Proceedings of the 54th Hawaii International Conference on System Sciences | 2021

The use of partially observable Markov decision processes to optimally implement moving target defense

Ashley S. M. McAbee
Naval Postgraduate School
asmcabee1@nps.edu

Murali Tummala
Naval Postgraduate School
mtummala@nps.edu

John C. McEachen
Naval Postgraduate School
mceachen@nps.edu

Abstract

For moving target defense (MTD) to shift advantage away from cyber attackers, we need techniques which render systems unpredictable but still manageable. We formulate a partially observable Markov decision process (POMDP) which facilitates optimized MTD capable of thwarting cyber attacks without excess overhead. This paper describes POMDP formulation including the use of an absorbing final state and attack penalty scaling factor to abstract defender-defined priorities into the model. An autonomous agent leverages the POMDP to select the optimal defense based on assessed cyber-attack phase. We offer an example formulation wherein attack suppression of greater than 99% and system availability of greater than 94% were maintained even as probability of detection of attack phase dropped to 74%.

1. Introduction

During the 2009 U.S. National Cyber Leap Year Summit, authorities touted moving target defense (MTD) as a game-changing cybersecurity concept that would finally reduce cyber attackers' long-held advantage [1]. MTD dynamically alters protected systems to make them less predictable and thus more difficult to attack [1]. Examples of the 90+ distinct MTD techniques include mutations of system addresses, randomization of memory layout, and variation of data formats [2]. Each technique impacts predictability; however, each can also impact the performance via imposition of overhead like temporary system outages or increased network traffic [3].

To achieve the early promise of MTD, we need techniques that amplify the unpredictability of the attack surface [3] while controlling the overhead imposed [4]. Unfortunately, these are often competing goals such that implementing useful MTD becomes an optimization effort to find the point of balance where an acceptably unpredictable and manageable system is achieved [5].

Seeking innovative ways to achieve this balance, we turned to biomimicry, the discipline of finding engineering solutions in nature [6], and examined predator-prey co-adaptation for relevant strategies because of the resemblance to the cybersecurity attacker-defender arms race [7]. Graded evasive response in which insects scale defensive response to the assessed imminence of predator attack [8] stood out as particularly well-aligned with our effort to optimize MTD. Where moths want to avoid bats for a minimum energy expenditure, cyber defenders similarly want to avoid attackers for the minimum performance sacrifice. With the moths in mind, we sought a path to implementing MTD as a graded reaction to cyber attack so that effectiveness *and* manageability are achieved.

We propose a novel approach to implementing MTD by formulation of a partially observable Markov decision process (POMDP) that reflects attack-defense dynamics, overhead of defenses, and defensive attack-risk tolerance. With these factors abstracted into a single model, an autonomous agent reasons over the model to optimally implement defensive actions in response to assessed attack imminence. The proposed MTD system is illustrated in Figure 1.

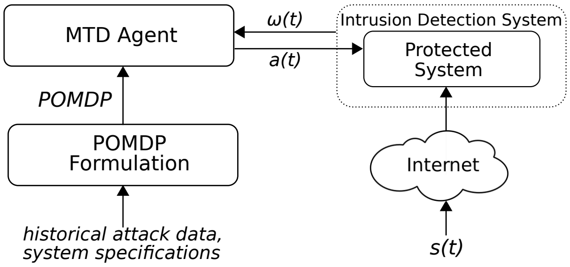
MTD Agent
ω(t)
a(t)
Intrusion Detection System
Protected System
POMDP
POMDP Formulation
Internet
historical attack data, system specifications
s(t)

Figure 1. The proposed system formulates a POMDP which the facilitates selection of the optimal action $a(t)$ based on attack phase $s(t)$.

At each time step t, the MTD agent leverages the formulated POMDP to select action $a(t)$ that thwarts the attacker effectively without undue overhead. This selection is informed by incoming observation $\omega(t)$,

URI: https://hdl.handle.net/10125/71461
978-0-9981331-4-0

which provides partial information regarding true attack phase $s(t)$. The focus of this paper is on POMDP formulation, though we give a brief example of the optimization gains possible when the complete proposed system is in place.

POMDPs are the dominant technique applied in systems that handle sequential decision-making under persistent state uncertainty [9]. While best known in the field of operations research, POMDPs have gained wider traction recently as new techniques achieve optimal decisions in increasingly complex systems [10]. At the core of the model is a finite Markov chain wherein the probability of a system being in one of a finite set of states at time t is conditioned only on the state of the system in the immediately preceding time step $t-1$. As such, this probability is called the state transition probability. The POMDP model extends the finite Markov chain to reflect two additional conditions: (1) a decision maker influences state transition probabilities in known ways by implementing one of a finite set actions and (2) true system state is only partially revealed to the decision maker via observations which occur with probabilities conditioned on true state. By applying a cost basis to states and actions reflecting the goals and priorities of the decision maker, expectations of cost under various action sequences can be explored to facilitate optimal decision making.

A POMDP is fully specified via the following components: the set of system states s, the set of available actions a, the set of observations ω, the set of state transition probability matrices P where matrix P_i describes the state transition probabilities under action a_i, the set of observation probability matrices O where matrix O_i describes observation likelihood under action a_i, and the set of cost matrices C where matrix C_i quantifies the cost of landing in each state under action a_i. In the context of our proposed system, POMDP are well suited as we require sequential selection of optimal MTD actions based on the partial observation of attack phase available from an imperfect upstream intrusion detection system (IDS).

