ukraine war, as a result, communication services in Ukraine and Europe were interrupted [18].

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

To effectively prevent attacks, active defense has been proposed in academia and industry, including honeypots, mimic defense [11] and Moving Target Defense (MTD) [6], where MTD constantly changes systems' attack surfaces and creates the asymmetric situation between attacks and defenses in cyber-security, thus proactively defending against network attacks. In MTD, defense timing selection, which is used to determine when to transform the attack surface, is crucial to increase defense capability and decrease defense cost.

From the perspective of defense timing selection, existing MTD can be divided into three categories: time-driven MTD, event-driven MTD, and hybrid-driven MTD, where a time-driven MTD changes the attack surface in a fixed or variable time period. Similarly, in an event-driven MTD, the attack surface is changed only after specific events are detected. The hybrid-driven scheme combines the above approaches and provides more proactive and immediate defense against attacks. However, existing hybrid-driven schemes cannot be directly applied to SCSs for the following reasons.

(1) **Defense resources in SCS are limited.** Although SCS owns huge communication resources, most resources are reserved for business and only negligible resources are for security defense. However, existing hybrid-driven MTDs consume a large number of communication resources because SCS has to execute multiple round communication interaction with users in MTD.
(2) **Existing hybrid-driven MTDs do not distinguish internal attacks from external attacks.** In SCS, internal attacks can succeed with a high probability and bring more serious consequences, but the probability of internal attacks is relatively low. Oppositely, external attacks against SCS happen with a high probability and succeed with a relatively low probability. As a result, if we do not distinguish these two types of attacks, we either consume more defense resources, or obtain poor defense effects.
(3) **Existing hybrid-driven MTDs rely on historical experience to select defense timing.** This scheme lacks quantitative analysis of the MTD effectiveness under different system security states. As a result, the selected defense timing may be unsuitable.

To address the above challenges, considering that network scanning is the first stage of network attacks, in the paper, a Markov Game based Defense Timing Selection (MGDTS) approach is proposed for MTD to guide the VM IP shuffling in SCS to resist network scanning. Our main contributions are as follows.

(1) Considering that defenders cannot accurately differentiate between interior attacks and exterior attacks, we formulate attack-defense adversarial relationship as a Markov game with incomplete information. In the game, to decrease the consumption of communication resources, explicit costs are used to define the communication resource consumption of a defender, and guide the defender's timing selection.
(2) We construct the defense timing decision equation using a Markov Decision Process (MDP), where the MTD effectiveness is quantified using Bellman

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

equations. Further, Real-Time Dynamic Programming (RTDP) is designed to solve the decision equation and decide the defense timing of MTD.

(3) A series of experiments are conducted and experimental results show that, compared with existing MTDs, MGDTS can effectively reduce the communication resource consumption of MTD while enhancing its security effect.

2 Related Work

MTD can be classified into time-driven, event-driven and hybrid time- and event-driven from the perspective of defense timing selection.

2.1 Time-driven MTD

Time-driven MTD employs a fixed or variable time interval between adjacent system configuration changes. Fixed-interval MTD uses game theory [10, 17, 26], Stochastic Petri Net [1, 27, 28], and quantitative analysis models [2, 14] to determine optimal time intervals. For example, Connell et al. [2] proposed a method to quantify the MTD security and system performance, and determined the optimal reconfiguration period by balancing the two factors. Variable-interval MTD adjusts time intervals based on attack severity [31] or system overhead [15], such as Zangeneh et al. [31] tuning the movement period inversely with the changing of the adversarial severity. Time-driven MTD has good proactivity, and can effectively defend against covert attacks, such as network infiltration [29]. But it cannot respond immediately and adaptively to detected attack behaviors.

2.2 Event-driven MTD

Event-driven MTD takes security alerts [22] and system events (like system calls [13], system errors [7], security level changes [32] and Quality of Service (QoS) variations [8]) as triggers for system configuration changing. For example, Smith et al. [22] proposed an MTD triggered by intrusion alerts to mitigate denial of service, employing Neuro-Evolution of Augmented Topologies (NEAT) to construct an intrusion detector and initiating IP hopping on alerts. Khan et al. [7] introduced an MTD triggered by system errors to resist ransomware, immediately changing file extensions on detection of system errors related to ransomware. Event-driven MTD responds immediately and adaptively to detected attack behaviors, but lags in reacting to network attacks, and missed detections of security events can reduce its effectiveness.

2.3 Hybrid-driven MTD

Hybrid-driven MTD combines the triggering conditions above, with the movement of the attack surface being initiated by time and events. For example, Potteiger et al. [19, 20] use the temporal schedule and attack alerts to trigger address space randomization, thereby mitigating memory corruption attacks.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

Huang et al. [4] rotate virtual servers based on a schedule and anomaly detection results to resist Web attacks. To counter scanning attacks, Prakash [21] and Xu [30] initiate MTD when pre-setting timers expire or attack behaviors are detected. Hybrid-driven MTD can synthesize the advantages of time-driven and event-driven MTD, offering proactive defense while responding immediately and adaptively to attacks. However, defense resources in SCS are limited, and existing hybrid-driven MTDs do not distinguish internal attacks from external attacks and rely on historical experience to select defense timing. As a result, existing hybrid-driven MTDs cannot be directly applied to SCS.

3 System and Threat Model

SCS enables sharing of computing satellite resources via resource pooling and virtualization, generating m VMs, each of which has a unique IP address assigned from the IP address pool Γ at any given moment (Fig. 1).

