A Model-Based, Decision-Theoretic Approach to Automating Cyber Response*

Lashon B. Booker and Scott A. Musman

The MITRE Corporation, 7515 Colshire Drive, McLean VA 22102, USA
booker@mitre.org, smusman@mitre.org

Abstract. Cyber-attacks can occur at machine speeds that are far too fast for human-in-the-loop (or sometimes on-the-loop) decision making to be a viable option. Although human inputs are still important, a defensive Artificial Intelligence (AI) system must have considerable autonomy in these circumstances. When the AI system is model-based, its behavior responses can be aligned with risk-aware cost/benefit tradeoffs that are defined by user-supplied preferences which capture the key aspects of how human operators understand the system, the adversary and the mission. This paper describes an innovative approach to automated cyber response that is designed along these lines. We combine a simulation of the system to be defended with an anytime online planner to solve cyber defense problems characterized as partially observable Markov decision problems (POMDPs). This technical approach appears to be a promising path toward computing tractable on-line solutions to complex cyber security problems in real-world scenarios. Ongoing work is setting the stage for deployment of this capability.

Keywords: Bayesian Reasoning, Anytime Planning, Simulation.

1 Introduction

Cyber analysts are faced with a daunting set of challenges as they try to craft responses to increasingly sophisticated cyber-attacks. Typically, analysts are overloaded with too many diverse and noisy alerts to process, making it difficult for them to adequately assess the cyber situation. This means they often must rely on incomplete and uncertain information as a starting point for making decisions about how to act. It also means that analysts can struggle to find coherent response sequences that address the broad spectrum of alerts received. In order to trace suspicious events to a root cause, it is often necessary to correlate information across multiple event streams and over multiple temporal windows. Moreover, analysts often do not understand the implication of their actions in terms of mission success or failure for the system being defended. This is all complicated by the fact that a timely response can be problematic when attacks occur

* **Approved for Public Release; Distribution Unlimited Case 20-2191; ©2020 The MITRE Corporation. ALL RIGHTS RESERVED**.

at machine speeds. Much of what occurs today relies on pre-determined responses to contingencies, seat-of-the-pants decisions, and sometimes knee-jerk reactions that may result in response actions that are worse than the attack itself.

Many applications of AI to cyber security problems are focused on helping analysts manage these challenges. There is a case to be made, though, that even with AI support, current approaches to cyber security might be overwhelmed by a new generation of AI-enabled attacks. Consequently, future systems will have to rely to some extent on automated reasoning and automated responses – with humans on the loop or out of the loop – to ensure mission success and continuously adapt to an evolving adversary.

This paper describes an approach to automating cyber response that is designed with these goals in mind. We start with the premise that, from an AI perspective, it is advantageous to frame the cyber response problem as a sequential decision-making problem under uncertainty. This leads naturally to using decision-theoretic approaches to represent the way a human operator understands the system, the adversary, and the mission; and generate responses that are aligned with risk-aware cost/benefit tradeoffs defined by user-supplied preferences.

2 Managing Uncertainty in Cyber Defense

As we have defined it, automated reasoning about cyber responses can be viewed as a form of game-playing where the defender and attacker are each afforded an opportunity to make a move. One way to account for the uncertainty about the system state and future projections is to address the cyber response problem directly as a partially observable stochastic game (e.g. as a partially observable competitive Markov decision process [1]). However, suitable state-of-the-art solution techniques for these games are only capable of solving relatively small games that must be fully specified in advance.

An alternative to a pure game-theoretic solution is to focus on resolving the defender's uncertainty about how to respond, rather than trying to solve the complete stochastic game. When the opponent's policy is fixed (either known or estimated from data), we can model a partially observable stochastic game as a partially observable Markov decision problem (POMDP) from the perspective of the protagonist [2]. The adversarial aspects of the stochastic game are incorporated into the transition function of the POMDP. This is an attractive option because recent advances in POMDP solution techniques make it possible to solve large-scale POMDPs in real time. Additionally, POMDP solvers can find policies that exploit opponent weaknesses. For these reasons, our research tackles the cyber response challenges using the formal framework of partially observable Markov decision problems[1].