The remainder of the paper is organized as follows: We review previous work influential to our approach in Section 2. Section 3 includes descriptions of each POMDP component with detail of what each represents in the context of the proposed system. We support our decision to adopt an absorbing final state for the model in Section 4. In Section 5, we detail an example formulation and quantify performance under the model in a simulated MTD system. Finally, Section 6 concludes the paper with a summary of our contribution and description of future work.

2. Related work

Related work falls into three categories. The first category includes other examples where POMDP was applied in a cybersecurity context, the second category supports the use of absorbing Markov chains for quantification of cybersecurity metrics, and the third category reviews other research toward optimized MTD.

We identified six examples in which POMDP was used to improve cybersecurity [11, 12, 13, 14, 10, 15]. These works were not specifically focused on MTD optimization, but aspects of their approaches to model formulation were still influential. For example, the multi-phase attack model in [11] was particularly well-aligned with the design goals of our system because MTD impacts each attack phase differently [16], and a model that represents attack phases can capture this impact. Additionally, the results in [11] support the need for appropriate treatment of incomplete observability in cybersecurity systems, as the authors' improved feedback controller based on the POMDP formulation was able to successfully reject false alarms from the IDS [11]. The ways in which variability of attackers was absorbed into the models in [15] and [10] helped develop our approach, as well.

Absorbing Markov chains became particularly important in our work. Abraham and Nair [17] also use an absorbing Markov chain to describe the dynamics of movement through an attack graph with expected path length τ used as a security metric for the network [17]. We expand upon their development to include the impact of sequential decision making and state uncertainty on τ so that the metric quantifies the impact of optimal defensive decisions.

Three works toward optimal implementation of MTD relate to and influence our proposed system [18, 4, 5]. Two of the works conduct analysis to identify static parameters for the frequency with which reconfigurations should be deployed [18, 4]. Both make strides toward more manageable MTD, but [4] was particularly influential on our proposed system because the authors look at how effectiveness and overhead change as multiple MTD techniques are used in combination. Where [19] minimized cost by identifying static reconfiguration timing parameters based on attack trends, we further reduce overhead by using real-time assessments of attack phase to dynamically trigger reconfiguration when threat justifies expense.

The final work explores an intelligent optimization system that triggers address reconfigurations both randomly and in reaction to detected attacker progress and changing system priorities much like the one we propose. DeLoach et al. [5] propose three runtime models to represent system performance constraints, available assets, and vulnerabilities. These models are reasoned over to identify optimal defensive reconfiguration timing. The research is complimentary to ours in that we both use stochastic models to control implementation of MTD under conditions of uncertainty regarding attack detection [5]. Our system differs from that of [5] in that we replace the three runtime models with a single POMDP.

3. POMDP model specification

The POMDP model is the core of the proposed system. Thus, the description of the system must begin by describing how the cyber attack-defense process is abstracted into the model. The POMDP components $\{s, a, \omega, P, C, O, \gamma\}$ are defined as listed in Table 1.

The components in the left hand column are generic to any POMDP [9], while the descriptions in the center column describe the way each component is defined to achieve our design goal of optimized MTD.

The proposed model takes advantage of the distinct phases exhibited in cyber attacks, which are each partially observable by the defender [20]. For the family of MTD techniques that carry per-reconfiguration overhead, the total cost of operating the system is directly proportional to the number of reconfigurations that occur. Thus, the MTD agent minimizes overall costs without sacrificing defensive effectiveness by keeping the most expensive defenses in reserve until the later phases of attack when the such cost is justified by the increased likelihood that the attack goal will be reached.

Each POMDP component is derived from analysis of available data related to attack, operation, and defense of the protected system. Examples of such data include forensics from previous attacks against similar systems, system specifications and requirements, and traffic analysis of the defended system. The formulation process can be broken up into four channels of attack analysis, defense analysis, prioritization of competing requirements, and assessment of the upstream IDS, as illustrated in Figure 2. The next subsections give more insight into each channel.

3.1. Attack analysis

The goal of attack analysis is to identify state vector s, observation vector ω, and transition probability matrix P_1. In our model, the phases of cyber attack become the states of a Markov chain as illustrated in Figure 3. While the model can have as few as two states, achieving our stated design goals requires inclusion of the intermediate attack phases which MTD is designed to impact. As such, sources for selecting the states of a model include both attack forensics and MTD specifications. The state vector $s = [s_1, s_2, ..., s_n]$ is a discrete list of the n phases. State s_1 is the earliest phase of the attack, and s_n represents the ultimate attack goal. The intermediate states represent incremental progress toward s_n, which may be skipped, but are ordered from 1 to n such that the imminence of attack can be inferred from the index.

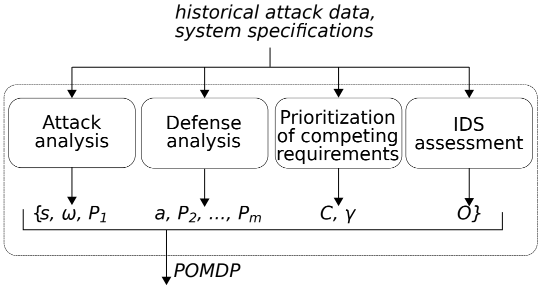

Figure 2. POMDP formulation process

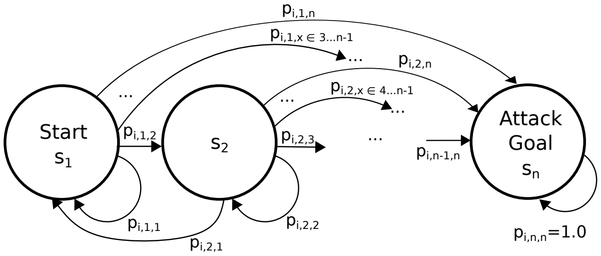

Figure 3. A Markov chain with state transition probabilities P_i describes the system dynamics under action a_i.

