Moving Target Defense under Uncertainty for Web Applications

Extended Abstract

Vignesh Viswanathan
University of Massachusetts, Amherst
Amherst, USA
vviswanathan@umass.edu

Megha Bose
International Institute of Information Technology
Hyderabad, India
megha.bose@research.iiit.ac.in

Praveen Paruchuri
International Institute of Information Technology
Hyderabad, India
praveen.p@iiit.ac.in

ABSTRACT

Moving target defense (MTD) has emerged as a key technique that can be used in various security applications to reduce the threat of attackers by taking away their ability to perform reconnaissance and exploit vulnerabilities. However, most of the existing research in the field assumes unrealistic access to information about the attacker's motivations and/or actions when developing MTD strategies. Many of the existing approaches also assume complete knowledge regarding the vulnerabilities of a particular application and how each of these vulnerabilities can be exploited by an attacker. In this work, we propose an algorithm that generates effective MTD strategies for web applications that does not rely on prior knowledge about the attackers. Our approach assumes that the only information the defender receives about its own reward function, is via interaction with the attacker in a repeated game setting. We evaluate our algorithm using data which is mined from the National Vulnerability Database to show that it matches the performance of the state of the art techniques, despite using much less information.

KEYWORDS

Moving Target Defense; Adaptive Strategy; Repeated Games

ACM Reference Format:
Vignesh Viswanathan, Megha Bose, and Praveen Paruchuri. 2022. Moving Target Defense under Uncertainty for Web Applications: Extended Abstract. In *Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), Online, May 9–13, 2022*, IFAAMAS, 3 pages.

1 INTRODUCTION

Deployment of web applications is a complex process with security playing a key role. Given the relative anonymity with which any person with an internet connection can access these applications, a number of adversaries generally exist who aim to exploit their vulnerabilities. This situation results in a game between a security engineer or analyst who is constantly looking to identify and fix these vulnerabilities while adversaries are looking to exploit them for personal gain. The growing complexity of web applications has made it harder to identify these vulnerabilities beforehand and the ample amount of time attackers have to probe for these vulnerabilities has made the situation even more complex for security analysts. The consequences of failure go beyond financial loss to the organisation and could result in a breach of privacy or a denial of service significantly harming the users of these applications.

Moving target defense offers a potential solution to handle this problem. Instead of maintaining a single implementation of the web application, analysts can maintain multiple implementations and alternate between them. The key idea behind this is to take away the advantage the attacker has i.e., to perform reconnaissance over time on a static implementation. In recent times, the research community has shown a lot of interest in this approach, with a number of works [1–5] studying its viability and suggesting methods to switch between the implementations.

Several works make use of the problem's natural model as a game between the analyst (modeled as a defender) and multiple attackers, to use game theoretic approaches to identify *switching strategies* i.e., which implementation to deploy at what phase. However, a lot of this research assumes an unrealistic amount of prior knowledge about the vulnerabilities in the implementations as well as the competency and motivation of the attacker.

2 PROBLEM FORMULATION

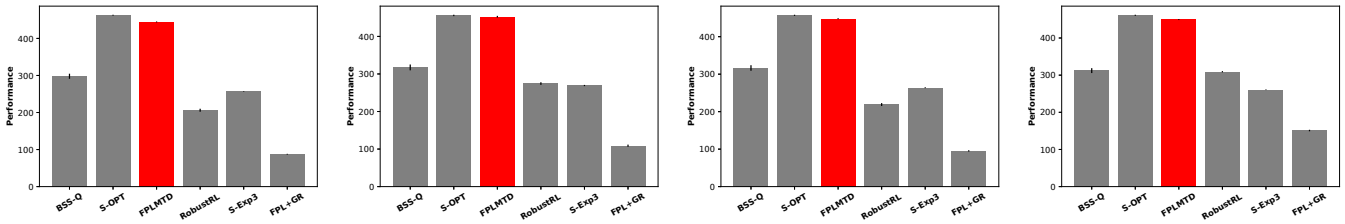
We model the setting of moving target defense as a *repeated Bayesian game* played by two different players: (a) a *defender* (denoted by θ), and (b) an *attacker* (denoted by Ψ). As commonly done in Bayesian-Stackelberg games [10], we model the attacker Ψ as having multiple types $\{\psi_1, \psi_2, \ldots, \psi_\tau\}$. The defender has a set $C = \{c_1, c_2, \ldots c_n\}$ of *n configurations* of the application it can deploy. Each configuration has a set of *vulnerabilities* $\mathcal{V}_c$.