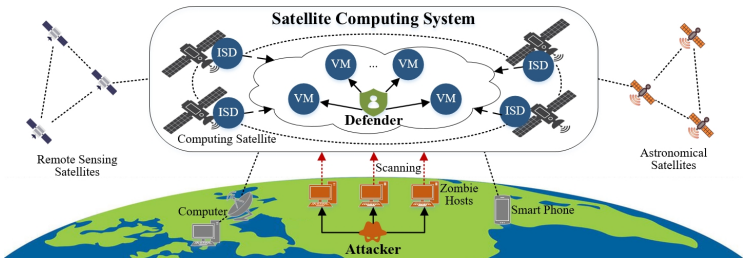

Fig. 1. System and threat model

To prepare for deeper active attacks, the ground-based attacker, who is either internal or external, uses zombie hosts and consumes resources such as money to scan VM IPs. The scanning policy is cyclic non-repetitive scanning, which involves iteratively scanning all IP addresses in Γ in a random order without repetition. To render the IP information obtained by the attacker invalid, the defender on SCS consumes communication resources to execute VM IP shuffling, which also affects the network QoS. When the IP Scanning Detector (ISD) on the computing satellite detects the zombie host IP scan, it informs the defender and blacklists the zombie host's IP. The defender selects shuffling timing based on the detection results from ISDs and time, and the attacker eliminates ISDs' interference by changing the zombie host IP.

Assume that whether the attacker is internal or external is his private information, while other information is public (including probabilities of the attacker's two identities). Both the rational attacker and defender make decisions based on all the information they possess. Assuming negligible VM IP switching time, the maximum transmission delays of the new IP address and the scanning packet determine the time required for each shuffling and each scan, respectively, and their maximum transmission delay is τ.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

4 Markov Game

MGDTS views the entire game process as a time sequence $\{\tau_1, \tau_2, \ldots\}$ composed of time steps. In each time step τ, each zombie host and each VM can perform IP scan and VM IP shuffling at most once, respectively. The state of each VM at each time step is defined based on the security indexes that the defender can actually measure.

Definition 1. The state s of a VM is a tuple $< k, t >$. $k \in \{0, 1\}$ indicates whether ISDs have detected $(k = 1)$ or not detected $(k = 0)$ a scan for the VM's IP since the last VM IP shuffling. $t \in \{0, 1, \ldots, t_{max}\}$ represents the number of time steps since the last VM IP shuffling, where t_{max} is the maximum validity period of IP. When $t = t_{max}$, VM requests a new IP address. The state of a VM at time step τ_i is denoted as $s_{\tau_i} =< k_{\tau_i}, t_{\tau_i} >$, where $i \in \{1, 2, \ldots\}$.

MGDTS defines the players of the game as the attacker and the defender. Whether the attacker is internal or external, which affects his benefit, is not the defender's knowledge. Thus the game is an incomplete information game.

Definition 2. The type θ of the player refers to all the private information possessed by the player. The type θ_1 of the attacker has two possible values: θ_{11} for the attacker being internal and θ_{12} for the attacker being external. As all the defender information needed by the attacker is public, the type θ_2 of the defender has only one possible value. The probabilities of the attacker's type being θ_{11} and θ_{12}, denoted as $p(\theta_{11})$ and $p(\theta_{12})$ respectively, satisfy: $p(\theta_{11}) + p(\theta_{12}) = 1$.

Each player has a policy that guides their actions at each time step.

Definition 3. The action a of the player refers to the decision variable of the player at a certain time point. The action $a_{1,\tau_i}(\theta_1)$ of the attacker represents that the attacker of type θ_1 uses $n_{\tau_i}(\theta_1)$ zombie hosts to execute IP scanning at time step τ_i, where $\theta_1 \in \{\theta_{11}, \theta_{12}\}$, $n_{\tau_i}(\theta_1) \in \{0, 1, \ldots, n_{max}\}$, and n_{max} represents the total number of zombie hosts owned by the attacker. According to the linear model [7], the relationship between the defender's scanning detection success rate α and $n_{\tau_i}(\theta_1)$ is: $\alpha = min(dn_{\tau_i}(\theta_1), 1)$, where d represents the change rate of α relative to $n_{\tau_i}(\theta_1)$. The action $a_2(s) \in \{es, ns\}$ of the defender represents whether the defender executes (es) or does not execute (ns) an IP address shuffling for the VM in state s.

Definition 4. The policy π of the player specifies the action that the player selects at each time point. The policy π_1 of the attacker specifies the number of zombie hosts used by different types of attackers at each time step. The policy π_2 of the defender specifies the action taken by the defender in each state.

The attacker expends resources to launch scans and cope with the defender's interference, preparing for deeper active attacks on SCS. The defender protects SCS's assets through VM IP shuffling, which consumes communication resources and impacts network QoS, such as increased network latency. The costs, benefits and utilities of the players are defined below:

Definition 5. The cost of the attacker's action $a_{1,\tau_i}(\theta_1)$ includes the scanning cost and the cost of changing zombie host IPs. The scanning cost is the product of the scanning times and the single scan cost $c_{11}(\theta_1)$. The cost $c_{11}(\theta_1)$ represents the total resources expended by the attacker of type θ_1 on executing

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

a scan. The cost of changing zombie host IPs is the product of the blacklisted IP count and the single IP changing cost $c_{12}(\theta_1)$. The cost $c_{12}(\theta_1)$ represents the total resources expended by the attacker of type θ_1 on changing a zombie host IP. The cost of the defender's action $a_2(s)$ includes explicit and implicit costs. The explicit cost $c_{21}(a_2(s))$ represents the total resources expended by the defender on executing $a_2(s)$. The implicit cost $c_{22}(a_2(s))$ represents the reduced network QoS when executing $a_2(s)$.