The next step is to determine a transition basis for the system in question. Event-based examples include per-connection, per-session, or per-IDS-alert. Time-based transitions are also possible.

We leverage attack forensics to estimate the state transition probabilities inherent to the attack process itself, absent any influence of the defender. To be most powerful, these probabilities are derived from forensic analysis of attacks against similar targets. When that information is unobtainable, insight available from ethical hackers and cybersecurity practitioners is leveraged. Examples of published resources useful for this process include cyber threat intelligence reports, cybersecurity industry white papers, and academic research papers.

Probability set P contains a set of m probability matrices, with probability matrix P_i describing the

Table 1. POMDP formulation

Component	Description	Size
state vector (s)	list of attack phases	$1 \times n$
action vector (a)	list of MTD	$1 \times m$
observation vector (ω)	list of IDS indications ($s \equiv \omega$)	$1 \times n$
probability matrix (P_i)	likelihood that system transitions between attack phases under MTD a_i,combining m P_i together, $P = \{P_1, P_2, ..., P_m\}$	$n \times n$
cost matrix (C_i)	overhead $(-)$ incurred for moving between any two phases under MTD a_i, combining m C_i together, $C = \{C_1, C_2, ..., C_m\}$	$n \times n$
observation matrix (O_i)	likelihood IDS indication aligns with attack phase under MTD a_i, combining m O_i together, $O = \{O_1, O_2, ..., O_m\}$	$n \times n$
discount factor (γ)	factor balancing immediate defensive overhead with long term attack penalties, $0 \leq \gamma \leq 1$	scalar

transition probabilities under MTD a_i as

$$P_i = \begin{bmatrix} p_{i,1,1} & p_{i,1,2} & \cdots & p_{i,1,n} \\ p_{i,2,1} & p_{i,2,2} & \cdots & p_{i,2,n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{i,n,1} & p_{i,n,2} & \cdots & p_{i,n,n} \end{bmatrix}$$

where $p_{i,x,y}$ describes the probability that the system transitions from s_x to s_y under action a_i. We adopt a convention in which a_1 always describes the *nil* option in which the defender takes no action at all, therefore P_1 describes the pattern of attackers' unencumbered efforts to compromise the system.

Generically, the Markov chain described by P_1 is fully connected to capture attacker ability to skip, loiter in, and revisit states. The only exception to full connectivity is s_n, as the final state is absorbing, i.e., $p_{1,n,n} = 1.0$ as described in Section 4. The other transition probabilities are estimated from occurrence counts in data of past attacks against similar devices.

The observation set ω is the discrete list of possible observations that could be received from the upstream IDS. For the proposed system, the burden of processing myriad attack indicators lies with the IDS such that possible observations are drawn from the possible attack phases, i.e., $s \equiv \omega$. These observations reflect the incomplete observability of attack phase such that individual observations may not align with true state, i.e., $s(t) \neq \omega(t)$. With s, ω, and P_1 determined, attack analysis is complete.

3.2. Defense analysis

The next step in POMDP formulation is an assessment of the available defenses to determine action vector a, the corresponding transition probability matrices in P, and the vector of defensive overhead C_{def}. The action set $a = [a_1, a_2, ..., a_m]$ is the discrete list of m available MTD. The transition probabilities under defense a_i are represented as P_i. We index defenses by either effectiveness in thwarting the attacker or overhead incurred for use. We have kept to this convention, so that a_m is the most effective defense available. We also assume a_m incurs the most overhead, as optimization goals would preclude installation of a defense that is more expensive unless that defense were also more effective.

The specific values in P_i for $1 < i \leq m$ require careful consideration of the phase-impact of defense a_i. The literature offers a starting point. For example, the catalog compiled by Ward et. al. [2] includes qualitative considerations of the phase-impact of more than 90 MTD techniques. Translating these assessments to a specific transition probability requires consideration of the technical descriptions of the defense in context of the attack model captured by s and P_1.

Defining P_i as a function of P_1 permits rapid update of the scheme when attack patterns change. Many MTD follow the general form of alternating some facet of the system between a discrete set of k choices, with repeat such that there is probability $p_s = \frac{k-1}{k}$ that the MTD succeeds in thwarting attack progress. A complimentary $p_f = \frac{1}{k}$ describes the likelihood of failure. These success and failure probabilities can be used to define P_i as a function of P_1 as

$$P_i = \begin{bmatrix} p_s & 0 & 0 & 0 & 0 \\ p_s & 0 & 0 & 0 & 0 \\ p_s & 0 & 0 & 0 & 0 \\ p_s & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & p_s \end{bmatrix} + p_f P_1 \qquad (1)$$

for $1 < i \leq m$, assuming the defense returns the system to s_1 when successful. Analysis to quantify unique values p_s and p_f must be conducted for each member of a.

Defense analysis also involves quantifying overhead c_i incurred when a_i occurs. In particular, the overhead must reflect the goals of the optimization effort. If the goal is to suppress attacks while minimizing downtime of services, c_i should reflect the downtime for deploying reconfiguration a_i. In general, one-time overhead expenses such as those required for installation do not translate into the proposed approach and must be accounted for separately. When multiple cost factors matter, the factors must be scaled and combined into a single value that reflects the prioritization of each. For the purposes of describing the proposed system, we discuss cost in terms of availability because availability offers a tangible and objective basis for comparison. Thus, c_i carries a unit of seconds and expresses the down time incurred when a_i is deployed such that $c_i \leq 0$. We define C_{def} as a $1 \times m$ vector of defensive costs to succinctly describe the MTD overhead.

