Adaptive Honeypot Engagement Through Reinforcement Learning of Semi-Markov Decision Processes

Linan Huang(✉) and Quanyan Zhu

Department of Electrical and Computer Engineering, New York University,
2 MetroTech Center, Brooklyn, NY 11201, USA
{lh2328,qz494}@nyu.edu

Abstract. A honeynet is a promising active cyber defense mechanism. It reveals the fundamental Indicators of Compromise (IoCs) by luring attackers to conduct adversarial behaviors in a controlled and monitored environment. The active interaction at the honeynet brings a high reward but also introduces high implementation costs and risks of adversarial honeynet exploitation. In this work, we apply infinite-horizon Semi-Markov Decision Process (SMDP) to characterize a stochastic transition and sojourn time of attackers in the honeynet and quantify the reward-risk trade-off. In particular, we design adaptive long-term engagement policies shown to be risk-averse, cost-effective, and time-efficient. Numerical results have demonstrated that our adaptive engagement policies can quickly attract attackers to the target honeypot and engage them for a sufficiently long period to obtain worthy threat information. Meanwhile, the penetration probability is kept at a low level. The results show that the expected utility is robust against attackers of a large range of persistence and intelligence. Finally, we apply reinforcement learning to the SMDP to solve the *curse of modeling*. Under a prudent choice of the learning rate and exploration policy, we achieve a quick and robust convergence of the optimal policy and value.

Keywords: Reinforcement learning · Semi-Markov decision processes · Active defense · Honeynet · Risk quantification

1 Introduction

Recent instances of `WannaCry` ransomware attack and `Stuxnet` malware have demonstrated an inadequacy of traditional cybersecurity techniques such as the firewall and intrusion detection systems. These passive defense mechanisms can detect low-level Indicators of Compromise (IoCs) such as hash values, IP addresses, and domain names. However, they can hardly disclose high-level indicators such as attack tools and Tactics, Techniques and Procedures (TTPs) of

Q. Zhu—This research is supported in part by NSF under grant ECCS-1847056, CNS-1544782, and SES-1541164, and in part by ARO grant W911NF1910041.

T. Alpcan et al. (Eds.): GameSec 2019, LNCS 11836, pp. 196–216, 2019.
https://doi.org/10.1007/978-3-030-32430-8_13

the attacker, which induces the attacker fewer pains to adapt to the defense mechanism, evade the indicators, and launch revised attacks as shown in the pyramid of pain [2]. Since high-level indicators are more effective in deterring emerging advanced attacks yet harder to acquire through the traditional passive mechanism, defenders need to adopt active defense paradigms to learn these fundamental characteristics of the attacker, attribute cyber attacks [35], and design defensive countermeasures correspondingly.

Honeypots are one of the most frequently employed active defense techniques to gather information on threats. A honeynet is a network of honeypots, which emulates the real production system but has no production activities nor authorized services. Thus, an interaction with a honeynet, e.g., unauthorized inbound connections to any honeypot, directly reveals malicious activities. On the contrary, traditional passive techniques such as firewall logs or IDS sensors have to separate attacks from a ton of legitimate activities, thus provide much more false alarms and may still miss some unknown attacks.

Besides a more effective identification and denial of adversarial exploitation through low-level indicators such as the inbound traffic, a honeynet can also help defenders to achieve the goal of identifying attackers' TTPs under proper engagement actions. The defender can interact with attackers and allow them to probe and perform in the honeynet until she has learned the attacker's fundamental characteristics. More services a honeynet emulates, more activities an attacker is allowed to perform, and a higher degree of interactions together result in a larger revelation probability of the attacker's TTPs. However, the additional services and reduced restrictions also bring extra risks. Attacks may use some honeypots as pivot nodes to launch attackers against other production systems [37].

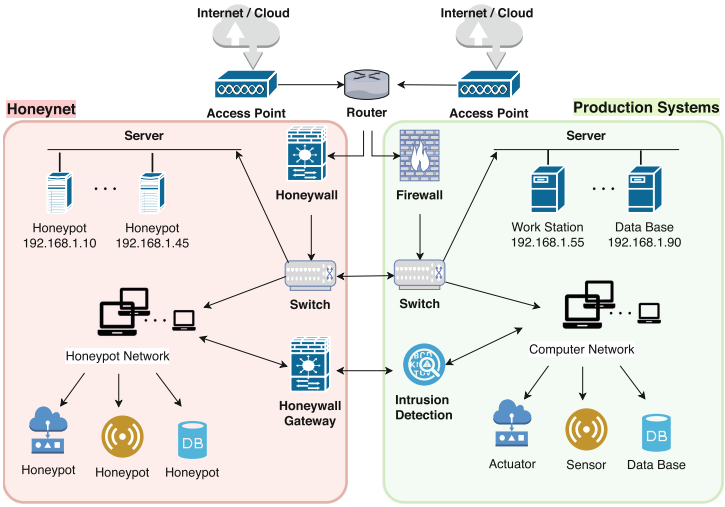

Fig. 1. The honeynet in red mimics the targeted production system in green. The honeynet shares the same structure as the production system yet has no authorized services. (Color figure online)

The current honeynet applies the honeywall as a gateway device to supervise outbound data and separate the honeynet from other production systems, as shown in Fig. 1. However, to avoid attackers' identification of the data control and the honeynet, a defender cannot block all outbound traffics from the honeynet, which leads to a trade-off between the rewards of learning high-level IoCs and the following three types of risks.

T1: Attackers identify the honeynet and thus either terminate on their own or generate misleading interactions with honeypots.
T2: Attackers circumvent the honeywall to penetrate other production systems [34].
T3: Defender's engagement costs outweigh the investigation reward.

We quantify risk T1 in Sect. 2.3, T2 in Sect. 2.5, and T3 in Sect. 2.4. In particular, risk T3 brings the problem of timeliness and optimal decisions on timing. Since a persistent traffic generation to engage attackers is costly and the defender aims to obtain timely threat information, the defender needs cost-effective policies to lure the attacker quickly to the target honeypot and reduce attacker's sojourn time in honeypots of low-investigation value.

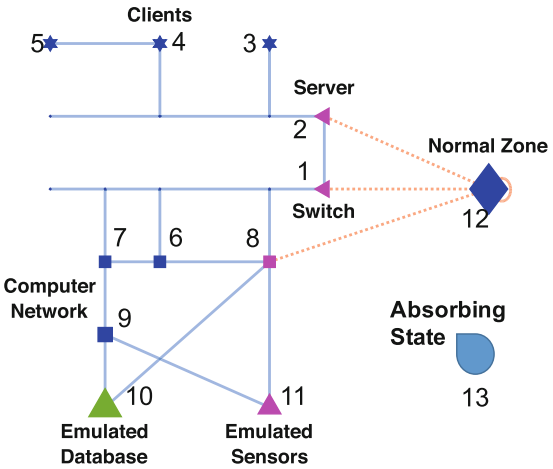

Fig. 2. Honeypots emulate different components of the production system. (Color figure online)

To achieve the goal of long-term, cost-effective policies, we construct the Semi-Markov Decision Process (SMDP) in Sect. 2 on the network shown in Fig. 2. Nodes 1 to 11 represent different types of honeypots, nodes 12 and 13 represent the domain of the production system and the virtual absorbing state, respectively. The attacker transits between these nodes according to the network topology in Fig. 1 and can remain at different nodes for an arbitrary period of

time. The defender can dynamically change the honeypots' engagement levels such as the amount of outbound traffic, to affect the attacker's sojourn time, engagement rewards, and the probabilistic transition in that honeypot.

In Sect. 3, we define security metrics related to our attacker engagement problem and analyze the risk both theoretically and numerically. These metrics answer important security questions in the honeypot engagement problem as follows. How likely will the attacker visit the normal zone at a given time? How long can a defender engage the attacker in a given honeypot before his first visit to the normal zone? How attractive is the honeynet if the attacker is initially in the normal zone? To protect against the Advanced Persistent Threats (APTs), we further investigate the engagement performance against attacks of different levels of persistence and intelligence.