There are T rounds in the repeated game. At each round t, an attacker of type $\psi_{f(t)}$ (where $f(\cdot)$ maps each round to an attacker type) attempts to exploit a potential vulnerability a_t of the deployed configuration $d_t \in C$. If the attacker is successful i.e. the exploited vulnerability a_t is indeed a vulnerability of the configuration d_t, then the defender receives a negative *reward* $r^t(\psi_{f(t)}, a_t, d_t) \in [-1, 0)$. The defender receives a reward of 0 otherwise. If the defender switches configurations between rounds d_t and d_{t-1}, they incurs an additional fixed *switching cost* $s(d_{t-1}, d_t) \in [0, 1]$ which is known a priori.

We are interested in developing algorithms which compute a strategy profile for the defender, $\mathcal{D} = (d_1, d_2, \ldots, d_T)$, that maximizes the reward of the defender subject to switching costs when the reward function r and the set of vulnerabilities of each configuration are not known before hand. More formally, our goal is to maximize the *total utility* of the defender:

$$TU(\mathcal{D}) = \sum_{t=1}^{T} (r^t(\psi_{f(t)}, a_t, d_t) - s(d_{t-1}, d_t))$$

Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), P. Faliszewski, V. Mascardi, C. Pelachaud, M.E. Taylor (eds.), May 9–13, 2022, Online.



(a) Best Response (b) FPL-UE (c) Stackelberg (d) Random

Figure 1: Performance of our algorithms on NVD-based data. On each of the graphs, from left to right: BSS-Q, S-OPT, FPLMTD, RobustRL, S-Exp3, FPL+GR. Performance is defined as the difference between the total utility of the algorithm and the total utility of a uniform random algorithm.

3 PROPOSED ALGORITHM

Our proposed algorithm, $FPL-MTD$ (Algorithm 1) is inspired by the *Follow the Peturbed Leader* (FPL) algorithm for the multi-armed bandit problem [7–9]. $FPL-MTD$ only assumes that at each round, the defender gets to know the reward it received from any exploited vulnerabilities during the round.

$FPL-MTD$ maintains a reward estimate for each configuration, $\hat{r}_c$, and then chooses a configuration that has the highest reward estimate (subject to minor perturbations). Our reward estimates $\hat{r}_c$ are an unbiased estimate of a natural reward estimation method:

$$\mathbb{E}[\hat{r}_c^t] = \frac{1}{t-1} \sum_{i \in [t-1]} r^i(\psi_{f(i)}, a_i, c)$$

Algorithm 1 $FPL-MTD$

Input: hyperparameters η and γ
$\hat{r}_c^1 \leftarrow 0 \quad \forall c \in C$
for t **in** 1 **to** T **do**
  Sample $q \sim \texttt{Bernoulli}(\gamma)$
  if $q = 1$ **then**
    Let d_t be a uniformly sampled configuration
  else
    Sample $z_c \sim \exp(\eta) \quad \forall c \in C$
    $u_c \leftarrow \hat{r}_c^t - z_c \quad \forall c \in C$
    $d_t \leftarrow \max_{c \in C}(u_c - s(d_{t-1}, c))$
  end if
  Adversary of unknown type $\psi_{f(t)}$ plays unknown action a_t, giving the defender a reward $r^t(\psi_{f(t)}, a_t, d_t)$
  Run GR to obtain $K(d_t)$
  $\hat{r}_d^{t+1} \leftarrow \frac{1}{t}\left[(t-1)\hat{r}_d^t + K(d_t) r^t(\psi_{f(t)}, a_t, d_t)\mathbb{I}\{d = d_t\}\right]$
end for

4 EXPERIMENTAL EVALUATION

In order to evaluate $FPL-MTD$, we randomly generate artificial web applications with vulnerabilities from the National Vulnerability Database. We then simulate our algorithm on these artificial web applications against a synthetic attacker and measure the cumulative reward. We compare our algorithm to the state of the art algorithms for the problem: $BSS-Q$ [11], $S-OPT$ [12], $RobustRL$ [14], $S-Exp3$ [6] and $FPL+GR$ [9]. $BSS-Q$ and $S-OPT$ assume knowledge of the set of vulnerabilities of each configuration along with the defender's and attacker's reward function. On the other hand, $RobustRL$, $S-Exp3$ and $FPL+GR$ use the same amount of information as $FPL-MTD$. We test all the algorithms against four attacker strategies: Best Response (also used by [13]), $FPL-UE$ [13], Stackelberg solution [10] and Uniform Random.