3.3. Prioritization of attack prevention

We translate the defensive tolerance for attack risk and overhead into the model by determining c_{atk} such that prioritization between these competing goals is reflected. The cost set C contains m matrices wherein matrix C_i captures the costs incurred by state under action a_i, inclusive of both the defensive costs from C_{def} determined during defense analysis and the attack penalty c_{atk}. We define c_{atk} relative to the most expensive defense as $c_{atk} = \nu \max[C_{def}]$ where ν is defined as an attack penalty scaling factor. The defender selects ν so that attacks are thwarted at acceptable cost.

The cost set C contains m cost matrices, with cost matrix C_i describing the overhead incurred under MTD a_i as

$$C_i = \begin{bmatrix} c_{i,1,1} & c_{i,1,2} & \cdots & c_{i,1,n} \\ c_{i,2,1} & c_{i,2,2} & \cdots & c_{i,2,n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{i,n,1} & c_{i,n,2} & \cdots & c_{i,n,n} \end{bmatrix}$$

where $c_{i,x,y}$ is the specific cost incurred if the system transitions from s_x to s_y under MTD a_i. MTD overhead is assumed to be independent of attack phase. As such, any individual element $c_{i,x,y}$ can be found as

$$c_{i,x,y} = \begin{cases} c_i & x < n \\ c_i + c_{atk} & x = n \end{cases}$$

because the attack penalty c_{atk} is avoided if $s(t) \neq s_n$.

The defender considers the tipping points in ν that result in changes in the optimal policy Π of an equivalent Markov decision process (MDP) that is formulated identical to the POMDP except that state uncertainty is ignored, i.e., $s(t) = \omega(t)$. For MDP, Π is of size $1 \times n$. Vector Π lists the action to take in each state to incur the lowest discounted cost over the infinite horizon and can be efficiently found via dynamic programming techniques [21].

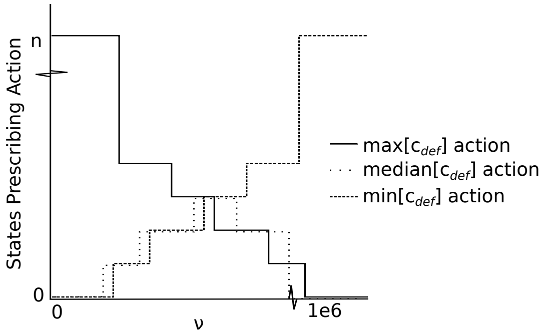

Figure 4. As ν increases, the proportion of states using the most costly defenses increases.

As ν increases, preventing attack becomes increasingly influential in minimizing the discounted cost of a given policy until attack prevention becomes the only influential factor. At that point, the system will conduct expensive but effective reconfigurations regardless of attack phase. Analysis of Π over a range of ν values will identify up to $n \times m$ points where ν drives a change in Π, as more expensive but effective defenses are used in increasingly more states.

These shifts are illustrated for a generic system in Figure 4. The vertical axis tracks the total number states in which a given action is applied. The horizontal axis indicates ν. Because costs are imposed as negative values, the $\max[c_{def}]$ action is the least expensive available defense. Under our convention that cost, effectiveness, and index are directly proportional such that the defense with the least cost is also the least adept at thwarting attacks, at $\nu = 0$, there is no attack penalty applied, and therefore the least-cost action is applied in n states. As $\nu \to \infty$, attack is weighted so heavily that a_m is applied in all n states. Given the interest in optimization prerequisite for considering the proposed system, the defender seeks to identify ν in the mid-range of values such that the most expensive defenses are used against imminent attack.

The discount factor γ influences the priority of immediate versus future rewards. The factor falls in the range $[0, 1]$, with the minimum value 0 resulting in future costs being ignored, and the maximum value 1 representing an equal emphasis on present and future costs. In this application, attack suppression requires consideration of future costs as c_{atk} is only incurred in s_n. Overhead control, on the other hand, requires emphasis on present costs. With both in play, γ sensitivity is reviewed similar to the way in which ν sensitivity was explored to find the tipping points and select that which best reflects defender priorities.

3.4. IDS assessment

The final component of POMDP formulation involves assessment of the upstream IDS to determine the probability of detection p_D, i.e., the probability that $\omega(t) = s(t)$. Observation set O contains m observation matrices denoted by O_i which contain the likelihood of alignment between $\omega(t)$ and $s(t)$ as

$$O_i = \begin{bmatrix} o_{i,1,1} & o_{i,1,2} & \cdots & o_{i,1,n} \\ o_{i,2,1} & o_{i,2,2} & \cdots & o_{i,2,n} \\ \vdots & \vdots & \ddots & \vdots \\ o_{i,n,1} & o_{i,n,2} & \cdots & o_{i,n,n} \end{bmatrix}$$

where $o_{i,x,y}$ describes the probability that observation ω_y occurs in state s_x if action a_i is taken. We introduce the general form of O to facilitate consideration of the way individual defenses may improve or degrade the ability to discern true system state, but so far our work only considers cases where defense and observation processes are independent such that O contains m identical members.

Assessment must be made to determine where the error falls by state. These values are collected into the observation matrix O_i, which is $n \times n$ in size and reflects the probability of receiving observation x in state y conditioned on action a_i as $o_{i,x,y}$. Probability p_D as well as the probabilities of false alarm p_{FA} and missed detection p_M must be determined, usually from tests of the IDS under conditions where attack activity is well understood. Our work so far considers optimization under intrusion detection performance of the form

$$O_i = \begin{bmatrix} p_D & p_{FA} & \cdots & 0 & 0 \\ p_M & p_D & p_{FA} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \vdots & p_M & p_D & p_{FA} \\ 0 & 0 & \cdots & p_M & p_D \end{bmatrix}$$

in which p_{FA} and p_M are restricted to immediate neighboring states. Together, the components $\{s, a, \omega, P, C, O, \gamma\}$ form the POMDP integral to our proposed system.