Definition 6. The benefit of the attacker's action $a_{1,\tau_i}(\theta_1)$ is the product of the scanning times and the single scan benefit $e_1(\theta_1)$. The benefit $e_1(\theta_1)$ represents the increase in SCS's asset risk resulting from the attacker of type θ_1 executing a scan. The asset risk is determined by the success rate of the subsequent active attack and the asset value λ. If an active attack is successful, SCS loses all its assets. The success rate of the active attack is proportional to the number of VM IPs obtained by the attacker. When the attacker of type θ_1 obtains all VM IPs, the success rate reaches its maximum value $\mu(\theta_1)$. The benefit $e_2(s, a_2(s))$ of the defender's action $a_2(s)$ on a VM in state s represents the reduction of SCS's asset risk resulting from executing $a_2(s)$.

Definition 7. The utility $v_{1,\pi_1,\tau_i}(\theta_1)$ of the attacker of type θ_1 represents the difference between his benefit and cost at time step τ_i under policy π_1. As the defender's benefit is related to the state of VM, the utility $v_{2,\pi_2}(s)$ of the defender represents the sum of his instantaneous utility and subsequent utility at a VM of state s under policy π_2. The instantaneous utility represents the difference between the defender's benefit and cost for executing his action, and the subsequent utility represents the expected utility in future states resulting from the defender's action.

As the attacker's utility is independent of VMs' state, the number of zombie hosts chosen by the attacker of type θ_1 at different time steps is the same, which can be denoted as $n(\theta_1)$. Thus the game is a Markov game associated with the process of VM state transition, where the attacker aims to decide $n(\theta_1)$ according to his utility and the defender aims to decide $a_2(s)$ in each state s.

5 Defense Timing Decision

5.1 Decision Equations of Players

Decision Equation of the Attacker The rational attacker and defender aim to maximize utilities in decision-making. The attacker's decision variable is his policy, represented by $\pi_1 =< n(\theta_{11}), n(\theta_{12}) >$, and the optimal policy is denoted as $\pi_1^* =< n^*(\theta_{11}), n^*(\theta_{12}) >$. The attacker's objective functions are the utilities of all types of attackers. According to Definition 5-7, the utility of the attacker of type θ_1 in τ under the policy π_1 is:

$$v_{1,\pi_1,\tau}(\theta_1) = n(\theta_1)\left(\frac{\lambda\mu(\theta_1)}{|\Gamma|} - c_{11}(\theta_1) - \alpha c_{12}(\theta_1)\right) \tag{1}$$

The attacker's constraints are the ranges of $n(\theta_{11})$ and $n(\theta_{12})$. Thus, the attacker's decision equation can be defined as:

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

$$\begin{cases} \forall\theta_1 \in \{\theta_{11}, \theta_{12}\}, max\left(v_{1,\pi_1,\tau}\left(\theta_1\right)\right) \\ s.t.\ \forall\theta_1 \in \{\theta_{11}, \theta_{12}\}, n\left(\theta_1\right) \in \{0, 1, \ldots, n_{max}\} \end{cases} \tag{2}$$

Decision Equation of the Defender As the defender's utility is related to the VM's state, the VM's state transition is analyzed before defining the defender's decision equation. When the attacker of type θ_1 and the defender execute actions on the VM in state $s_{\tau_i} =< k_{\tau_i}, t_{\tau_i} >$ at time step τ_i, the VM's state changes based on the following rules:

(1) $a_2\left(s_{\tau_i}\right) = es$: The VM's state at the next time step changes to $s_{\tau_{i+1}} =< 0, 0 >$.
(2) $a_2\left(s_{\tau_i}\right) = ns$ and $k_{\tau_i} = 0$: Given the attacker's cyclic non-repetitive scanning policy and the condition $k_{\tau_i} = 0$, the probability $p\left(0|k_{\tau_i} = 0\right)$ that the attacker has scanned the VM's IP 0 times in the current round of non-repetitive scanning can be calculated using Bayes' theorem:

$$\begin{aligned} p\left(0|k_{\tau_i} = 0\right) &= \frac{p(0) \times p\left(k_{\tau_i} = 0|0\right)}{\sum_{x\in\{0,1\}} p(x) \times p\left(k_{\tau_i} = 0|x\right)} \\ &= \frac{|\Gamma| - n\left(\theta_1\right) t_{\tau_i} + \left\lfloor \frac{n(\theta_1) t_{\tau_i}}{|\Gamma|} \right\rfloor \left|\Gamma\right|}{|\Gamma| - n\left(\theta_1\right) t_{\tau_i}\alpha + \left\lfloor \frac{n(\theta_1) t_{\tau_i}}{|\Gamma|} \right\rfloor \left|\Gamma\right|\alpha} \end{aligned} \tag{3}$$