[1] Note that a POMDP approach can compute the kind of general-purpose conservative solution one would expect from a game-theoretic approach if we formulate the POMDP to assume a robust adversary like a min-max opponent.

2.1 Partially Observable Markov Decision Problems (POMDPs)

Formally, a POMDP can be expressed as a tuple (**S**, **A**, **Z**, **T**, **O**, **R**) where **S** is a set of states, **A** is a set of actions, **Z** is a set of observations, **T**(s, a, s') is a transition function giving the probability p(s' | s, a) of transitioning to state s' when the agent takes action a in state s, **O**(s, a, z) is an observation function giving the probability p(z | s, a) of observing z if the agent takes action a and ends in state s, and **R**(s, a) is a reward function giving the immediate reward for taking action a in state s. The goal of the decision maker is to maximize the expected reward accrued over a sequence of actions. Since the states in a POMDP are not fully observable, the only basis for decision making is the sequence of prior actions and subsequent observations. A sufficient statistic summarizing the probability of being in a particular state, given a history of actions and observations, is called a belief, and a probability distribution over all states is called a belief state. Solving a POMDP is a planning problem that involves finding an optimal policy which maps belief states to actions.

Clearly, any search involving probabilistic belief states and arbitrarily long histories of actions and observations quickly becomes computationally intractable [3]. Although state-of-the-art offline methods for solving POMDPs have made great strides, they are not yet powerful enough to address the challenges of real-world cyber response problems. Fortunately, there are approaches available to (sometimes approximately) solve POMDPs online in real time that appear to be suitable for our purposes.

An alternative to offline planning is to select actions online, one at a time, using a fixed-horizon forward search (e.g. see [4] and [5]). Here, the key to making this idea effective for real-world problems relies on sampling the belief space, rather than fully exploring it. In particular, great efficiencies can be achieved[2] by using a black-box simulation of the problem to generate samples of possible action outcomes. The DESPOT algorithm [6] is a widely used approach that leverages simulation in this way. Moreover, DESPOT is an anytime algorithm for POMDP planning that avoids the worst-case behavior of other widely used online solution methods.

Theoretical results show that, given a suitable number of scenarios to work with, the DESPOT algorithm can reliably find near optimal policies with a regret bound that depends on the size of the optimal policy. This approach has been successfully applied to compute solutions to complex POMDP planning problems for autonomous vehicles in real time. Its performance characteristics, and its characteristics as a decision-theoretic planner [7], make this algorithm a good choice as the starting point for building a POMDP planner to address cyber security problems.

2.2 Representing Cyber Defense Problems as POMDPs

The starting point for our POMDP representation of cyber defense problems is a state representation that has been used previously [1] to model cyber defense problems as partially observable competitive Markov decision processes (POCMDP) to be solved

---

[2] A state of the art algorithm like POMCP [17] can solve POMDPs with state spaces as large as 10^{56} with only a few seconds of computation.

offline. The basic idea is to represent system state with a bit string consisting of probabilistic intrusion detection system (IDS) alert predicates for each asset.

One or more IDS sensors report the state associated with each asset, and we assume they report the binary status of an asset as either compromised or uncompromised. We currently assume these sensors operate independently and we characterize their reliability using a false negative rate and a false positive rate. Observations for the defender are binary strings showing the (possibly noisy) sensor returns.

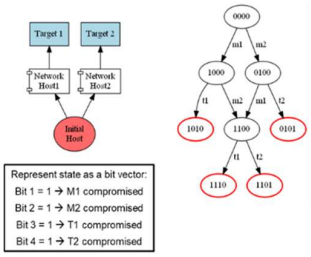


Fig. 1. Simplified micro network

Figure 1 shows a small micro network illustrating this representation. The network contains an attacker start point, and two target nodes (t1, t2) that can be compromised to cause mission impact, each with a middle node (m1, m2) separating the attacker from the target. There are four defender actions, consisting of two targeted actions (*Rm1* - reset m1; *Rm2* - reset m2), one global action (*RA* - reset all hosts), and the option to do nothing (*NOP*).