4. Importance of the absorbing state

Although recovery after attack is represented in all but one of the POMDP-based works discussed in Section 2, we found that including the recovery process within the POMDP was problematic for two reasons. First, recovery is difficult to accurately represent as a stochastic process. Once an exploit lands on a device, recovery involves a manual process dependent on the extent of the damage. Recovery can take weeks. Second, in initial exploration of performance of our MTD agent, we found that inclusion of the recovery process in the model led to undesirable a_1 action selection in the high-risk states closest to s_n. Just when defenses were needed most, they failed to deploy.

To overcome both of these issues, we adopt an absorbing state at s_n such that $p_{i,n,n} = 1.0$ for $1 \leq i \leq m$. An illustration of the influence of the absorbing state on optimal decision-making is presented in Figure 5. These box plots contain the projected costs of all possible discounted futures when either a_1:*nil* or a_2:*def* is taken in the next time step. In both scenarios (a) and (b), the system is currently in penultimate state s_{n-1} and $p_{2,(1,\ldots,n-1),1} = 1.0$ such that selecting action a_2:*def* always returns the system to safety. In scenario (a), the model includes recovery such that $p_{i,n,1} = 1.0$. In scenario (b), attack recovery is not possible as s_n is an absorbing sate, i.e., $p_{i,n,n} = 1.0$.

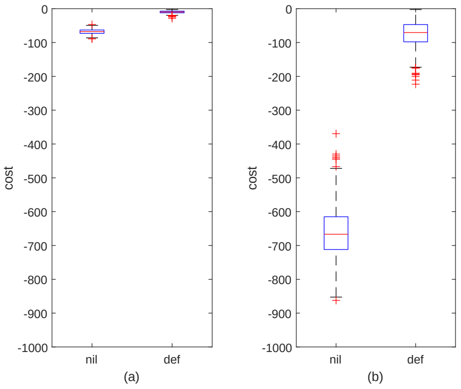

Figure 5. As compared to (a), incorporation of an absorbing state in (b) spreads projected costs under either action ten-fold to ensure achievement of optimization goals.

The optimal decision in either scenario is the one that minimizes cost. In both cases, a_2:*def* is the better choice, indicated by the *def* box plots being closest to zero, but the wider spread between *nil* and *def* costs in scenario (b) sets better conditions for the simulation-based approach used by our MTD agent to select the optimal decision.

Our POMDP formulation with the absorbing end state is similar to the goal attack states in [10], with a key difference: in our model, lower numbered states may be revisited from any state except s_n, while the model in [10] assumes monotonicity in that even intermediate phases toward attack accomplishment cannot be reverted once achieved. Because our system is focused on optimizing MTD, the monotonicity assumption no longer holds [5]. MTD introduces uncertainty for attackers with the specific intent of forcing them to revisit earlier attack phases. Thus, a model of MTD requires incorporation of backwards potential progress through state s_{n-1}.

Once the absorbing state was adopted, we found it useful in establishing metrics for system performance. Under absorbing Markov chain theory, P_1 can be used to understand qualities of the underlying attack process. Given the standard form of P_1, partition Q_1 is the upper left $n-1 \times n-1$ partition of transient state transition probabilities [22]. The fundamental matrix $N_1 = (I - Q_1)^{-1}$ where I is an identify matrix of size $n-1$ [22]. We assume the system begins in s_1 represented by starting transient state probability vector $\alpha = [1, 0, ..., 0]$. Therefore, the total expected number of state visits is found via

$$\tau_1 = \alpha N_1 \xi \tag{2}$$

where ξ is a column vector of $n-1$ entries of 1 [22]. Serving as a baseline, τ_1 is a critical metric. We define attack suppression under the proposed system ϕ relative to τ_1 as $\phi = (1 - \frac{\tau_1}{\tau_{\Pi}}) \times 100\%$ where τ_{Π} describes the steps before reaching s_n under optimal defense.

5. Results

In this section we step through POMDP formulation toward optimal defense against the five stage attack diagrammed in Figure 6. States $\{s_1, s_2, s_3, s_4, s_5\}$ correspond to the progression from *start* to *attacked*. We selected this five-phase attack model for the example formulation because a similar model was used to study the effectiveness and overhead of MTD in [4], which will permit comparison between our approaches. Following formulation, we offer results from simulated operation of the proposed system to demonstrate the value of the proposed POMDP-based approach to MTD optimization.

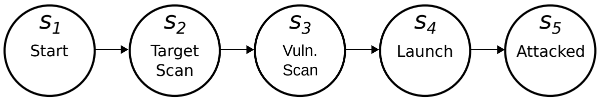

Figure 6. Five-phase attack process

In *start*, s_1, no attacker is yet working against the system. Next, *target scan*, s_2, represents the initial efforts performed by the attacker to locate the system. An example of this type of activity is an internet control message protocol (ICMP) *ping* sweep used by an attacker to identify all hosts in range of network addresses. If the system enters this phase, activity has been detected that indicates an attacker has located the system. The next phase is *vulnerability scan*, s_3, which represents technical reconnaissance efforts like operating system fingerprinting and similar techniques which can identify particular vulnerabilities of the protected system. Finally, in *launch*, s_4, the attacker actually attempts a compromise, which, if successful, results in the protected system entering s_5, *attacked.*

To estimate the values in P_1, we translated honeypot data into a Markov chain. The data used are published in [23] and were collected over 48-days as attackers interacted with two honeypot web servers behind a university firewall. Our attack analysis assumed that the event counts are gathered under $p_D = 1.0$ such that the reported attack progressions reflect P_1, not O.