where $p(x)$ represents the probability that the attacker has scanned the VM's IP x times in the current round of non-repetitive scanning, $p\left(k_{\tau_i} = 0|x\right)$ represents the probability of $k_{\tau_i} = 0$ given that the attacker has scanned the VM's IP x times in the current round of non-repetitive scanning. Therefore, when $a_2\left(s_{\tau_i}\right) = ns$ and $k_{\tau_i} = 0$, the probability that the VM's state changes to $s_{\tau_{i+1}} =< k_{\tau_i} + 1, t_{\tau_i} + 1 >$ in the next time step is:

$$\begin{aligned} \beta_{t_{\tau_i}}\left(\theta_1\right) &= p\left(0|k_{\tau_i} = 0\right) \times \frac{n\left(\theta_1\right)}{|\Gamma| - n\left(\theta_1\right) t_{\tau_i} + \left\lfloor \frac{n(\theta_1) t_{\tau_i}}{|\Gamma|} \right\rfloor |\Gamma|} \times \alpha \\ &= \frac{n\left(\theta_1\right)\alpha}{|\Gamma| - n\left(\theta_1\right) t_{\tau_i}\alpha + \left\lfloor \frac{n(\theta_1) t_{\tau_i}}{|\Gamma|} \right\rfloor \left|\Gamma\right|\alpha} \end{aligned} \tag{4}$$

and the probability that the VM's state changes to $s_{\tau_{i+1}} =< k_{\tau_i}, t_{\tau_i} + 1 >$ in the next time step is $1 - \beta_{t_{\tau_i}}\left(\theta_1\right)$.
(3) $a_2\left(s_{\tau_i}\right) = ns$ and $k_{\tau_i} = 1$: The VM's state at the next time step changes to $s_{\tau_{i+1}} =< k_{\tau_i}, t_{\tau_i} + 1 >$.

When $t_{max} = 2$, the state transition process of VM is shown in Fig. 2. As the VM state is finite and the state transition is random and Markovian, MGDTS constructs the defender's decision equation based on MDP.

The defender's MDP is a six-tuple $< S, A, p\left(s' \middle| s, a_2(s), \theta_1\right), \pi_2, v_{2,\pi_2}(s), \gamma >$:

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

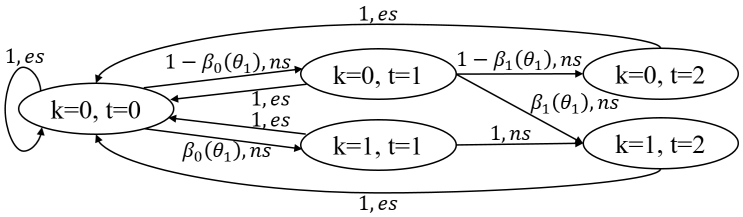

Fig. 2. Example of the state transition process

(1) $S = \{s =< k, t > | k \in \{0, 1\}, t \in \{0, 1, \ldots, t_{max}\}, k \leq t\}$ is a set of states.
(2) $A = \{es, ns\}$ is a set of actions. The defender can only choose one action $a_2(s)$ from A to execute for VM in state $s \in S$.
(3) $p\left(s^{'} \middle| s, a_2(s), \theta_1\right)$ is the probability that VM changes to the new state $s^{'}$ given the current state s, the taken action $a_2(s)$, and the attacker type θ_1.
(4) $\pi_2 =< a_2(s_1), a_2(s_2), \ldots, a_2\left(s_{|S|}\right) >$ is the defender's policy.
(5) $v_{2,\pi_2}(s)$ is the defender's utility obtained in state s under policy π_2.
(6) $\gamma \in [0, 1]$ is the discount factor representing the importance of future utilities.

The defender's decision variable is the defender's policy, where the optimal policy is denoted as $\pi_2^* =< a_2^*(s_1), a_2^*(s_2), \ldots, a_2^*\left(s_{|S|}\right) >$.

The defender's objective functions are the defender's utilities in all possible VM states. According to Definition 7 and MDP, the defender's utility in state s under policy π_2 is defined using the Bellman equation:

$$v_{2,\pi_2}(s) = \sum_{\theta_1 \in \{\theta_{11}, \theta_{12}\}} p(\theta_1) v_{2,\pi_2}(s, \theta_1) \tag{5}$$

$$\begin{aligned} v_{2,\pi_2}(s, \theta_1) = {} & e_2(s, a_2(s)) - c_{21}(a_2(s)) - c_{22}(a_2(s)) \\ & + \gamma \sum_{s^{'} \in S} p\left(s^{'} \middle| s, a_2(s), \theta_1\right) v_{2,\pi_2}\left(s^{'}, \theta_1\right) \end{aligned} \tag{6}$$

where $v_{2,\pi_2}(s, \theta_1)$ is the defender's utility given the policy π_2, state s, and attacker type θ_1. According to Definition 6, the defender's benefit $e_2(s, a_2(s))$ is calculated as follows:

$$e_2(s, a_2(s)) = \sum_{s^{'} \in S} p\left(s^{'} \middle| s, a_2(s), \theta_1\right) \left(r(\theta_1, s) - r\left(\theta_1, s^{'}\right)\right) \tag{7}$$

where $r(\theta_1, s)$ is the risk value brought to SCS by VM in state s given the attacker type θ_1. Assuming that all VMs are identical in terms of configuration and software, except for their IP addresses, the increased risk value after VM IP leakage is $\frac{\lambda\mu(\theta_1)}{m}$. Then, the risk value $r(\theta_1, s)$ is calculated as follows:

$$r(\theta_1, s) = \eta(\theta_1, s) \times \frac{\lambda\mu(\theta_1)}{m} \tag{8}$$

where $\eta(\theta_1, s)$ is the probability of VM IP leakage in state s given the attacker type θ_1, and is calculated based on the elements k and t of state s:

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

$$\eta\left(\theta_1, s\right)=\begin{cases}\frac{n(\theta_1)t-n(\theta_1)t\alpha}{|\Gamma|-n(\theta_1)t\alpha} & k=0, t<\frac{|\Gamma|}{n(\theta_1)} \\ 1 & other\end{cases} \tag{9}$$

The defender's constraints include the ranges of $a_2\left(s_1\right)$, $a_2\left(s_2\right), \ldots, a_2\left(s_{|S|}\right)$, as well as $n\left(\theta_{11}\right)=n^*\left(\theta_{11}\right)$ and $n\left(\theta_{12}\right)=n^*\left(\theta_{12}\right)$.

Based on the decision variable, objective functions and constraints, the defender's decision equation can be defined as:

$$\begin{cases}\forall s \in S, max\left(v_{2,\pi_2}(s)\right) \\ s.t.\ \forall s \in\{<k, t>|<k, t>\in S, t \neq t_{max}\}, a_2(s) \in A \\ \forall s \in\{<k, t>|<k, t>\in S, t=t_{max}\}, a_2(s) \in\{es\} \\ n\left(\theta_{11}\right)=n^*\left(\theta_{11}\right), n\left(\theta_{12}\right)=n^*\left(\theta_{12}\right)\end{cases} \tag{10}$$

5.2 Solutions of Decision Equations

Solution of the Attacker From Formula (2), the attacker's decision equation can be decomposed into 2 independent integer programming problems, and each problem corresponds to an objective function. The integer programming problem can be solved by the branch and bound [9] method, which can reduce the computational complexity of the solution process through pruning. The solution of the decision equation is obtained by integrating the solutions of all integer programming problems.

Solution of the Defender The size of the defender's policy space is exponentially related to the state space, rendering Exhaustive Enumeration (EE) [5] computationally intractable. Synchronous dynamic programming algorithms, such as Value Iteration (VI) [3], update the value function or policy for all states simultaneously, which are inefficient when the state space is large. Asynchronous dynamic programming algorithms, such as RTDP, are more efficient by selectively updating states. Therefore, MGDTS uses RTDP to find the defender's optimal policy, and the process is shown in Algorithm 1.

6 Experiments

6.1 Experimental Environment and Settings

To evaluate the performance and effect of MGDTS, a SCS is constructed using the OMNET++ simulation software, as shown in Fig. 3.

The SCS consists of 3 computing satellites ($ComSat0$~$ComSat2$) and the attacker is *Attacker*. According to references [16,23,25,33], the parameter values of the system, attacker and defender are listed in Table 1:

The single scan cost is determined by the satellite tariff of the scan packet. The implicit and explicit costs of each VM IP shuffling are determined by the satellite tariffs of the IP notification packet and affected user service traffic, respectively.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

Algorithm 1 Solution of the defender

Input: the defender's decision variable π_2, objective functions: $\forall s \in S, v_{2,\pi_2}(s)$, constraints, state transition probability $p\left(s^{'} \middle| s, a_2(s), \theta_1\right)$, threshold of objective function change th_{φ}, maximum validity period of IP address t_{max}
Output: the optimal policy π_2^*

$\forall s \in S, v_{2,\pi_2}(s) = 0$, $\pi_2^* =< es, es, \ldots, es >$
Initialize the maximum change of objective functions: $\varphi = 0$, initialize the variables: $k = 0$, $t = 0$, $flag = false$
while $k \neq 0$ or $t \neq 0$ or $!flag$ **do**
    if $k = 0$ and $t = 0$ **then**
        $flag = true$
    Generate a new episode with the initial state $< k_{\tau_1}, t_{\tau_1} >=< k, t >$
    Set the episode's current time step τ_i to τ_1
    while $k_{\tau_i}! = 0$ or $t_{\tau_i}! = 0$ or $\tau_i = \tau_1$ **do**
        Based on the current values of objective functions of all states, calculate the utilities of performing all possible actions in state $< k_{\tau_i}, t_{\tau_i} >$
        Update $v_{2,\pi_2}\left(< k_{\tau_i}, t_{\tau_i} >\right)$ with the larger utility, and update π_2^* with the action that generates the larger utility
        Randomly select the next time step state $< k_{\tau_{i+1}}, t_{\tau_{i+1}} >$ based on the above action and the state transition probability
        $k_{\tau_i} = k_{\tau_{i+1}}$, $t_{\tau_i} = t_{\tau_{i+1}}$, $\tau_i = \tau_{i+1}$
    Calculate φ in the episode
    if $\varphi < th_{\varphi}$ **then**
        $t = \left(t + \left\lfloor \frac{k+1}{\min(2,t+1)} \right\rfloor\right) \%\, (t_{max} + 1)$, $k = (k+1)\,\%min(2, t+1)$
    else
        $flag = false$
return π_2^*