Despite its simplicity, this micro network and its associated POMDP provide a useful starting point for assessing automated response solutions since it incorporates multi-stage attacks, probabilistic actions, and uncertain sensing. Previous work has shown that optimal policies for even simple cyber security problems in this network can be surprisingly complex [8]. This can be true for scenarios involving just small amounts of sensor noise[3], different kinds of sensor noise mixed together, or sensor noise combined with uncertainty about when the attacker will make a move.

2.3 Simulating the Cyber Terrain and Attacker/Defender Interactions

The black box simulator needed for our online planning approach is provided by a modified version of the Cyber Security Game (CSG) [9]. CSG is a coarse-grained simulation of attacker and defender interactions in cyberspace. The original implementation of CSG focused on assessing defensive architectures and deploying static cyber defenses. CSG uses a cyber mission impact assessment (CMIA) model [10] [11] to translate the occurrence of incidents in cyberspace into mission outcome impacts. CSG's defensive cyber decision-making focuses primarily on defending the mission that the cyber assets are intended to support. This mission focus helps reduce the scope of the cyber defender's problem since often only a subset of the system's cyber assets is relevant at any given time.

[3] Some optimal policies for defending this small network when sensor noise is present can require policy graphs having over 100,000 nodes and more than 300,000 edges!

A challenge in cybersecurity is to be able to comprehensively consider the potential impacts of what is a staggering number of exploits and cyberattack methods[4]. To avoid having to reason about every possible attack instance, CSG reasons about the *effects* of successful attacks, rather than the attack instances themselves. The effects of cyber compromises are represented by the set of incident effects in the DIMFUI [12] taxonomy. These effects are summarized in Table 1.

Table 1. - The DIMFUI taxonomy

DIMFUI	Explanation	Typical Attacks
Degradation	1. Reduction in performance or capacity of an IT system 2. Reduction in bandwidth of a communication medium 3. Reduction in data quality	4. Limited-effect DoS 5. Zombie processes using up CPU and slowing server 6. Transfer of non-mission related data over a link that slows the transfer of mission data 7. Dropped packets cause an image to have less resolution
Interruption	IT asset becomes unusable or unavailable	1. Ping of Death 2. Wireless Jamming 3. Wipe disk
Modification	Modify data, protocol, software, firmware, component	1. Change or corrupt data 2. Modify access controls 3. Modify/Replace system files
Fabrication	Attacker inserts information into a system or fakes components	1. Replay attacks 2. DB data additions 3. Counterfeit software/ components
Unauthorized Use	Attacker uses system resources for illegitimate purposes. Related and often a precondition for other DIMFUI.	1. Access account or raise privileges in order to modify/degrade/interrupt the OS 2. Subvert service to spawn a program on remote machine 3. Bandwidth used surfing for porn degrades mission critical exchanges
Interception	Attacker gains access to information or assets used in the system	1. Keylogger 2. SQL injection 3. Crypto key theft 4. Man-in-middle attacks 5. Knowledge of component or process that is meant to be secret

The DIMFUI effects provide a robust representation of cyber incidents. They can account for every successful cyber compromise that exists in CVE, and which is described by a CAPEC attack pattern, as well as the more operational mapping of the techniques used by malicious cyber actors found in the MITRE ATT&CK [13] framework. All but one of the DIMFUI effects correspond to simple binary states of a cyber asset. This makes DIMFUI a useful abstraction that allows a cyber defender to reason only about binary representations of cyber incidents derived from the impact of 6 DIMFUI incident effects per asset, rather than hundreds or thousands of attack instances. Moreover, the DIMFUI representation aligns nicely with our approach to representing cyber defense problems as POMDPs.