Finally, for systems with a large number of governing random variables, it is often hard to characterize the exact attack model, which is referred to as the *curse of modeling*. Hence, we apply reinforcement learning methods in Sect. 4 to learn the attacker's behaviors represented by the parameters of the SMDP. We visualize the convergence of the optimal engagement policy and the optimal value in a video demo[1]. In Sect. 4.1, we discuss challenges and future works of reinforcement learning in the honeypot engagement scenario where the learning environment is non-cooperative, risky, and sample scarce.

1.1 Related Works

Active defenses [23] and defensive deceptions [1] to detect and deter attacks have been active research areas. Techniques such as honeynets [30,49], moving target defense [17,48], obfuscation [31,32], and perturbations [44,45] have been introduced as defensive mechanisms to secure the cyberspace. The authors in [11] and [16] design two proactive defense schemes where the defender can manipulate the adversary's belief and take deceptive precautions under stealthy attacks, respectively. In particular, many works [10,26] including ones with Markov Decision Process (MDP) models [22,30] and game-theoretic models [20,40,41] focus on the adaptive honeypot deployment, configuration, and detection evasion to effectively gather threat information without the attacker's notice. A number of quantitative frameworks have been proposed to model proactive defense for various attack-defense scenarios building on Stackelberg games [25,31,46], signaling games [27,29,33,42,51], dynamic games [7,15,36,47], and mechanism design theory [5,9,43,50]. Pawlick et al. in [28] have provided a recent survey of game-theoretic methods for defensive deception, which includes a taxonomy of deception mechanisms and an extensive literature of game-theoretic deception.

Most previous works on honeypots have focused on studying the attacker's break-in attempts yet pay less attention to engaging the attacker after a successful penetration so that the attackers can thoroughly expose their post-compromise behaviors. Moreover, few works have investigated timing issues and risk assessment during the honeypot engagement, which may result in an

[1] See the demo following URL: https://bit.ly/2QUz3Ok.

improper engagement time and uncontrollable risks. The work most related to this one is [30], which introduces a continuous-state infinite-horizon MDP model where the defender decides when to eject the attacker from the network. The author assumes a maximum amount of information that a defender can learn from each attack. The type of systems, i.e., either a normal system or a honeypot, determines the transition probability. Our framework, on the contrary, introduces following additional distinct features:

- The upper bound on the amount of information which a defender can learn is hard to obtain and may not even exist. Thus, we consider a discounted factor to penalize the timeliness as well as the decreasing amount of unknown information as time elapses.
- The transition probability not only depends on the type of systems but also depends on the network topology and the defender's actions.
- The defender endows attackers the freedom to explore the honeynet and affects the transition probability and the duration time through different engagement actions.
- We use reinforcement learning methods to learn the parameter of the SMDP model. Since our learning algorithm constantly updates the engagement policy based on the up-to-date samples obtained from the honeypot interactions, the acquired optimal policy adapts to the potential evolution of attackers' behaviors.

SMDP generalizes MDP by considering the random sojourn time at each state, and is widely applied to machine maintenance [4], resource allocation [21], infrastructure protection [13,13,14], and cybersecurity [38]. This work aims to leverage the SMDP framework to determine the optimal attacker engagement policy and to quantify the trade-off between the value of the investigation and the risk.

1.2 Notations

Throughout the paper, we use calligraphic letter $\mathcal{X}$ to define a set. The upper case letter X denotes a random variable and the lower case x represents its realization. The boldface $\mathbf{X}$ denotes a vector or matrix and $\mathbf{I}$ denotes an identity matrix of a proper dimension. Notation Pr represents the probability measure and $\star$ represents the convolution. The indicator function $\mathbf{1}_{\{x=y\}}$ equals one if $x = y$, and zero if $x \neq y$. The superscript k represents decision epoch k and the subscript i is the index of a node or a state. The pronoun 'she' refers to the defender, and 'he' refers to the attacker.

2 Problem Formulation

To obtain optimal engagement decisions at each honeypot under the probabilistic transition and the continuous sojourn time, we introduce the continuous-time infinite-horizon discounted SMDPs, which can be summarized by the tuple $\{t \in [0,\infty), \mathcal{S}, \mathcal{A}(s_j), tr(s_l|s_j,a_j), z(\cdot|s_j,a_j,s_l), r^{\gamma}(s_j,a_j,s_l), \gamma \in [0,\infty)\}$. We describe each element of the tuple in this section.

2.1 Network Topology

We abstract the structure of the honeynet as a finite graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$. The node set $\mathcal{N} := \{n_1, n_2, \cdots, n_N\} \cup \{n_{N+1}\}$ contains N nodes of hybrid honeypots. Take Fig. 2 as an example, a node can be either a virtual honeypot of an integrated database system or a physical honeypot of an individual computer. These nodes provide different types of functions and services, and are connected following the topology of the emulated production system. Since we focus on optimizing the value of investigation in the honeynet, we only distinguish between different types of honeypots in different shapes, yet use one extra node n_{N+1} to represent the entire domain of the production system. The network topology $\mathcal{E} := \{e_{jl}\}, j, l \in \mathcal{N}$, is the set of directed links connecting node n_j with n_l, and represents all possible transition trajectories in the honeynet. The links can be either physical (if the connecting nodes are real facilities such as computers) or logical (if the nodes represent integrated systems). Attackers cannot break the topology restriction. Since an attacker may use some honeypots as pivots to reach a production system, and it is also possible for a defender to attract attackers from the normal zone to the honeynet through these bridge nodes, there exist links of both directions between honeypots and the normal zone.

2.2 States and State-Dependent Actions

At time $t \in [0, \infty)$, an attacker's state belongs to a finite set $\mathcal{S} := \{s_1, s_2, \cdots, s_N, s_{N+1}, s_{N+2}\}$ where $s_i, i \in \{1, \cdots, N+1\}$, represents the attacker's location at time t. Once attackers are ejected or terminate on their own, we use the extra absorbing state s_{N+2} to represent the virtual location. The attacker's state reveals the adversary visit and exploitation of the emulated functions and services. Since the honeynet provides a controlled environment, we assume that the defender can monitor the state and transitions persistently without uncertainties. The attacker can visit a node multiple times for different purposes. A stealthy attacker may visit the honeypot node of the database more than once and revise data progressively (in a small amount each time) to evade detection. An attack on the honeypot node of sensors may need to frequently check the node for the up-to-date data. Some advanced honeypots may also emulate anti-virus systems or other protection mechanisms such as setting up an authorization expiration time, then the attacker has to compromise the nodes repeatedly.

At each state $s_i \in \mathcal{S}$, the defender can choose an action a_i from a state-dependent finite set $\mathcal{A}(s_i)$. For example, at each honeypot node, the defender can conduct action a_E to eject the attacker, action a_P to purely record the attacker's activities, low-interactive action a_L, or high-interactive action a_H to engage the attacker, i.e., $\mathcal{A}(s_i) := \{a_E, a_P, a_L, a_H\}, i \in \{1, \cdots, N\}$. The high-interactive action is costly to implement yet both increases the probability of a longer sojourn time at honeypot n_i, and reduces the probability of attackers penetrating the normal system from n_i if connected. If the attacker resides in the normal zone either from the beginning or later through the pivot honeypots, the defender can choose either action a_E to eject the attacker immediately, or action

a_A to attract the attacker to the honeynet by exposing some vulnerabilities intentionally, i.e., $\mathcal{A}(s_{N+1}) := \{a_E, a_A\}$. Note that the instantiation of the action set and the corresponding consequences are not limited to the above scenario. For example, the action can also refer to a different degree of outbound data control. A strict control reduces the probability of attackers penetrating the normal system from the honeypot, yet also brings less investigation value.