The authors of [23] categorized activity as one of four different events based on a count of packets-per-connection. These classifications were possible because the honeypots served no true function, and consequently there was no valid reason to communicate with them. Two protocols were identified in traffic, namely ICMP and transmission control protocol (TCP). All ICMP traffic was assumed to be scanning activity, regardless of packet volume. The TCP traffic was of three types, with the lowest volume connections labeled as port scans, intermediate volume as vulnerability scans, and high volume as launches.

Over the 48 day window, 59,468 connections were collected, 22,710 of which went to the two honeypots. Of those honeypot connections, 6,203 unique records occurred, representing 5,540 individual attacks. To align the available data in [23] with our five state attack model, we considered ICMP and port scans both indications of s_2. Under the assumption that the attack stages may be skipped but not reordered from $\{$*start, target scan, vulnerability scan, launch*$\}$, the tallied events from [23] translate into the Markov chain in Figure 7 with values from Table 2. These probabilities assume that attacks occur via consecutive connections. State transitions occur on a per-connection basis between clients and the protected server.

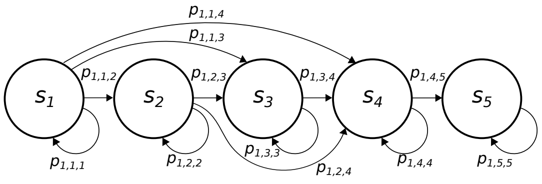

Figure 7. Markov chain modeling stochastic nature of cyber attacks, developed from data in [23].

We extend the model to the fifth state to account for the failure of exploits to take effect and cause damage, which was not quantified in [23]. In the presence of layered defense, $p_{1,4,5}$ represents the probability of all other defenses (e.g., anti-virus software, privilege control, security training) failing. We apply a value of 0.5 to represent the likelihood another layer of defense prevents entry into s_n. Together, the probabilities form transition matrix

$$P_1 = \begin{bmatrix} 0.6611 & 0.2300 & 0.0856 & 0.0233 & 0 \\ 0 & 0.9235 & 0.0687 & 0.0078 & 0 \\ 0 & 0 & 0.7900 & 0.2100 & 0 \\ 0 & 0 & 0 & 0.5000 & 0.5000 \\ 0 & 0 & 0 & 0 & 1.0000 \end{bmatrix}.$$

Following equation 2, there are $\tau_1 = 17.928$ expected total state visits before reaching s_5. Based on the connection rate of six per minute, this system, absent defense, enters s_5 within approximately 3 minutes.

To prevent such rapid success, we implement two MTDs. While there are many more defenses available, both from the catalog of MTD detailed in [2] and other non-MTD options like restarting resources or partitioning network connectivity, we implement just three defenses at this juncture for two reasons. First, this dimensionality can be accommodated via a variety of MTD agent techniques, which permits verification of the optimization gains possible via POMDP in general, without limiting choice of agent. Second, implementing these specific defenses facilitates direct comparison between our system and the system in [4].

When effective, each returns the system to s_1. Defense a_2 represents a dynamic platform change between $x = 3$ services with repeat wherein $x_v = 1$ are vulnerable to attack such that $p_s = \frac{x - x_v}{x}$. The next, a_3, represents a dynamic network change in IP address among $\rho = 256$ addresses with repeat such that $p_s = \frac{\rho - 1}{\rho}$. To align with the defenses implemented in [4], we implement a_2 and a_3 into our model with specifications as recorded in Table 3. We measure overhead in terms of availability such that c_i represents the loss in system availability in seconds when MTD a_i is deployed as measured in [4] assuming a 10 second inter-arrival rate between connections. Transition matrices P_2 and P_3 follow Equation 1.

Following the same flow of analysis used to determine τ_1, a_2 would extend the expected time before attack 15 times over to $\tau_2 = 105$, or $\phi_2 = 83.9\%$, while a_3 extends the expected time before attack to $\tau_3 = 5.25 \times 10^6$, or $\phi_3 = 99.\overline{9}\%$. By far the more effective defense, a_3 is also the most expensive, and system availability under such consistent use would be just 4.1%. The other options either thwart the attack moderately well, for moderate expense (a_2) or not at all, but for free (a_1).

Attack penalty scaling factor ν is determined by reviewing the locations of the policy shifts as ν increases are shown by state in Figure 8. The optimal policy vector Π_ν describes the action that should be taken in each state for a given value of ν, determined via policy iteration as implemented in [24]. For approximately $\nu \leq 100$, $\Pi_\nu = [a_1, a_1, a_2, a_3, a_1]$. For $\nu > 10^4$, $\Pi_\nu = [a_3, a_3, a_3, a_3, a_3]$, wherein the optimization effort is effectively abandoned, as the policy indicates taking the most expensive defense, a_3, in every state.

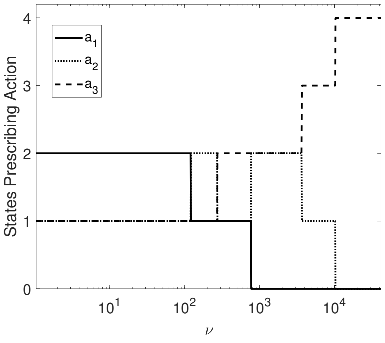

Figure 8. Policy shifts by attack penalty scaling factor ν.