Algorithm 2 GR

$K(d_t) \leftarrow M$
for k **in** 1 **to** M **do**
  Follow lines 4 to 11 in Algorithm 1 to produce $\tilde{d}$ as a simulation of d_t
  if $\tilde{d} = d_t$ **then**
    $K(d_t) \leftarrow \min(K(d_t), k)$
  end if
end for
return $K(d_t)$

We present our results in Figure 1. Each graph plots the performance of each of the above mentioned algorithms against each attacker strategy. We measure the *performance* of every algorithm defined as the difference between the total utility of the algorithm and the total utility of the uniform random algorithm. The graphs in Figure 1 show a clear trend across the different attacker strategies. $S-OPT$ performs significantly well outperforming $BSS-Q$; we attribute this to the slow convergence rate of $BSS-Q$ resulting in a lack of convergence to the optimal strategy even after 25 episodes. Across all the four attacker strategies, we find that our algorithm has a performance similar to $S-OPT$ even marginally surpassing it in a few cases. Our algorithm also performs significantly better than the other baselines $RobustRL$, $S-Exp3$ and $FPL+GR$. More specifically, against the Best Response attacker strategy, $FPL-MTD$ has a performance which is 71%, 72% and 430% greater than that of $Robust-RL$, $S-Exp3$ and $FPL+GR$ respectively. We obtain similar percentages for the other attacker strategies as well (as can be seen in Figure 1).

ACKNOWLEDGMENTS

This work was performed in part using high performance computing equipment obtained under a grant from the Collaborative R&D Fund managed by the Massachusetts Technology Collaborative.

REFERENCES

[1] Ehab Al-Shaer, Qi Duan, and Jafar Haadi Jafarian. 2012. Random Host Mutation for Moving Target Defense. In *Proceedings of the 8th International Conference on Security and Privacy in Communication Networks*.

[2] Noam Ben-Asher, James Morris-King, Brian Thompson, and William Glodek. 2016. Attacker Skill, Defender Strategies, and the Effectiveness of Migration-based Moving Target Defense in Cyber Systems. In *Proceedings of the 11th International Conference on International Warfare and Security (ICCWS)*. 21–30.

[3] Guilin Cai, Baosheng Wang, Wei Hu, and Tianzuo Wang. 2016. Moving target defense: state of the art and characteristics. *Frontiers of Information Technology & Electronic Engineering* 17 (2016), 1122–1153.

[4] Jin-Hee Cho, Dilli P. Sharma, Hooman Alavizadeh, Seunghyun Yoon, Noam Ben-Asher, Terrence J. Moore, Dong Seong Kim, Hyuk Lim, and Frederica F. Nelson. 2020. Toward Proactive, Adaptive Defense: A Survey on Moving Target Defense. *IEEE Communications Surveys and Tutorials* 22 (2020), 709–745.

[5] Michael B. Crouse, E. Fulp, and D. Cañas. 2012. Improving the Diversity Defense of Genetic Algorithm-Based Moving Target Approaches. In *Proceedings of the National Symposium on Moving Target Research*.

[6] Ofer Dekel, Ambuj Tewari, and Raman Arora. 2012. Online bandit learning against an adaptive adversary: from regret to policy regret. In *Proceedings of the 29th International Conference on Machine Learning (ICML)*.

[7] James Hannan. 1958. Approximation to Bayes Risk in Repeated Play. *Contributions to the Theory of Games* 3 (1958), 97–140.

[8] Adam Kalai and Santosh Vempala. 2005. Efficient algorithms for online decision problems. *J. Comput. System Sci.* 71 (2005), 291–307.

[9] Gergely Neu and Gábor Bartók. 2016. Importance Weighting Without Importance Weights: An Efficient Algorithm for Combinatorial Semi-Bandits. *Journal of Machine Learning Research* 17 (2016), 1–21.

[10] Praveen Paruchuri, Jonathan P. Pearce, Janusz Marecki, Milind Tambe, Fernando Ordonez, and Sarit Kraus. 2008. Playing Games for Security: An Efficient Exact Algorithm for Solving Bayesian Stackelberg Games. In *Proceedings of the 7th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*. 895–902.

[11] Sailik Sengupta and Subbarao Kambhampati. 2020. Multi-agent Reinforcement Learning in Bayesian Stackelberg Markov Games for Adaptive Moving Target Defense. arXiv:2007.10457 [cs.GT]

[12] Sailik Sengupta, Satya Gautam Vadlamudi, Subbarao Kambhampati, Adam Doupé, Ziming Zhao, Marthony Taguinod, and Gail-Joon Ahn. 2017. A Game Theoretic Approach to Strategy Generation for Moving Target Defense in Web Applications. In *Proceedings of the 16th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*. 178—-186.

[13] Haifeng Xu, Long Tran-Thanh, and Nicholas R. Jennings. 2016. Playing Repeated Security Games with No Prior Knowledge. In *Proceedings of the 15th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*. 104–112.

[14] Minghui Zhu, Zhisheng Hu, and Peng Liu. 2014. Reinforcement Learning Algorithms for Adaptive Cyber Defense against Heartbleed. In *Proceedings of the 1st ACM Workshop on Moving Target Defense (MTD)*. 51–58.