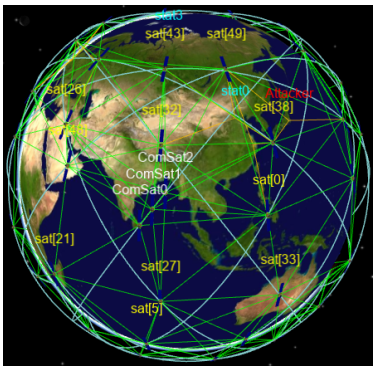

Fig. 3. Simulation construction of SCS

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

Table 1. Parameters of the system, attacker and defender

	Parameter	Value
SCS	VM IP address space	192.168.0.1~192.168.15.255
	User bandwidth	200Mbps
	Average user service traffic in a VM	6.14MB/s
	Packet transmission delay	$\leq$ 200ms
	Satellite tariff	0.25¥/MB
Attacker	Probabilities of θ_{11} & θ_{12}	0.3 & 0.7
	Max. active attack success rate of θ_{11} & θ_{12}	0.04 & 0.02
	Total number of zombie hosts	100
	Scanning packet size	78B
	Cost of changing a single zombie host IP	0.2¥
Defender	Asset value	18000
	Change rate of α relative to $n(\theta_1)$	0.1
	IP notification packet size	40KB
	Discount factor	0.9

6.2 Performance Evaluation

To evaluate RTDP's performance in solving MDP, this experiment compares the solutions and the update counts of objective functions of EE [5], VI [3] and RTDP in state sets with different sizes (by changing t_{max}). Table 2 shows that under different sizes of state sets, the solutions obtained by the 3 algorithms are consistent with the exact solution. Additionally, Fig. 4 reveals that under different sizes of state sets, the number of updates in RTDP is far lower than that in EE and VI. The above results indicate that RTDP can efficiently obtain solutions that are consistent with the exact solutions.

Table 2. Comparison of the solutions

$\|S\|$	Is the same as the exact solution?		
	EE	VI	RTDP
20481	Yes	Yes	Yes
40961	Yes	Yes	Yes
61441	Yes	Yes	Yes
81921	Yes	Yes	Yes
102401	Yes	Yes	Yes

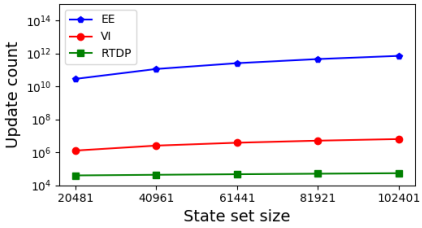

Fig. 4. Comparison of update counts

6.3 Defense Effect Evaluation

To evaluate MGDTS's defensive effect against IP scanning attacks, this experiment compares it with the typical time-driven, event-driven and hybrid-driven MTD methods, which are Quantitative Analytic Model based MTD (QAMMTD) [2], NEAT Detector based MTD (NDMTD) [22] and Attack Detector and Timer based MTD (ADTMTD) [30], respectively. All 4 MTD methods randomly select

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

the new IP address of VM. The evaluation indexes include: defender utility, IP leakage time, bandwidth consumption and network QoS. The following results are the average of 10 simulation runs. Each run lasts 3600s.

Defender Utility Fig. 5 shows the defender utility obtained by the 4 MTD methods for each VM state $< k, t >$. Fig. 5(a) and Fig. 5(b) demonstrate the trend of defender utility changing with t when $k = 0$, and Fig. 5(c) and Fig. 5(d) demonstrate the trend when $k = 1$, with $t_{max} = 10240$. According to Fig. 5, MGDTS achieves higher defender utilities compared to the other 3 methods for all states, especially when MGDTS takes different actions from the other 3 methods, such as when $k = 1$ and $0 \leq t < 65$ for QAMMTD, $k = 0$ and $90 < t < 10240$ for NDMTD, and $k = 0$ and $90 < t < 100$ for ADTMTD.

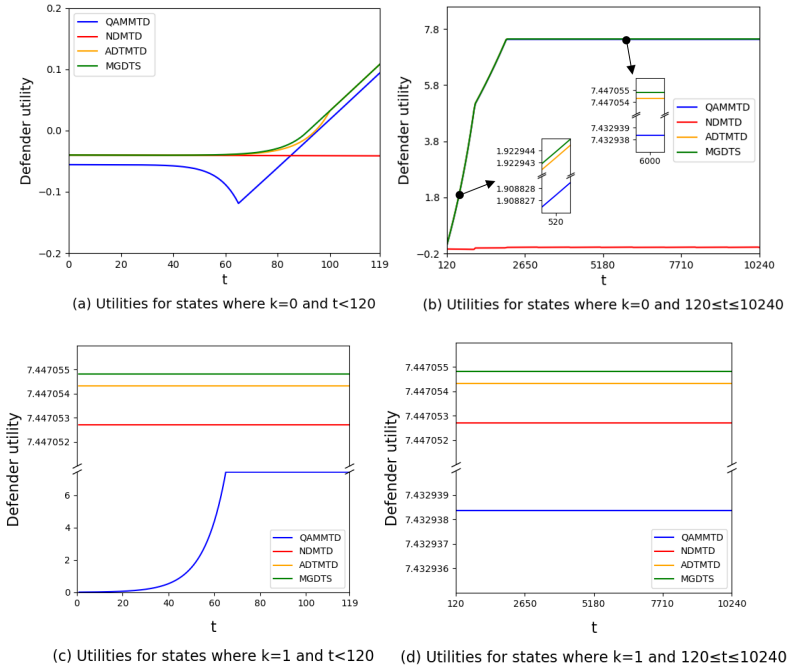

Fig. 5. Defender utilities obtained by 4 MTD methods

IP Leakage Time The leakage time of a VM IP address is the duration from the moment an attacker successfully scans the address to the time when the address is changed. This experiment's IP leakage time is the sum of the leakage time of all VMs' used IP addresses. The change of IP leakage time in the simulation time under 4 MTD methods is shown in Fig. 6. Overall, the IP leakage time of MGDTS is less than that of the other 3 methods. This indicates that when the VM IP is leaked, MGDTS can replace the secure IP for VM more immediately, making the attacker's information invalid earlier, thereby improving the security of SCS.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