2.3 Continuous-Time Process and Discrete Decision Model

Based on the current state $s_j \in \mathcal{S}$, the defender's action $a_j \in \mathcal{A}(s_j)$, the attacker transits to state $s_l \in \mathcal{S}$ with a probability $tr(s_l|s_j, a_j)$ and the sojourn time at state s_j is a continuous random variable with a probability density $z(\cdot|s_j, a_j, s_l)$. Note that the risk T1 of the attacker identifying the honeynet at state s_j under action $a_j \neq A_E$ can be characterized by the transition probability $tr(s_{N+2}|s_j, a_j)$ as well as the duration time $z(\cdot|s_j, a_j, s_{N+2})$. Once the attacker arrives at a new honeypot n_i, the defender dynamically applies an interaction action at honeypot n_i from $\mathcal{A}(s_i)$ and keeps interacting with the attacker until he transits to the next honeypot. The defender may not change the action before the transition to reduce the probability of attackers detecting the change and become aware of the honeypot engagement. Since the decision is made at the time of transition, we can transform the above continuous time model on horizon $t \in [0, \infty)$ into a discrete decision model at decision epoch $k \in \{0, 1, \cdots, \infty\}$. The time of the attacker's k^{th} transition is denoted by a random variable T^k, the landing state is denoted as $s^k \in \mathcal{S}$, and the adopted action after arriving at s^k is denoted as $a^k \in \mathcal{A}(s^k)$.

2.4 Investigation Value

The defender gains a reward of investigation by engaging and analyzing the attacker in the honeypot. To simplify the notation, we divide the reward during time $t \in [0, \infty)$ into ones at discrete decision epochs $T^k, k \in \{0, 1, \cdots, \infty\}$. When $\tau \in [T^k, T^{k+1}]$ amount of time elapses at stage k, the defender's reward of investigation

$$r(s^k, a^k, s^{k+1}, T^k, T^{k+1}, \tau) = r_1(s^k, a^k, s^{k+1})\mathbf{1}_{\{\tau=0\}} + r_2(s^k, a^k, T^k, T^{k+1}, \tau),$$

at time τ of stage k, is the sum of two parts. The first part is the immediate cost of applying engagement action $a^k \in \mathcal{A}(s^k)$ at state $s^k \in \mathcal{S}$ and the second part is the reward rate of threat information acquisition minus the cost rate of persistently generating deceptive traffics. Due to the randomness of the attacker's behavior, the information acquisition can also be random, thus the actual reward rate r_2 is perturbed by an additive zero-mean noise w_r.

Different types of attackers target different components of the production system. For example, an attacker who aims to steal data will take intensive

adversarial actions at the database. Thus, if the attacker is actually in the honeynet and adopts the same behavior as he is in the production system, the defender can identify the target of the attack based on the traffic intensity. We specify r_1 and r_2 at each state properly to measure the risk T3. To maximize the value of the investigation, the defender should choose proper actions to lure the attacker to the honeypot emulating the target of the attacker in a short time and with a large probability. Moreover, the defender's action should be able to engage the attacker in the target honeypot actively for a longer time to obtain more valuable threat information. We compute the optimal long-term policy that achieves the above objectives in Sect. 2.5.

As the defender spends longer time interacting with attackers, investigating their behaviors and acquires better understandings of their targets and TTPs, less new information can be extracted. In addition, the same intelligence becomes less valuable as time elapses due to the timeliness. Thus, we use a discounted factor of $\gamma \in [0, \infty)$ to penalize the decreasing value of the investigation as time elapses.

2.5 Optimal Long-Term Policy

The defender aims at a policy $\pi \in \mathit{\Pi}$ which maps state $s^k \in \mathcal{S}$ to action $a^k \in \mathcal{A}(s^k)$ to maximize the long-term expected utility starting from state s^0, i.e.,

$$u(s^0, \pi) = \mathbb{E}\left[\sum_{k=0}^{\infty} \int_{T^k}^{T^{k+1}} e^{-\gamma(\tau+T^k)}(r(S^k, A^k, S^{k+1}, T^k, T^{k+1}, \tau) + w_r)d\tau\right].$$

At each decision epoch, the value function $v(s^0) = \sup_{\pi \in \mathit{\Pi}} u(s^0, \pi)$ can be represented by dynamic programming, i.e.,

$$v(s^0) = \sup_{a^0 \in \mathcal{A}(s^0)} \mathbb{E}\left[\int_{T^0}^{T^1} e^{-\gamma(\tau+T^0)} r(s^0, a^0, S^1, T^0, T^1, \tau)d\tau + e^{-\gamma T^1} v(S^1)\right]. \tag{1}$$

We assume a constant reward rate $r_2(s^k, a^k, T^k, T^{k+1}, \tau) = \bar{r}_2(s^k, a^k)$ for simplicity. Then, (1) can be transformed into an equivalent MDP form, i.e., $\forall s^0 \in \mathcal{S}$,

$$v(s^0) = \sup_{a^0 \in \mathcal{A}(s^0)} \sum_{s^1 \in \mathcal{S}} tr(s^1|s^0, a^0)(r^\gamma(s^0, a^0, s^1) + z^\gamma(s^0, a^0, s^1)v(s^1)), \tag{2}$$

where $z^\gamma(s^0, a^0, s^1) := \int_0^\infty e^{-\gamma\tau} z(\tau|s^0, a^0, s^1)d\tau \in [0, 1]$ is the Laplace transform of the sojourn probability density $z(\tau|s^0, a^0, s^1)$ and the equivalent reward $r^\gamma(s^0, a^0, s^1) := r_1(s^0, a^0, s^1) + \frac{\bar{r}_2(s^0, a^0)}{\gamma}(1 - z^\gamma(s^0, a^0, s^1)) \in [-m_c, m_c]$ is assumed to be bounded by a constant m_c.

A classical regulation condition of SMDP to avoid the probability of an infinite number of transitions within a finite time is stated as follows: there exists constants $\theta \in (0, 1)$ and $\delta > 0$ such that

$$\sum_{s^1 \in \mathcal{S}} tr(s^1|s^0, a^0) z(\delta|s^0, a^0, s^1) \leq 1 - \theta, \forall s^0 \in \mathcal{S}, a^0 \in \mathcal{A}(s^0). \tag{3}$$

It is shown in [12] that condition (3) is equivalent to $\sum_{s^1 \in \mathcal{S}} tr(s^1|s^0, a^0)$ $z^\gamma(s^0, a^0, s^1) \in [0, 1)$, which serves as the equivalent stage-varying discounted factor for the associated MDP. Then, the right-hand side of (1) is a contraction mapping and there exists a unique optimal policy $\pi^* = arg \max_{\pi \in \Pi} u(s^0, \pi)$ which can be found by value iteration, policy iteration or linear programming.

Cost-Effective Policy. The computation result of our 13-state example system is illustrated in Fig. 2. The optimal policies at honeypot nodes n_1 to n_{11} are represented by different colors. Specifically, actions a_E, a_P, a_L, a_H are denoted in red, blue, purple, and green, respectively. The size of node n_i represents the state value $v(s_i)$.

In the example scenario, the honeypot of database n_{10} and sensors n_{11} are the main and secondary targets of the attacker, respectively. Thus, defenders can obtain a higher investigation value when they manage to engage the attacker in these two honeypot nodes with a larger probability and for a longer time. However, instead of naively adopting high interactive actions, a savvy defender also balances the high implantation cost of a_H. Our quantitative results indicate that the high interactive action should only be applied at n_{10} to be cost-effective. On the other hand, although the bridge nodes n_1, n_2, n_8 which connect to the normal zone n_{12} do not contain higher investigation values than other nodes, the defender still takes action a_L at these nodes. The goal is to either increase the probability of attracting attackers away from the normal zone or reduce the probability of attackers penetrating the normal zone from these bridge nodes.

Engagement Safety Versus Investigation Values. Restrictive engagement actions endow attackers less freedom so that they are less likely to penetrate the normal zone. However, restrictive actions also decrease the probability of obtaining high-level IoCs, thus reduces the investigation values.