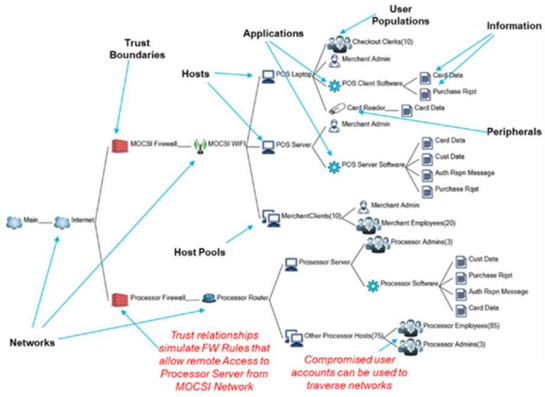


Fig. 2. - Typical details for a CSG cyber terrain model

In addition to the CMIA model, CSG also uses models of the cyber terrain and the capabilities of the attacker and defender. A typical cyber terrain model used in CSG (shown in Figure 2)

[4] e.g., The Common Vulnerabilities and Exposures (CVE) list has over 80,000 entries and the Common Attack Pattern Enumeration and Classification (CAPEC) list enumerates over 500 attack patterns [20].

consists of networks, network components (i.e. switches, routers, firewalls), hosts on the networks, user groups having access to the hosts, peripherals, applications, services and interactors that run on the host, and information used in the performing mission function. The representation of user groups, that may have access to multiple assets in the network, provides a way to simulate how compromised user credentials can be used to access hosts.

CSG was designed to represent a fully-observable, probabilistic outcome, zero-sum game for assessing the employment of static defenses. In order to use CSG with an online planner, we had to modify CSG to support queries from an external agile defender with partial and uncertain knowledge of the game state.

3 Automated Reasoning about Cyber Response

Our previous analysis of the optimal POMDP solutions for even simplified cyber security problems showed how quickly the decisions the defender must make become too complex for humans to easily develop on their own. This underscores the need for automated methods that can solve large-scale POMDPs in real time. Our work on Automated Reasoning about Cyber Response (ARCR) is a step toward addressing that need by combining the capabilities of CSG with an online planner.

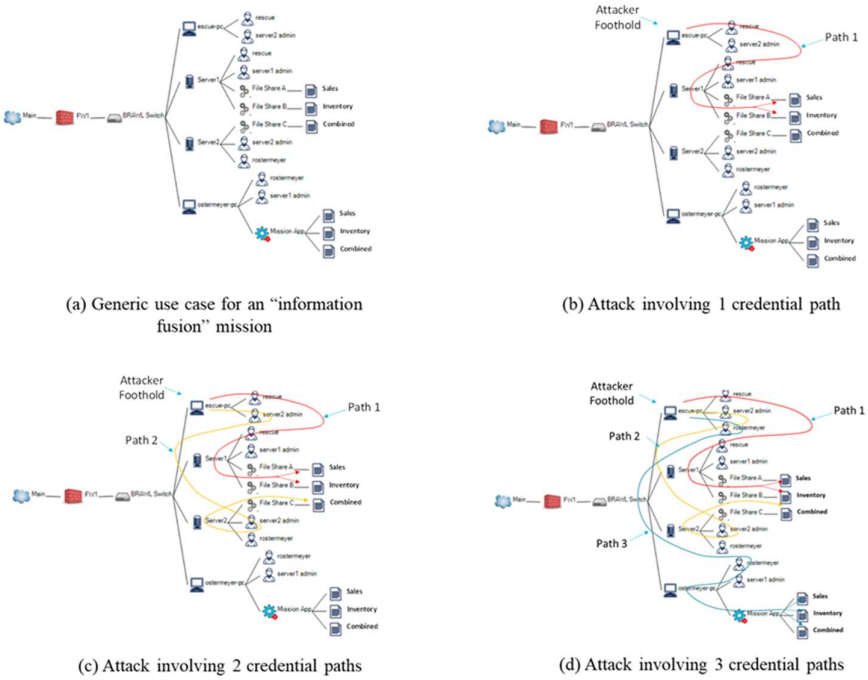
3.1 Experimental Tests of the ARCR Prototype