These two cases represent the policies at either extreme, with the former prioritizing overhead control and the later prioritizing attack suppression. This trade-off is illustrated in Figure 9 wherein predicted metrics of attack suppression and availability as a function of ν are displayed. These expected metrics are calculated via an extension of absorbing Markov chain theory using probabilities of state and action occurrence weighted by the impact of partial observability. The stair-step shifts in value align with the policy shifts by state in Figure 8. Even for $p_D = 0.5$, attack suppression of greater than 99% is achievable, but not unless the user is willing to accept availability on the order of 2%. Because the objective of our work is to implement MTD with overhead control, we set $\nu = 100$ to explore system performance in the range where both attack suppression and availability are above 90%.

The discount factor γ influences the priority of immediate versus future rewards. The optimal policy in this case is not particularly sensitive to γ. We selected $\gamma = 0.75$ to ensure both long-term attack suppression and near-term availability were achieved, but could have selected any value in the range $0.34 \leq \gamma \leq 0.99$ with no impact on Π as generated via policy iteration implemented in [24].

Because the IDS is upstream of the proposed system, we explore performance of the proposed system across

Table 2. The translation of occurrences observed in [23] into transition probabilities for a five state Markov chain

s	arrival	s_1		s_2		s_3		s_4		s_5	
	count	count	$p_{1,i,1}$	count	$p_{1,i,2}$	count	$p_{1,i,3}$	count	$p_{1,i,4}$	count	$p_{1,i,5}$
s_1	16,347	10,807	0.661	3,760	0.23	1,399	0.086	381	0.0230	0	0.0
s_2	3,760	0	0.0	3,473	0.924	258	0.069	29	0.008	0	0.0
s_3	1,657	0	0.0	0	0.0	1,307	0.789	350	0.211	0	0.0
s_4	760	0	0.0	0	0.0	0	0.0	380	0.5	380	0.5
s_5	380	0	0.0	0	0.0	0	0.0	0	0.0	380	1.0

Table 3. Available MTD, specifications adapted from [4].

a_i	Basis	Effectiveness		Overhead	
		p_s	(%)	C_i (sec)	Avail. (%)
a_1	No Action	0	0	0.000	100%
a_2	Service	$\frac{2}{3}$	66.7%	-0.635	93.7%
a_3	IP address	$\frac{255}{256}$	99.6%	-9.590	4.1%

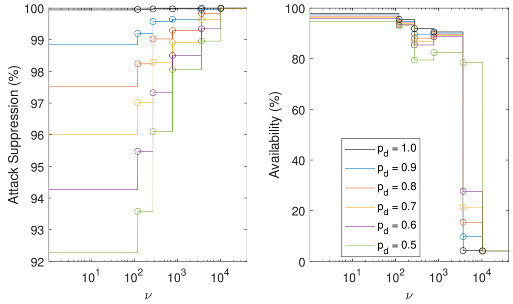

Figure 9. Sensitivity of attack suppression and availability to variation in attack penalty ν.

a range of capabilities for IDS. We consider IDS performance to be independent of MTD such that $O_i = O_j$ for $i, j \in [1, ..., m]$ and consider system performance for $0.74 \leq p_D \leq 1.00$ with error $1 - p_D$ split evenly between p_{FA} and p_{MD}.

Using the model to facilitate optimal decision-making requires an agent to estimate current system state and leverage that estimate to find the optimal action [25]. Our agent uses an online policy technique leveraging the determinized sparse partially observable tree planning (DESPOT) algorithm [26] as implemented in [27]. Attack suppression and availability performance under simulated operation of the proposed system are presented in Figure 10. We highlight the mean value across all simulations and include error bars representing the 15 and 85 quantile values for reference as to the range of performance expected. Even as p_D degrades to 0.75, the system maintains attack suppression at greater than 99% and availability at greater than 94%. As compared to repeated use of the most effective and expensive defense, a_3, the proposed system achieves nearly equivalent attack suppression while gaining upwards of 90 percentage points in availability.

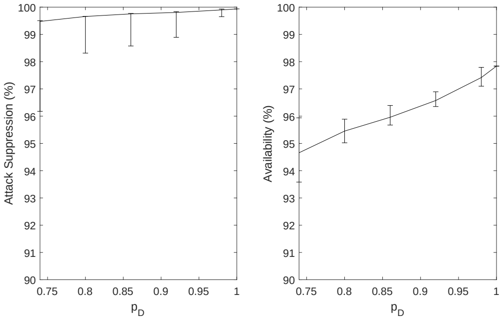

Figure 10. Availability and attack suppression performance under the proposed system.

6. Conclusion

This paper presented formulation of POMDP to facilitate optimization of MTD by assessed attack phase. Our abstraction of the defensive process into a POMDP to achieve MTD optimization goals is the main contribution of this paper. We described how our mechanism uses an attack penalty scaling factor and an absorbing ultimate state to abstract defender priorities into the model. Based on an example formulation, we quantified the significant gains in terms of system availability that can be achieved while maintaining attack suppression well beyond acceptable levels.

The most important next steps involve understanding the impact of model accuracy. The results described in this paper assume the model is perfectly aligned with the real world, so we will now work to relax this assumption and quantify the impact and source of model error for the proposed system. Examples of potential model error include misrepresentation of attacker dynamics in P_1 or defense effectiveness in P_i. Either could be devastating, with the system failing to implement defenses as needed to thwart inbound attacks. Confident adoption of the proposed system requires quantification of the tolerances for model error within which system performance remains acceptable. Further, while progress has been made, computational tractability of POMDP remains a concern [26] such that more research is needed to understand the number of MTDs and intermediate attack phases the proposed system can incorporate before the agent becomes unable to identify the optimal action on a reasonable time scale.