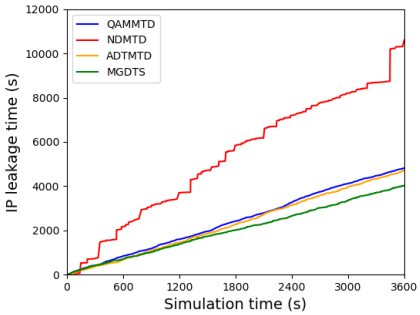

Fig. 6. IP leakage time under 4 MTD methods

Bandwidth Consumption and Network QoS The change of the cumulative bandwidth consumption and QoS impact of 4 MTD methods in the simulation time is shown in Fig. 7. Overall, the cumulative bandwidth consumption and QoS impact of MGDTS are lower than those of the other 3 methods. Combined with the experimental results of IP leakage time, it can be concluded that MGDTS provides better security for the system while reducing the explicit and implicit costs of MTD. Therefore, MGDTS is a cost-effective timing selection method.

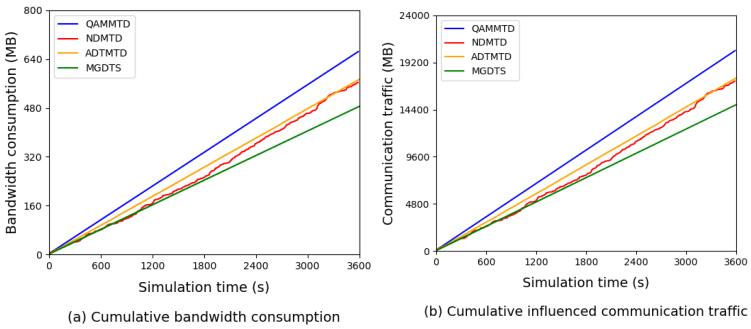

Fig. 7. Cumulative bandwidth consumption and QoS impact of 4 MTD methods

7 Conclusion

This paper introduces MGDTS, an approach for selecting the defense timing of MTD. It disrupts network attacks during the network scanning phase by guiding VM IP shuffling in SCS. MGDTS uses a Markov game with incomplete information to formulate attack-defense confrontation, and explicit costs are used to define the resource consumption of a defender. For defense timing decision, MGDTS uses a MDP and a RTDP to construct and solve the decision equation. Experiments show that MGDTS is efficient. It has higher defender utilities than existing MTDs in all VM states. Compared with these MTDs, MGDTS can better enhance the security of SCS while reducing the explicit and implicit costs of MTD. Consequently, it is cost-effective in defense timing selection.

Future work can consider combining VM IP shuffling with other types of MTD, such as VM migration.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

Acknowledgments. This work is supported by the National Key Research and Development Program of China (No.2023YFB3107605), the National Natural Science Foundation of China (No.62202463), the Youth Innovation Promotion Association CAS (No.2021154) and the Pandeng Project of IIE CAS.

References

1. Chen, Z., Chang, X., Han, Z., Yang, Y.: Numerical evaluation of job finish time under mtd environment. IEEE Access **8**, 11437–11446 (2020)
2. Connell, W., Menasce, D.A., Albanese, M.: Performance modeling of moving target defenses with reconfiguration limits. IEEE Transactions on Dependable and Secure Computing (2018). https://doi.org/10.1109/TDSC.2018.2882825
3. Farahmand, A.m., Ghavamzadeh, M.: Pid accelerated value iteration algorithm. In: International Conference on Machine Learning. pp. 3143–3153. PMLR (2021)
4. Huang, Y., Ghosh, A.K.: Introducing diversity and uncertainty to create moving attack surfaces for web services. In: Moving Target Defense: Creating Asymmetric Uncertainty for Cyber Threats, pp. 131–151. Springer (2011)
5. Ingels, F., Azaïs, R.: Enumeration of irredundant forests. Theoretical Computer Science **922**, 312–334 (2022). https://doi.org/https://doi.org/10.1016/j.tcs.2022.04.033
6. Jajodia, S., Ghosh, A.K., Swarup, V., Wang, C., Wang, X.S.: Moving target defense: creating asymmetric uncertainty for cyber threats, vol. 54. Springer Science & Business Media (2011)
7. Khan, M.M., Hyder, M.F., Khan, S.M., Arshad, J., Khan, M.M.: Ransomware prevention using moving target defense based approach. Concurrency and Computation: Practice and Experience **35**(7), e7592 (2023)
8. Kim, D.S., Kim, M., Cho, J.H., Lim, H., Moore, T.J., Nelson, F.F.: Design and performance analysis of software defined networking based web services adopting moving target defense. In: DSN-S. pp. 43–44. IEEE (2020)
9. Lan, J., Brückner, B., Lomuscio, A.: A semidefinite relaxation based branch-and-bound method for tight neural network verification. In: AAAI conference on artificial intelligence (2023). https://doi.org/10.1609/aaai.v37i12.26745
10. Lei, C., Zhang, H.Q., Wan, L.M., Liu, L., Ma, D.h.: Incomplete information markov game theoretic approach to strategy generation for moving target defense. Computer Communications **116**, 184–199 (2018)
11. Lingshu, L., Jiangxing, W., Hongchao, H., Wenyan, L., Zehua, G.: Secure cloud architecture for 5g core network. Chinese Journal of Electronics (2021)
12. Maidenberg, Micah and Winkler, R.: Starlink surges but is still far short of spacex's goals, documents show (2023), https://www.wsj.com/tech/spacexs-starlink-demonstrates-its-power-but-still-needs-growth-9906c5b0?mod=hp_lead_pos5
13. Masumoto, T., Oo, W.K.K., Koide, H.: Mtd: Run-time system call mapping randomization. In: ISCSIC. pp. 257–263. IEEE (2021)
14. Mendonça, J., Cho, J.H., Moore, T.J., Nelson, F.F., Lim, H., Zimmermann, A., Kim, D.S.: Performability analysis of services in a software-defined networking adopting time-based moving target defense mechanisms. In: Annual ACM Symposium on Applied Computing. pp. 1180–1189 (2020)
15. Mercado-Velázquez, A.A., Escamilla-Ambrosio, P.J., Ortiz-Rodriguez, F.: A moving target defense strategy for internet of things cybersecurity. IEEE Access **9**, 118406–118418 (2021). https://doi.org/10.1109/ACCESS.2021.3107403