We used the Approximate POMDP Planning (APPL) toolkit[5] to build an online planner that employs the DESPOT algorithm. This toolkit makes it possible to implement a customized planner that includes problem-specific heuristic bounds on forward search, arbitrary representations for POMDP states, beliefs, actions and observations, and a clearly defined interface for our black-box simulator.

Our current work is applying the ARCR planner to realistic cyber defense problems that involve several DIMFUI effects. One series of simulated experimental scenarios is illustrated in Figure 3. Figure 3a shows a simple use case involving an information fusion mission. Business transaction agents (not shown) generate Sales and Inventory files that are placed in File Shares A and B respectively being served from Server 1. A client agent accesses paired Sales and Inventory files, performs some (unspecified) fusion operation on them and produces a combined status update file as an output, which is placed in Shared Folder C being served on Server 2. It is presumed that there is mission value to generating the combined status files in a timely fashion, while maintaining their integrity and confidentiality.

Experiments with this use case assumed a persistent attacker with a greedy strategy that selects the highest payoff path to a target. In the first scenario (Figure 3b), the attacker steals a user credential on its foothold, then uses that credential to move laterally from the foothold to Server 1. Once on Server 1, the attacker modifies the Sales or

[5] http://bigbird.comp.nus.edu.sg/pmwiki/farm/appl/



(a) Generic use case for an "information fusion" mission

(b) Attack involving 1 credential path

(c) Attack involving 2 credential paths

(d) Attack involving 3 credential paths

Fig. 3. Generic mission use case and cyber defense scenarios

Inventory data, thereby causing adverse impact to the mission. Assuming the available sensors do not detect that credentials are stolen but do detect the lateral move, the easiest defensive response is to eject the attacker and prevent impact by restoring the compromised host with action RX. While this response defends against the attack, it does not eliminate the threat and the attacker can simply go after the host again. If the defender is provided with an action that can disable a user account (DA), the planner can determine that the DA action completely blocks the attacker from doing any damage and is therefore the preferred solution (unless disabling the account is too costly or adversely impacts the mission). Note that because the planner is using a model-driven search, it can consider such response options and block a vulnerable credential pathway even without reliable sensor input. We are not aware of any other approach to automating cyber defense that can provide this capability in the presence of probabilistic outcome assessments and sensor noise.

It becomes a bit more complicated to determine the correct defensive response when more than one credential pathway is threatened. The scenario in Figure 3c shows an attack that, in addition to the compromised user account enabling access to Server 1, also includes a compromise of the Server 2 admin account, giving the attacker access to Server 2 and the combined status file. The planner correctly recognizes that that if the credentials are not disabled right at the beginning of this scenario, the defender will be forced to take a much more costly action later to avoid adverse mission impact.

The final scenario shown in Figure 3d illustrates how important it is for ARCR to have an appropriate model of the problem in order to be successful. We assume the attacker manages to grab three user credentials. Because there are now 3 paths to targets, but only 2 game steps needed to compromise one of them, a target may get compromised and need to be reset before the credential that accesses it can be disabled. This dilemma is a consequence of the choice to model the actions that disable accounts individually, with only one of those actions executable on a given step. A more effective model would give the planner an action that disables all compromised accounts. That easily handles the need for concurrent primitive model actions in a manner consistent with the POMDP formalism.