To quantify the system value under the trade-off of the engagement safety and the reward from the investigation, we visualize the trade-off surface in Fig. 3. In the x-axis, a larger penetration probability $p(s_{N+1}|s_j, a_j), j \in \{s_1, s_2, s_8\}, a_j \neq a_E$, decreases the value $v(s_{10})$. In the y-axis, a larger reward $r^\gamma(s_j, a_j, s_l), j \in \mathcal{S} \backslash \{s_{12}, s_{13}\}, l \in \mathcal{S}$, increases the value. The figure also shows that value $v(s_{10})$ changes in a higher rate, i.e., are more sensitive when the penetration probability is small and the reward from the investigation is large. In our scenario, the penetration probability has less influence on the value than the investigation reward, which motivates a less restrictive engagement.

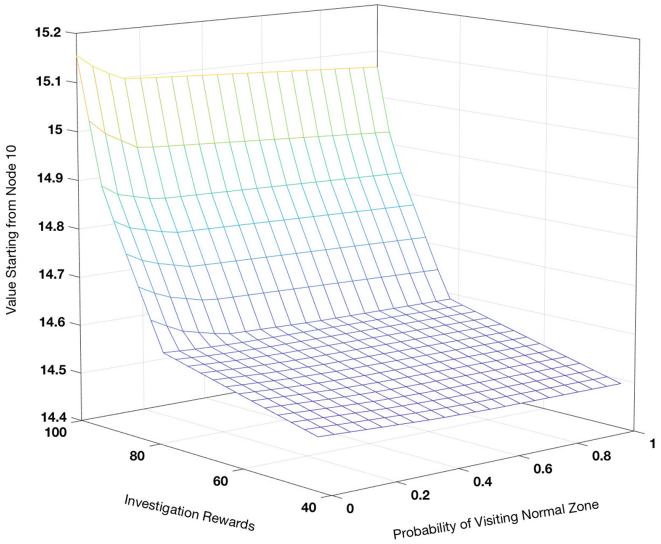

Fig. 3. The trade-off surface of $v(s_{10})$ in z-axis under different values of penetration probability $p(s_{N+1}|s_j, a_j), j \in \{s_1, s_2, s_8\}, a_j \neq a_E$, in x-axis, and the reward $r^\gamma(s_j, a_j, s_l), j \in \mathcal{S}\backslash\{s_{12}, s_{13}\}, l \in \mathcal{S}$, in y-axis.

3 Risk Assessment

Given any feasible engagement policy $\pi \in \Pi$, the SMDP becomes a semi-Markov process [24]. We analyze the evolution of the occupancy distribution and first passage time in Sects. 3.1 and 3.2, respectively, which leads to three security metrics during the honeypot engagement. To shed lights on the defense of APTs, we investigate the system performance against attackers with different levels of persistence and intelligence in Sect. 3.3.

3.1 Transition Probability of Semi-Markov Process

Define the cumulative probability $q_{ij}(t)$ of the one-step transition from $\{S^k = i, T^k = t^k\}$ to $\{S^{k+1} = j, T^{k+1} = t^k + t\}$ as $\Pr(S^{k+1} = j, T^{k+1} - t^k \leq t | S^k = i, T^k = t^k) = tr(j|i, \pi(i)) \int_0^t z(\tau|i, \pi(i), j) d\tau, \forall i, j \in \mathcal{S}, t \geq 0$. Based on a variation of the forward Kolmogorov equation where the one-step transition lands on an intermediate state $l \in \mathcal{S}$ at time $T^{k+1} = t^k + u, \forall u \in [0, t]$, the transition probability of the system in state j at time t, given the initial state i at time 0 can be represented as

$$p_{ii}(t) = 1 - \sum_{h \in \mathcal{S}} q_{ih}(t) + \sum_{l \in \mathcal{S}} \int_0^t p_{li}(t-u) dq_{il}(u),$$

$$p_{ij}(t) = \sum_{l \in \mathcal{S}} \int_0^t p_{lj}(t-u) dq_{il}(u) = \sum_{l \in \mathcal{S}} p_{lj}(t) \star \frac{dq_{il}(t)}{dt}, \forall i, j \in \mathcal{S}, j \neq i, \forall t \geq 0,$$

where $1 - \sum_{h \in \mathcal{S}} q_{ih}(t)$ is the probability that no transitions happen before time t. We can easily verify that $\sum_{l \in \mathcal{S}} p_{il}(t) = 1, \forall i \in \mathcal{S}, \forall t \in [0, \infty)$. To compute $p_{ij}(t)$ and $p_{ii}(t)$, we can take Laplace transform and then solve two sets of linear equations.

For simplicity, we specify $z(\tau|i, \pi(i), j)$ to be exponential distributions with parameters $\lambda_{ij}(\pi(i))$, and the semi-Markov process degenerates to a continuous time Markov chain. Then, we obtain the infinitesimal generator via the Leibniz integral rule, i.e.,

$$\bar{q}_{ij} := \left.\frac{dp_{ij}(t)}{dt}\right|_{t=0} = \lambda_{ij}(\pi(i)) \cdot tr(j|i, \pi(i)) > 0, \forall i, j \in \mathcal{S}, j \neq i,$$
$$\bar{q}_{ii} := \left.\frac{dp_{ii}(t)}{dt}\right|_{t=0} = -\sum_{j \in \mathcal{S} \setminus \{i\}} \bar{q}_{ij} < 0, \forall i \in \mathcal{S}.$$

Define matrix $\bar{\mathbf{Q}} := [\bar{q}_{ij}]_{i,j \in \mathcal{S}}$ and vector $\mathbf{P}_i(t) = [p_{ij}(t)]_{j \in \mathcal{S}}$, then based on the forward Kolmogorov equation,

$$\frac{d\mathbf{P}_i(t)}{dt} = \lim_{u \to 0^+} \frac{\mathbf{P}_i(t+u) - \mathbf{P}_i(t)}{u} = \lim_{u \to 0^+} \frac{\mathbf{P}_i(u) - \mathbf{I}}{u} \mathbf{P}_i(t) = \bar{\mathbf{Q}} \mathbf{P}_i(t).$$

Thus, we can compute the first security metric, the *occupancy distribution* of any state $s \in \mathcal{S}$ at time t starting from the initial state $i \in \mathcal{S}$ at time 0, i.e.,

$$\mathbf{P}_i(t) = e^{\bar{\mathbf{Q}} t} \mathbf{P}_i(0), \forall i \in \mathcal{S}. \tag{4}$$

We plot the evolution of $p_{ij}(t), i = s_{N+1}, j \in \{s_1, s_2, s_{10}, s_{12}\}$, versus $t \in [0, \infty)$ in Fig. 4 and the limiting occupancy distribution $p_{ij}(\infty), i = s_{N+1}$, in Fig. 5. In Fig. 4, although the attacker starts at the normal zone $i = s_{N+1}$, our engagement policy can quickly attract the attacker into the honeynet. Figure 5 demonstrates that the engagement policy can keep the attacker in the honeynet with a dominant probability of 91% and specifically, in the target honeypot n_{10} with a high probability of 41%. The honeypots connecting the normal zone also have a higher occupancy probability than nodes $n_3, n_4, n_5, n_6, n_7, n_9$, which are less likely to be explored by the attacker due to the network topology.

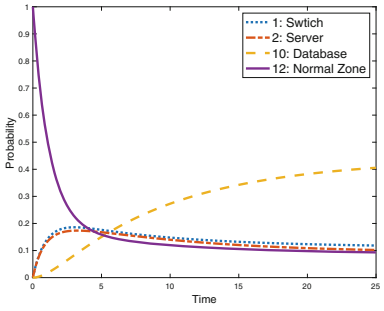

Fig. 4. Evolution of $p_{ij}(t), i = s_{N+1}$.

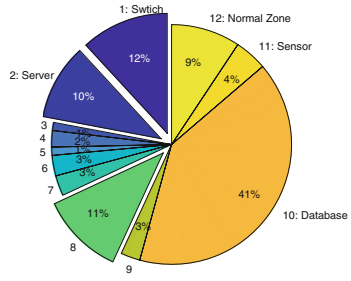

Fig. 5. The limiting occupancy distribution.