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16

16. Michel, F., Trevisan, M., Giordano, D., Bonaventure, O.: A first look at starlink performance. In: ACM Internet Measurement Conference. pp. 130–136 (2022)
17. Osei, A.B., Yeginati, S.R., Al Mtawa, Y., Halabi, T.: Optimized moving target defense against ddos attacks in iot networks: When to adapt? In: GLOBECOM 2022-2022 IEEE Global Communications Conference. pp. 2782–2787. IEEE (2022)
18. Pearson, J., Satter, Raphael and Bing, C.J.: Exclusive: U.s. spy agency probes sabotage of satellite internet during russian invasion, sources say (2022), https://www.reuters.com/world/europe/exclusive-us-spy-agency-probes-sabotage-satellite-internet-during-russian-2022-03-11/
19. Potteiger, B., Cai, F., Dubey, A., Koutsoukos, X., Zhang, Z.: Security in mixed time and event triggered cyber-physical systems using moving target defense. In: International Symposium on Real-Time Distributed Computing. IEEE (2020)
20. Potteiger, B., Dubey, A., Cai, F., Koutsoukos, X., Zhang, Z.: Moving target defense for the security and resilience of mixed time and event triggered cyber–physical systems. Journal of Systems Architecture **125**, 102420 (2022)
21. Prakash, A., Wellman, M.P.: Empirical game-theoretic analysis for moving target defense. In: ACM workshop on moving target defense. pp. 57–65 (2015). https://doi.org/10.1145/2808475.2808483
22. Smith, R.J., Zincir-Heywood, A.N., Heywood, M.I., Jacobs, J.T.: Initiating a moving target network defense with a real-time neuro-evolutionary detector. In: Genetic and Evolutionary Computation Conference Companion. pp. 1095–1102 (2016)
23. SpaceX: Starlink (2024), https://www.starlink.com/
24. SpaceX: We are the spacex software team, ask us anything (2024), https://old.reddit.com/r/spacex/comments/gxb7j1/we_are_the_spacex_software_team_ask_us_anything/?limit=500
25. StormProxies: Stormproxies (2024), https://www.stormproxies.cn/
26. Tan, J., Zhang, H., Zhang, H., Hu, H., Lei, C., Qin, Z.: Optimal temporospatial strategy selection approach to moving target defense: A flipit differential game model. Computers & Security **108**, 102342 (2021)
27. Torquato, M., Maciel, P., Vieira, M.: Pymtdevaluator: A tool for time-based moving target defense evaluation: Tool description paper. In: International Symposium on Software Reliability Engineering. pp. 357–366. IEEE (2021). https://doi.org/10.1109/ISSRE52982.2021.00045
28. Torquato, M., Maciel, P., Vieira, M.: Software rejuvenation meets moving target defense: Modeling of time-based virtual machine migration approach. In: International Symposium on Software Reliability Engineering. pp. 205–216. IEEE (2022)
29. Venkatesan, S., Albanese, M., Cybenko, G., Jajodia, S.: A moving target defense approach to disrupting stealthy botnets. In: Proceedings of the 2016 ACM Workshop on Moving Target Defense. pp. 37–46 (2016)
30. Xu, X., Hu, H., Liu, Y., Zhang, H., Chang, D.: An adaptive ip hopping approach for moving target defense using a light-weight cnn detector. Security and Communication Networks **2021**, 1–17 (2021). https://doi.org/https://doi.org/10.1155/2021/8848473
31. Zangeneh, V., Shajari, M.: A cost-sensitive move selection strategy for moving target defense. Computers & Security **75**, 72–91 (2018)
32. Zheng, J., Siami Namin, A.: A markov decision process to determine optimal policies in moving target. In: ACM SIGSAC conference on computer and communications security (2018). https://doi.org/10.1145/3243734.3278489
33. Zhou, Y., Cheng, G., Jiang, S., Zhao, Y., Chen, Z.: Cost-effective moving target defense against ddos attacks using trilateral game and multi-objective markov decision processes. Computers & Security **97**, 101976 (2020)

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63749-0_16