3.2 Steps Toward Deployment for Real Applications

While there is much we can learn about the ARCR approach in simulation, our goal is to deploy ARCR in real systems. Toward that end, we are currently implementing a test harness on virtual machines that will include an automated adversary emulator [14] as the attacker and a collection of analytics to stream the sensor information ARCR requires. This will enable us to test ARCR performance on real machines for the use

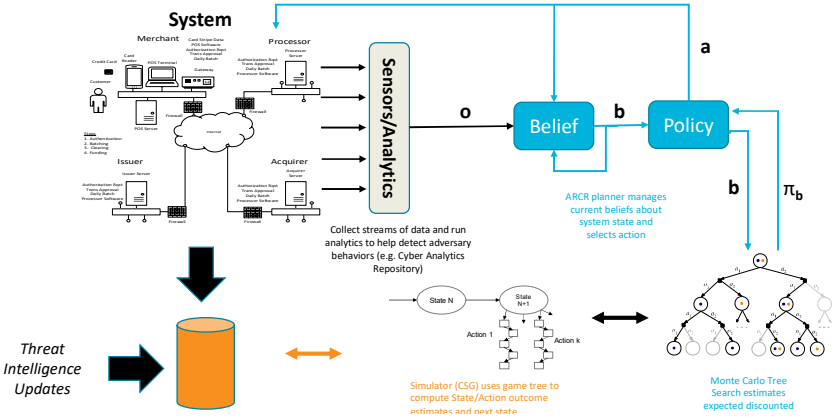


Fig. 4. Concept of operations for ARCR

cases described above and many others.

The test results for the ARCR prototype to date have demonstrated it can perform efficiently for problems of moderate[6] complexity. Work is underway on modifications to ARCR to improve its ability to handle larger scale problems. We have implemented new representations for abstract states and actions in the planner that will facilitate significant reductions in the branching factor of the belief tree. We intend to use these

[6] In simple use cases for systems with 1024 assets, the ARCR planner can compute a defensive response in less than 10 seconds on a standard laptop. When there are 512 assets, response time is under 2 seconds.

new representations as the building blocks for converting the planner into a hierarchical POMDP solver [15] [16], which will significantly speed up the search of the belief tree.

The concept of operations for using ARCR in an application is shown in Figure 4. ARCR can be applied to any cyber system and repertoire of tactics that can be modeled in our modified version of CSG. Clearly, the simulator models must be kept up-to-date to reflect changes in the system, its vulnerabilities and the attacker tools that exploit them.

4 Summary

Future systems will have to rely to some extent on automated reasoning and automated responses – with humans on the loop or out of the loop – to ensure mission success and continuously adapt to an evolving adversary.

This paper describes research suggesting that it is feasible to address this challenge by using decision-theoretic techniques to build an automated, rational AI agent that can work with human analysts to achieve shared goals in uncertain situations where the system mission is at risk. Decision-theoretic approaches can represent the way a human operator understands the system, the adversary, and the mission; and generate responses that are aligned with risk-aware cost/benefit tradeoffs defined by user-supplied preferences.

Our work on Automated Reasoning about Cyber Response (ARCR) has taken several successful steps in this direction. By framing the cyber response problem as a POMDP, we bring together state-of-the-art techniques for anytime online planning in large state spaces with the capabilities for modeling cyber security problems found in the Cyber Security Game (CSG). This combination appears to be a promising path toward computing tractable solutions to complex real-world cyber security problems.

5 References

1. S. Zonouz, H. Khurana, W. Sanders and T. Yardley, "RRE: A game-theoretic intrusion response and recovery engine.," *IEEE Transactions on Parallel and Distributed Systems,* vol. 25, no. 2, pp. 395-406, 2014.
2. F. Oliehoek, M. T. Spaan and N. Vlassis, "Best-response play in partially observable card games," in *Proceedings of the 14th Annual Machine Learning Conference of Belgium and the Netherlands*, M. van Otterlo, M. Poel and A. Nijholt, Eds., Enschede, Centre for Telematics and Information Technology (CTIT), 2005, pp. 45-50.
3. J. Pineau, G. G. and S. Thrun, "Point-based value iteration: An anytime algorithm for POMDPs," in *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, San Francisco, Morgan Kaufmann Publishers Inc., 2003, pp. 1025-1030.
4. S. Ross, J. Pineau, S. Paquet and B. Chaib-Draa, "Online planning algorithms for POMDPs," *Journal of Artificial Intelligence Research,* vol. 32, pp. 663-704, 2008.
5. R. He, E. Brunskill and N. Roy, "Efficient planning under uncertainty with macro-actions," *Journal of Artificial Intelligence Research,* vol. 40, pp. 523-570, 2011.