3.2 First Passage Time

Another quantitative measure of interest is the first passage time $T_{i\mathcal{D}}$ of visiting a set $\mathcal{D} \subset \mathcal{S}$ starting from $i \in \mathcal{S}\backslash\mathcal{D}$ at time 0. Define the cumulative probability function $f^c_{i\mathcal{D}}(t) := \Pr(T_{i\mathcal{D}} \leq t)$, then $f^c_{i\mathcal{D}}(t) = \sum_{h\in\mathcal{D}} q_{ih}(t) + \sum_{l\in\mathcal{S}\backslash\mathcal{D}} \int_0^t f^c_{l\mathcal{D}}(t-u)dq_{il}(u)$. In particular, if $\mathcal{D} = \{j\}$, then the probability density function $f_{ij}(t) := \frac{df^c_{ij}(t)}{dt}$ satisfies

$$p_{ij}(t) = \int_0^t p_{jj}(t-u)df^c_{ij}(u) = p_{jj}(t) \star f_{ij}(t), \forall i, j \in \mathcal{S}, j \neq i.$$

Take Laplace transform $\bar{p}_{ij}(s) := \int_0^\infty e^{-st}p_{ij}(t)dt$, and then take inverse Laplace transform on $\bar{f}_{ij}(s) = \frac{\bar{p}_{ij}(s)}{\bar{p}_{jj}(s)}$, we obtain

$$f_{ij}(t) = \int_0^\infty e^{st}\frac{\bar{p}_{ij}(s)}{\bar{p}_{jj}(s)}ds, \forall i, j \in \mathcal{S}, j \neq i. \tag{5}$$

We define the second security metric, the *attraction efficiency* as the probability of the first passenger time $T_{s_{12},s_{10}}$ less than a threshold t_{th}. Based on (4) and (5), the probability density function of $T_{s_{12},s_{10}}$ is shown in Fig. 6. We take the mean denoted by the orange line as the threshold t_{th} and the attraction efficiency is 0.63, which means that the defender can attract the attacker from the normal zone to the database honeypot in less than $t_{th} = 20.7$ with a probability of 0.63.

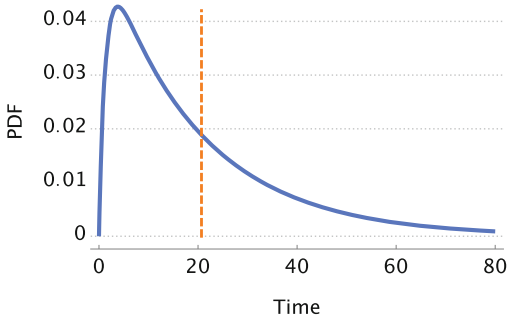

Fig. 6. Probability density function of $T_{s_{12},s_{10}}$.

Mean First Passage Time. The third security metric of concern is the *average engagement efficiency* defined as the Mean First Passage Time (MFPT) $t^m_{i\mathcal{D}} = E[T_{i\mathcal{D}}], \forall i \in \mathcal{S}, \mathcal{D} \subset \mathcal{S}$. Under the exponential sojourn distribution, MFPT can be computed directly through the a system of linear equations, i.e.,

$$\begin{aligned} &t^m_{iD} = 0, i \in \mathcal{D}, \\ &1 + \sum_{l\in\mathcal{S}} \bar{q}_{il}t^m_{l\mathcal{D}} = 0, i \notin \mathcal{D}. \end{aligned} \tag{6}$$

In general, the MFPT is asymmetric, i.e., $t^m_{ij} \neq t^m_{ji}, \forall i, j \in \mathcal{S}$. Based on (6), we compute the MFPT from and to the normal zone in Figs. 7 and 8, respectively. The color of each node indicates the value of MFPT. In Fig. 7, the honeypot nodes that directly connect to the normal zone have the shortest MFPT, and it takes attackers much longer time to visit the honeypots of clients due to the network topology. Figure 8 shows that the defender can engage attackers in the target honeypot nodes of database and sensors for a longer time. The engagements at the client nodes are yet much less attractive. Note that two figures have different time scales denoted by the color bar value, and the comparison shows that it generally takes the defender more time and efforts to attract the attacker from the normal zone.

The MFPT from the normal zone $t^m_{s_{12},j}$ measures the average time it takes to attract attacker to honeypot state $j \in \mathcal{S}\backslash\{s_{12}, s_{13}\}$ for the first time. On the contrary, the MFPT to the normal zone $t^m_{i,s_{12}}$ measures the average time of the attacker penetrating the normal zone from honeypot state $i \in \mathcal{S}\backslash\{s_{12}, s_{13}\}$ for the first time. If the defender pursues absolute security and ejects the attack once it goes to the normal zone, then Fig. 8 also shows the attacker's average sojourn time in the honeynet starting from different honeypot nodes.

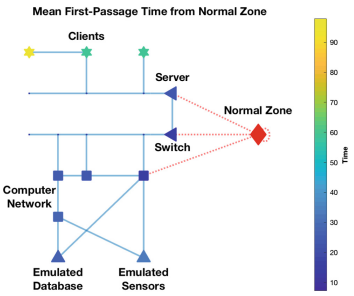

Fig. 7. MFPT from the normal zone $t^m_{s_{12},j}$.

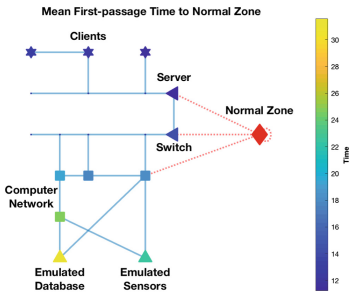

Fig. 8. MFPT to the normal zone $t^m_{i,s_{12}}$.

3.3 Advanced Persistent Threats

In this section, we quantify three engagement criteria on attackers of different levels of persistence and intelligence in Figs. 9 and 10, respectively. The criteria are the stationary probability of normal zone $p_{i,s_{12}}(\infty), \forall i \in \mathcal{S}\backslash\{s_{13}\}$, the utility of normal zone $v(s_{12})$, and the expected utility over the stationary probability, i.e., $\sum_{j\in\mathcal{S}} p_{ij}(\infty)v(j), \forall i \in \mathcal{S}\backslash\{s_{13}\}$.

As shown in Fig. 9, when the attacker is at the normal zone $i = s_{12}$ and the defender chooses action $a = a_A$, a larger $\lambda := \lambda_{ij}(a_A), \forall j \in \{s_1, s_2, s_8\}$, of the exponential sojourn distribution indicates that the attacker is more inclined to respond to the honeypot attraction and thus less time is required to attract the attacker away from the normal zone. As the persistence level λ increases from 0.1

to 2.5, the stationary probability of the normal zone decreases and the expected utility over the stationary probability increases, both converge to their stable values. The change rate is higher during $\lambda \in (0, 0.5]$ and much lower afterward. On the other hand, the utility loss at the normal zone decreases approximately linearly during the entire period $\lambda \in (0, 2.5]$.

As shown in Fig. 10, when the attacker becomes more advanced with a larger failure probability of attraction, i.e., $p := p(j|s_{12}, a_A), \forall j \in \{s_{12}, s_{13}\}$, he can stay in the normal zone with a larger probability. A significant increase happens after $p \geq 0.5$. On the other hand, as p increases from 0 to 1, the utility of the normal zone reduces linearly, and the expected utility over the stationary probability remains approximately unchanged until $p \geq 0.9$.

Figures 9 and 10 demonstrate that the expected utility over the stationary probability receives a large decrease only at the extreme cases of a high transition frequency and a large penetration probability. Similarly, the stationary probability of the normal zone remains small for most cases except for the above extreme cases. Thus, our policy provides a robust expected utility as well as a low-risk engagement over a large range of changes in the attacker's persistence and intelligence.

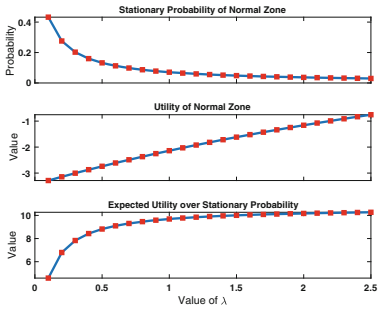

Fig. 9. Three engagement criteria under different persistence levels $\lambda \in (0, 2.5]$.