Even with the work remaining, optimization of MTD is critical to ensuring it becomes the invaluable cyber defense tool sought. Our work justifies continued research and investment toward model-based strategies for achieving MTD that is both manageable and unpredictable.

References

[1] U.S. Cyber Security and Information Assurance Interagency Working Group NITRD Subcommittee, "Cybersecurity research and development recommendations," 2010. [Online]. Available: https://www.nitrd.gov/Publications/PublicationDetail.aspx?pubid=24

[2] B. Ward, S. Gomez, R. W. Skowyra, D. Bigelow, J. Martin, J. Landry, and H. Okhravi, "Survey of cyber moving targets," MIT Lincoln Laboratory, Lexington, Massachusetts, Tech. Rep. 1228, Jan 2018.

[3] M. Carvalho and R. Ford, "Moving-target defenses for computer networks," *IEEE Security & Privacy*, vol. 12, no. 2, pp. 73–76, Mar 2014.

[4] W. Connell, L. H. Pham, and S. Philip, "Analysis of concurrent moving target defenses," in *Proceedings of the 5th ACM Workshop on Moving Target Defense*. New York, NY, USA: ACM, 2018, pp. 21–30.

[5] S. A. DeLoach, X. Ou, R. Zhuang, and S. Zhang, "Model-driven, moving-target defense for enterprise network security," in *Models@run.time: Foundations, Applications, and Roadmaps*, N. Bencomo, R. France, B. H. C. Cheng, and U. Aßmann, Eds. Cham: Springer International Publishing, 2014, pp. 137–161.

[6] J. M. Benyus, *Biomimicry: innovation inspired by nature*, 1st ed. New York: Perennial, 2002.

[7] W. Mazurczyk, S. Drobniak, and S. Moore, "Towards a systematic view on cybersecurity ecology," *CoRR*, vol. abs/1505.04207, 2015. [Online]. Available: http://arxiv.org/abs/1505.04207

[8] H. M. ter Hofstede and J. M. Ratcliffe, "Evolutionary escalation: the bat–moth arms race," *J. of Experimental Biology*, vol. 219, no. 11, pp. 1589–1602, 2016.

[9] M. J. Kochenderfer, *Decision making under uncertainty: theory and application*. Cambridge, Massachusetts: MIT Press, 2015.

[10] E. Miehling, M. Rasouli, and D. Teneketzis, "A POMDP approach to the dynamic defense of large-scale cyber networks," *IEEE Trans. on Information Forensics and Security*, vol. 13, no. 10, pp. 2490–2505, Oct 2018.

[11] O. P. Kreidl and T. M. Frazier, "Feedback control applied to survivability: a host-based autonomic defense system," *IEEE Trans. on Reliability*, vol. 53, no. 1, pp. 148–166, March 2004.

[12] C. Sarraute, O. Buffet, and J. Hoffmann, "POMDPs make better hackers: Accounting for uncertainty in penetration testing," in *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[13] R. Tipireddy, S. Chatterjee, P. Paulson, M. Oster, and M. Halappanavar, "Agent-centric approach for cybersecurity decision-support with partial observability," in *2017 IEEE International Symposium on Technologies for Homeland Security (HST)*, April 2017, pp. 1–6.

[14] S. Musman, L. Booker, A. Applebaum, and B. Edmonds, "Steps toward a principled approach to automating cyber responses," in *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, T. Pham, Ed., vol. 11006, International Society for Optics and Photonics. SPIE, 2019, pp. 490 – 504.

[15] S. A. Zonouz, H. Khurana, W. H. Sanders, and T. M. Yardley, "RRE: A game-theoretic intrusion response and recovery engine," *IEEE Trans. on Parallel and Distributed Syst.*, vol. 25, no. 2, pp. 395–406, Feb 2014.

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

[17] S. Abraham and S. Nair, "Cyber security analytics: a stochastic model for security quantification using absorbing Markov chains," *Journal of Communications*, vol. 9, no. 12, pp. 899–907, 2014.

[18] C. Wang and V. M. Bier, "Quantifying adversary capabilities to inform defensive resource allocation," *Risk Analysis*, vol. 36, no. 4, pp. 756–775, 2016.

[19] W. J. Connell, "A quantitative framework for cyber moving target defenses," Ph.D. dissertation, George Mason University, 2017. [Online]. Available: http://ebot.gmu.edu/handle/1920/11325

[20] H. Debar, M. Dacier, and A. Wespi, "Towards a taxonomy of intrusion-detection systems," *Computer Networks*, vol. 31, no. 8, pp. 805 – 822, 1999.

[21] M. L. Puterman, *Markov decision processes*. New York: John Wiley and Sons, Inc., 1994.

[22] J. G. Kemeny and J. L. Snell, *Finite Markov chains*. New York: Van Nostrand, 1960.

[23] S. Panjwani, S. Tan, K. M. Jarrin, and M. Cukier, "An experimental evaluation to determine if port scans are precursors to an attack," in *2005 International Conference on Dependable Syst. and Networks (DSN'05)*, June 2005, pp. 602–611.

[24] M. J. Cros, "Markov decision processes toolbox," MATLAB Central File Exchange. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/25786-markov-decision-processes-mdp-toolbox

[25] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.

[26] A. Somani, N. Ye, D. Hsu, and W. Lee, "DESPOT: Online POMDP planning with regularization," in *Advances in neural information processing systems*, 2013, pp. 1772–1780.

[27] N. Ye, A. Somani, D. Hsu, and W. Lee, "Approximate POMDP planning online toolkit," National University of Singapore. [Online]. Available: https://github.com/AdaCompNUS/despot