6. N. Ye, A. Somani, D. Hsu and W. S. Lee, "DESPOT: Online POMDP planning with regularization," *Journal of Artificial Intelligence Research,* vol. 58, pp. 231-266, 2017.
7. C. Boutilier, T. Dean and S. Hanks, "Decision-theoretic planning: Structural assumptions and computational leverage," *Journal of Artificial Intelligence Research,* vol. 11, pp. 1-94, 1999.
8. S. Musman, L. Booker, A. Applebaum and B. Edmonds, "Steps toward a principled approach to automating cyber responses," Proc. SPIE 11006, Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications, 2019.
9. S. Musman and A. J. Turner, "A game theoretic approach to cyber security risk management," *The Journal of Defense Modeling and Simulation,* vol. 15, no. 2, pp. 127-146, 2018.
10. S. Musman, A. Temin, M. Tanner, D. Fox and B. Pridemore, "Evaluating the Impact of Cyber Attacks on Missions," in *Proceedings of the 5th International Conference on Information Warfare and Security*, Dayton, OH, Academic Conferences International Limited, 2010, pp. 446-456.
11. S. Musman and A. Temin, "A Cyber Mission Impact Assessment Tool," in *Proceedings of The IEEE International Symposium on Technologies for Homeland Security (HST)*, Boston, IEEE, 2015, pp. 1-7.
12. A. Temin and S. Musman, "A language for capturing cyber impact effects," The MITRE Corporation, MITRE Technical Report MTR-100344 , Public Release Case Number 10-3793, McLean, VA, 2010.
13. MITRE Corporation, "ATT&CK," 2019. [Online]. Available: https://attack.mitre.org/. [Accessed 1 November 2019].
14. A. Applebaum, D. Miller, B. Strom, C. Korban and R. Wolf, "Intelligent, automated red team emulation," in *ACSAC '16: Proceedings of the 32nd Annual Conference on Computer Security Applications*, Los Angeles, CA: ACM, 2016, pp. 363-373.
15. N. A. Vien and M. Toussaint, "Hierarchical Monte-Carlo planning," in *Proceedings of the Twenty-Ninth Conference on Artificial Intelligence (AAAI-15)*, 2015, pp. 3613-3619.
16. A. Bai, S. Srivastava and S. J. Russell, " Markovian State and Action Abstractions for MDPs via Hierarchical MCTS," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)*, 2016, pp. 3029-3039.
17. D. Silver and J. Veness, "Monte-Carlo planning in large POMDPs," in *Advances in Neural Information Processing Systems 23 (NIPS 2010)*, Vancouver, Curran Associates, Inc., 2010, pp. 2164-2172.
18. N. P. Garg, D. Hsu and W. S. Lee, "DESPOT-Alpha: Online POMDP Planning with Large State and Observation Spaces," in *Robotics: Science and Systems XV*, Online Proceedings, ISBN 978-0-9923747-5-4, 2019, p. 6.
19. H. Kurniawati, D. Hsu and W. S. Lee, "SARSOP: Efficient Point-Based POMDP Planning by Approximating Optimally Reachable Belief Spaces," vol. IV, Online Proceedings, Robotics: Science and Systems IV, 2008, p. 9.
20. MITRE Corporation, "Making Security Measurable," 2013. [Online]. Available: https://makingsecuritymeasurable.mitre.org. [Accessed 1 November 2019].