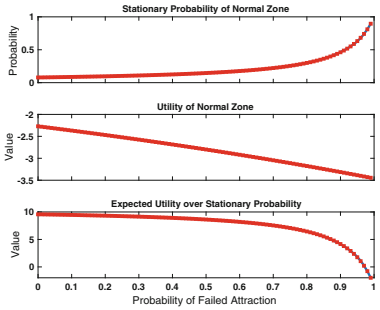

Fig. 10. Three engagement criteria under different intelligence levels $p \in [0, 1]$.

4 Reinforcement Learning of SMDP

Due to the absent knowledge of an exact SMDP model, i.e., the investigation reward, the attacker's transition probability (and even the network topology), and the sojourn distribution, the defender has to learn the optimal engagement policy based on the actual experience of the honeynet interactions. As one of the classical model-free reinforcement learning methods, the Q-learning algorithm for SMDP has been stated in [3], i.e.,

$$Q^{k+1}(s^k, a^k) := (1 - \alpha^k(s^k, a^k))Q^k(s^k, a^k) + \alpha^k(s^k, a^k)[\bar{r}_1(s^k, a^k, \bar{s}^{k+1}) + \bar{r}_2(s^k, a^k)\frac{(1 - e^{-\gamma\bar{\tau}^k})}{\gamma} - e^{-\gamma\bar{\tau}^k} \max_{a' \in \mathcal{A}(\bar{s}^{k+1})} Q^k(\bar{s}^{k+1}, a')], \quad (7)$$

where s^k is the current state sample, a^k is the current selected action, $\alpha^k(s^k, a^k) \in (0, 1)$ is the learning rate, $\bar{s}^{k+1}$ is the observed state at next stage, $\bar{r}_1, \bar{r}_2$ is the observed investigation rewards, and $\bar{\tau}^k$ is the observed sojourn time at state s^k. When the learning rate satisfies $\sum_{k=0}^{\infty} \alpha^k(s^k, a^k) = \infty, \sum_{k=0}^{\infty} (\alpha^k(s^k, a^k))^2 < \infty, \forall s^k \in \mathcal{S}, \forall a^k \in \mathcal{A}(s^k)$, and all state-action pairs are explored infinitely, $\max_{a' \in \mathcal{A}(s^k)} Q^k(s^k, a'), k \to \infty$, in (7) converges to value $v(s^k)$ with probability 1.

At each decision epoch $k \in \{0, 1, \cdots\}$, the action a^k is chosen according to the ϵ-greedy policy, i.e., the defender chooses the optimal action $arg\max_{a' \in \mathcal{A}(s^k)} Q^k(s^k, a')$ with a probability $1 - \epsilon$, and a random action with a probability ϵ. Note that the exploration rate $\epsilon \in (0, 1]$ should not be too small to guarantee sufficient samples of all state-action pairs. The Q-learning algorithm under a pure exploration policy $\epsilon = 1$ still converges yet at a slower rate.

In our scenario, the defender knows the reward of ejection action a_A and $v(s_{13}) = 0$, thus does not need to explore action a_A to learn it. We plot one learning trajectory of the state transition and sojourn time under the ϵ-greedy exploration policy in Fig. 11, where the chosen actions a_E, a_P, a_L, a_H are denoted in red, blue, purple, and green, respectively. If the ejection reward is unknown, the defender should be restrictive in exploring a_A which terminates the learning process. Otherwise, the defender may need to engage with a group of attackers who share similar behaviors to obtain sufficient samples to learn the optimal engagement policy.

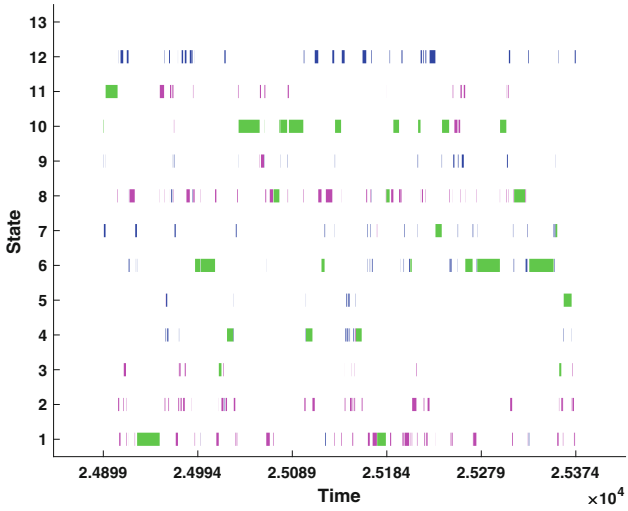

Fig. 11. One instance of Q-learning on SMDP where the x-axis shows the sojourn time and the y-axis represents the state transition. The chosen actions a_E, a_P, a_L, a_H are denoted in red, blue, purple, and green, respectively. (Color figure online)

In particular, we choose $\alpha^k(s^k, a^k) = \frac{k_c}{k_{\{s^k,a^k\}}-1+k_c}, \forall s^k \in \mathcal{S}, \forall a^k \in \mathcal{A}(s^k)$, to guarantee the asymptotic convergence, where $k_c \in (0, \infty)$ is a constant parameter and $k_{\{s^k,a^k\}} \in \{0, 1, \cdots\}$ is the number of visits to state-action pair $\{s^k, a^k\}$ up to stage k. We need to choose a proper value of k_c to guarantee a good numerical performance of convergence in finite steps as shown in Fig. 12. We shift the green and blue lines vertically to avoid the overlap with the red line and represent the corresponding theoretical values in dotted black lines. If k_c is too small as shown in the red line, the learning rate decreases so fast that new observed samples hardly update the Q-value and the defender may need a long time to learn the right value. However, if k_c is too large as shown in the green line, the learning rate decreases so slow that new samples contribute significantly to the current Q-value. It causes a large variation and a slower convergence rate of $\max_{a' \in \mathcal{A}(s_{12})} Q^k(s_{12}, a')$.

We show the convergence of the policy and value under $k_c = 1, \epsilon = 0.2$, in the video demo (See URL: https://bit.ly/2QUz3Ok). In the video, the color of each node n^k distinguishes the defender's action a^k at state s^k and the size of the node is proportional to $\max_{a' \in \mathcal{A}(s^k)} Q^k(s^k, a')$ at stage k. To show the convergence, we decrease the value of ϵ gradually to 0 after 5000 steps.

Since the convergence trajectory is stochastic, we run the simulation for 100 times and plot the mean and the variance of $Q^k(s_{12}, a_P)$ of state s_{12} under the optimal policy $\pi(s_{12}) = a_P$ in Fig. 13. The mean in red converges to the theoretical value in about 400 steps and the variance in blue reduces dramatically as step k increases.

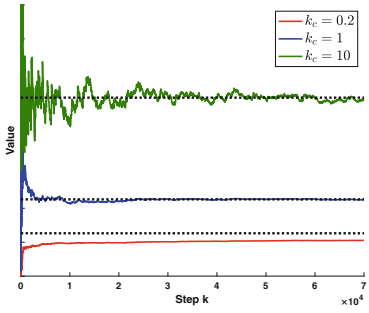

Fig. 12. The convergence rate under different values of k_c. (Color figure online)

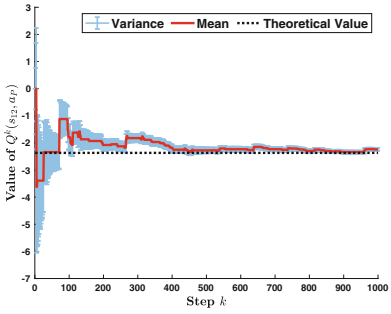

Fig. 13. The evolution of the mean and the variance of $Q^k(s_{12}, a_P)$. (Color figure online)

4.1 Discussion

In this section, we discuss the challenges and related future directions about reinforcement learning in the honeypot engagement.

Non-cooperative and Adversarial Learning Environment. The major challenge of learning under the security scenario is that the defender lacks full control of the learning environment, which limits the scope of feasible reinforcement learning algorithms. In the classical reinforcement learning task, the learner can choose to start at any state at any time, and repeatedly simulate the path from the target state. In the adaptive honeypot engagement problem, however, the defender can remove attackers but cannot arbitrarily draw them to the target honeypot and force them to show their attacking behaviors because the true threat information is revealed only when attackers are unaware of the honeypot engagements. The future work could generalize the current framework to an adversarial learning environment where a savvy attacker can detect the honeypot and adopt deceptive behaviors to interrupt the learning process.

Risk Reduction During the Learning Period. Since the learning process is based on samples from real interactions, the defender needs to concern the system safety and security during the learning period. For example, if the visit and sojourn in the normal zone bring a significant amount of losses, we can use the SARSA algorithm to conduct a more conservative learning process than Q-learning. Other safe reinforcement learning methods are stated in the survey [8], which are left as future work.

Asymptotic Versus Finite-Step Convergence. Since an attacker can terminate the interaction on his own, the engagement time with attacker may be limited. Thus, comparing to an asymptotic convergence of policy learning, the defender aims more to conduct speedy learning of the attacker's behaviors in finite steps, and meanwhile, achieve a good engagement performance in these finite steps.

Previous works have studied the convergence rate [6] and the non-asymptotic convergence [18,19] in the MDP setting. For example, [6] have shown a relationship between the convergence rate and the learning rate of Q-learning, [19] has provided the performance bound of the finite-sample convergence rate, and [18] has proposed E^3 algorithm which achieves near-optimal with a large probability in polynomial time. However, in the honeypot engagement problem, the defender does not know the remaining steps that she can interact with the attacker because the attacker can terminate on his own. Thus, we cannot directly apply the E^3 algorithm which depends on the horizon time. Moreover, since attackers may change their behaviors during the long learning period, the learning algorithm needs to adapt to the changes of SMDP model quickly.

In this preliminary work, we use the ϵ-greedy policy for the trade-off of the exploitation and exploration during the finite learning time. The ϵ can be set at a relatively large value without the gradual decrease so that the learning algorithm persistently adapts to the changes in the environment. On the other hand, the defender can keep a larger discounted factor γ to focus on the immediate investigation reward. If the defender expects a short interaction time, i.e., the

attacker is likely to terminate in the near future, she can increase the discounted factor in the learning process to adapt to her expectations.

Transfer Learning. In general, the learning algorithm on SMDP converges slower than the one on MDP because the sojourn distribution introduces extra randomness. Thus, instead of learning from scratch, the defender can attempt to reuse the past experience with attackers of similar behaviors to expedite the learning process, which motivates the investigation of transfer learning in reinforcement learning [39]. Some side-channel information may also contribute to the transfer learning.

5 Conclusion

A honeynet is a promising active defense scheme. Comparing to traditional passive defense techniques such as the firewall and intrusion detection systems, the engagement with attackers can reveal a large range of Indicators of Compromise (IoC) at a lower rate of false alarms and missed detection. However, the active interaction also introduces the risks of attackers identifying the honeypot setting, penetrating the production system, and a high implementation cost of persistent synthetic traffic generations. Since the reward depends on honeypots' type, the defender aims to lure the attacker into the target honeypot in the shortest time. To satisfy the above requirements of security, cost, and timeliness, we leverage the Semi-Markov Decision Process (SMDP) to model the transition probability, sojourn distribution, and investigation reward. After transforming the continuous time process into the equivalent discrete decision model, we have obtained long-term optimal policies that are risk-averse, cost-effective, and time-efficient.

We have theoretically analyzed the security metrics of the *occupancy distribution*, *attraction efficiency*, and *average engagement efficiency* based on the transition probability and the probability density function of the first passenger time. The numerical results have shown that the honeypot engagement can engage the attacker in the target honeypot with a large probability and in a desired speed. In the meantime, the penetration probability is kept under a bearable level for most of the time. The results also demonstrate that it is a worthy compromise of the immediate security to allow a small penetration probability so that a high investigation reward can be obtained in the long run.

Finally, we have applied reinforcement learning methods on the SMDP in case the defender can not obtain the exact model of the attacker's behaviors. Based on a prudent choice of the learning rate and exploration-exploitation policy, we have achieved a quick convergence rate of the optimal policy and the value. Moreover, the variance of the learning process has decreased dramatically with the number of observed samples.

References

1. Al-Shaer, E.S., Wei, J., Hamlen, K.W., Wang, C.: Autonomous Cyber Deception: Reasoning, Adaptive Planning, and Evaluation of HoneyThings. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-02110-8
2. Bianco, D.: The pyramid of pain (2013). http://detect-respond.blogspot.com/2013/03/the-pyramid-of-pain.html
3. Bradtke, S.J., Duff, M.O.: Reinforcement learning methods for continuous-time Markov decision problems. In: Advances in Neural Information Processing Systems, pp. 393–400 (1995)
4. Chen, D., Trivedi, K.S.: Optimization for condition-based maintenance with semi-Markov decision process. Reliab. Eng. Syst. Saf. **90**(1), 25–29 (2005)
5. Chen, J., Zhu, Q.: Security as a service for cloud-enabled internet of controlled things under advanced persistent threats: a contract design approach. IEEE Trans. Inf. Forensics Secur. **12**(11), 2736–2750 (2017)
6. Even-Dar, E., Mansour, Y.: Learning rates for Q-learning. J. Mach. Learn. Res. **5**(Dec), 1–25 (2003)
7. Farhang, S., Manshaei, M.H., Esfahani, M.N., Zhu, Q.: A dynamic Bayesian security game framework for strategic defense mechanism design. In: Poovendran, R., Saad, W. (eds.) GameSec 2014. LNCS, vol. 8840, pp. 319–328. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-12601-2_18
8. Garcıa, J., Fernández, F.: A comprehensive survey on safe reinforcement learning. J. Mach. Learn. Res. **16**(1), 1437–1480 (2015)
9. Hayel, Y., Zhu, Q.: Attack-aware cyber insurance for risk sharing in computer networks. In: Khouzani, M., Panaousis, E., Theodorakopoulos, G. (eds.) GameSec 2015. LNCS, vol. 9406, pp. 22–34. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-25594-1_2
10. Hecker, C.R.: A methodology for intelligent honeypot deployment and active engagement of attackers. Ph.D. thesis (2012). aAI3534194
11. Horák, K., Zhu, Q., Bošanskỳ, B.: Manipulating adversary's belief: a dynamic game approach to deception by design for proactive network security. In: Rass, S., An, B., Kiekintveld, C., Fang, F., Schauer, S. (eds.) GameSec 2017. LNCS, vol. 10575, pp. 273–294. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68711-7_15
12. Hu, Q., Yue, W.: Markov Decision Processes with Their Applications, vol. 14. Springer, Boston (2008). https://doi.org/10.1007/978-0-387-36951-8
13. Huang, L., Chen, J., Zhu, Q.: Distributed and optimal resilient planning of large-scale interdependent critical infrastructures. In: 2018 Winter Simulation Conference (WSC), pp. 1096–1107. IEEE (2018)
14. Huang, L., Chen, J., Zhu, Q.: Factored Markov game theory for secure interdependent infrastructure networks. In: Rass, S., Schauer, S. (eds.) Game Theory for Security and Risk Management. SDGTFA, pp. 99–126. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-75268-6_5
15. Huang, L., Zhu, Q.: Adaptive strategic cyber defense for advanced persistent threats in critical infrastructure networks. ACM SIGMETRICS Perform. Eval. Rev. **46**(2), 52–56 (2018)
16. Huang, L., Zhu, Q.: A dynamic games approach to proactive defense strategies against advanced persistent threats in cyber-physical systems. arXiv preprint arXiv:1906.09687 (2019)
17. Jajodia, S., Ghosh, A.K., Swarup, V., Wang, C., Wang, X.S.: Moving Target Defense: Creating Asymmetric Uncertainty for Cyber Threats, vol. 54. Springer, New York (2011). https://doi.org/10.1007/978-1-4614-0977-9

18. Kearns, M., Singh, S.: Near-optimal reinforcement learning in polynomial time. Mach. Learn. **49**(2–3), 209–232 (2002)
19. Kearns, M.J., Singh, S.P.: Finite-sample convergence rates for q-learning and indirect algorithms. In: Advances in Neural Information Processing Systems, pp. 996–1002 (1999)
20. La, Q.D., Quek, T.Q., Lee, J., Jin, S., Zhu, H.: Deceptive attack and defense game in honeypot-enabled networks for the internet of things. IEEE Internet Things J. **3**(6), 1025–1035 (2016)
21. Liang, H., Cai, L.X., Huang, D., Shen, X., Peng, D.: An SMDP-based service model for interdomain resource allocation in mobile cloud networks. IEEE Trans. Veh. Technol. **61**(5), 2222–2232 (2012)
22. Luo, T., Xu, Z., Jin, X., Jia, Y., Ouyang, X.: IoTCandyJar: Towards an intelligent-interaction honeypot for IoT devices. Black Hat (2017)
23. Mudrinich, E.M.: Cyber 3.0: the department of defense strategy for operating in cyberspace and the attribution problem. AFL Rev. **68**, 167 (2012)
24. Nakagawa, T.: Stochastic Processes: with Applications to Reliability Theory. Springer, London (2011). https://doi.org/10.1007/978-0-85729-274-2
25. Paruchuri, P., Pearce, J.P., Marecki, J., Tambe, M., Ordonez, F., Kraus, S.: Playing games for security: an efficient exact algorithm for solving Bayesian stackelberg games. In: Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems, vol. 2, pp. 895–902. International Foundation for Autonomous Agents and Multiagent Systems (2008)
26. Pauna, A., Iacob, A.C., Bica, I.: QRASSH-a self-adaptive SSH honeypot driven by Q-learning. In: 2018 International Conference on Communications (COMM), pp. 441–446. IEEE (2018)
27. Pawlick, J., Colbert, E., Zhu, Q.: Modeling and analysis of leaky deception using signaling games with evidence. IEEE Trans. Inf. Forensics Secur. **14**(7), 1871–1886 (2018)
28. Pawlick, J., Colbert, E., Zhu, Q.: A game-theoretic taxonomy and survey of defensive deception for cybersecurity and privacy. ACM Comput. Surv. (CSUR) (2019, to appear)
29. Pawlick, J., Farhang, S., Zhu, Q.: Flip the cloud: cyber-physical signaling games in the presence of advanced persistent threats. In: Khouzani, M., Panaousis, E., Theodorakopoulos, G. (eds.) GameSec 2015. LNCS, vol. 9406, pp. 289–308. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-25594-1_16
30. Pawlick, J., Nguyen, T.T.H., Colbert, E., Zhu, Q.: Optimal timing in dynamic and robust attacker engagement during advanced persistent threats. In: 2019 17th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), pp. 1–6. IEEE (2019)
31. Pawlick, J., Zhu, Q.: A Stackelberg game perspective on the conflict between machine learning and data obfuscation. In: 2016 IEEE International Workshop on Information Forensics and Security (WIFS), pp. 1–6. IEEE (2016). http://ieeexplore.ieee.org/abstract/document/7823893/
32. Pawlick, J., Zhu, Q.: A mean-field stackelberg game approach for obfuscation adoption in empirical risk minimization. In: 2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP), pp. 518–522. IEEE (2017)
33. Pawlick, J., Zhu, Q.: Proactive defense against physical denial of service attacks using poisson signaling games. In: Rass, S., An, B., Kiekintveld, C., Fang, F., Schauer, S. (eds.) GameSec 2017. LNCS, vol. 10575, pp. 336–356. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68711-7_18

34. Pouget, F., Dacier, M., Debar, H.: White paper: honeypot, honeynet, honeytoken: terminological issues. Rapport technique EURECOM 1275 (2003)
35. Rid, T., Buchanan, B.: Attributing cyber attacks. J. Strateg. Stud. **38**(1–2), 4–37 (2015)
36. Sahabandu, D., Xiao, B., Clark, A., Lee, S., Lee, W., Poovendran, R.: DIFT games: dynamic information flow tracking games for advanced persistent threats. In: 2018 IEEE Conference on Decision and Control (CDC), pp. 1136–1143. IEEE (2018)
37. Spitzner, L.: Honeypots: Tracking Hackers, vol. 1. Addison-Wesley, Reading (2003)
38. Sun, Y., Uysal-Biyikoglu, E., Yates, R.D., Koksal, C.E., Shroff, N.B.: Update or wait: how to keep your data fresh. IEEE Trans. Inf. Theory **63**(11), 7492–7508 (2017)
39. Taylor, M.E., Stone, P.: Transfer learning for reinforcement learning domains: a survey. J. Mach. Learn. Res. **10**(Jul), 1633–1685 (2009)
40. Wagener, G., State, R., Dulaunoy, A., Engel, T.: Self adaptive high interaction honeypots driven by game theory. In: Guerraoui, R., Petit, F. (eds.) SSS 2009. LNCS, vol. 5873, pp. 741–755. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-05118-0_51
41. Wang, K., Du, M., Maharjan, S., Sun, Y.: Strategic honeypot game model for distributed denial of service attacks in the smart grid. IEEE Trans. Smart Grid **8**(5), 2474–2482 (2017)
42. Xu, Z., Zhu, Q.: A cyber-physical game framework for secure and resilient multi-agent autonomous systems. In: 2015 IEEE 54th Annual Conference on Decision and Control (CDC), pp. 5156–5161. IEEE (2015)
43. Zhang, R., Zhu, Q., Hayel, Y.: A bi-level game approach to attack-aware cyber insurance of computer networks. IEEE J. Sel. Areas Commun. **35**(3), 779–794 (2017)
44. Zhang, T., Zhu, Q.: Dynamic differential privacy for ADMM-based distributed classification learning. IEEE Trans. Inf. Forensics Secur. **12**(1), 172–187 (2017). http://ieeexplore.ieee.org/abstract/document/7563366/
45. Zhang, T., Zhu, Q.: Distributed privacy-preserving collaborative intrusion detection systems for vanets. IEEE Trans. Sig. Inf. Process. Netw. **4**(1), 148–161 (2018)
46. Zhu, Q., Başar, T.: Game-theoretic methods for robustness, security, and resilience of cyberphysical control systems: games-in-games principle for optimal cross-layer resilient control systems. IEEE Control Syst. Mag. **35**(1), 46–65 (2015)
47. Zhu, Q., Başar, T.: Dynamic policy-based IDS configuration. In: Proceedings of the 48th IEEE Conference on Decision and Control, 2009 Held Jointly with the 2009 28th Chinese Control Conference, CDC/CCC 2009, pp. 8600–8605. IEEE (2009)
48. Zhu, Q., Başar, T.: Game-theoretic approach to feedback-driven multi-stage moving target defense. In: Das, S.K., Nita-Rotaru, C., Kantarcioglu, M. (eds.) GameSec 2013. LNCS, vol. 8252, pp. 246–263. Springer, Cham (2013). https://doi.org/10.1007/978-3-319-02786-9_15
49. Zhu, Q., Clark, A., Poovendran, R., Basar, T.: Deployment and exploitation of deceptive honeybots in social networks. In: 2013 IEEE 52nd Annual Conference on Decision and Control (CDC), pp. 212–219. IEEE (2013)
50. Zhu, Q., Fung, C., Boutaba, R., Başar, T.: GUIDEX: a game-theoretic incentive-based mechanism for intrusion detection networks. IEEE J. Sel. Areas Commun. **30**(11), 2220–2230 (2012)
51. Zhuang, J., Bier, V.M., Alagoz, O.: Modeling secrecy and deception in a multiple-period attacker-defender signaling game. Eur. J. Oper. Res. **203**(2), 409–418 (2010)