Physical Security and Vulnerability Modeling for Infrastructure Facilities

Dean A. Jones
Chad E. Davis
Sandia National Laboratories
Albuquerque, NM 87185

dajones@sandia.gov
cedavis@sandia.gov

Mark A. Turnquist
Linda K. Nozick
Cornell University
Ithaca, NY 14853

mat14@cornell.edu
lkn3@cornell.edu

Abstract

A model of malicious intrusions in infrastructure facilities is developed, using a network representation of the system structure together with Markov models of intruder progress and strategy. This structure provides an explicit mechanism to estimate the probability of successful breaches of physical security, and to evaluate potential improvements. An example of an intruder attempting to place an explosive device on an airplane at an airport gate illustrates the structure and potential application of the model.

1. Introduction

There is widespread interest in protection of critical infrastructures from malicious attack. The attacks might be either physical intrusions (e.g., to steal vital material, plant a bomb, etc.) or cyber intrusions (e.g., to disrupt information systems, steal data, etc.). The attackers may be international terrorists, home-grown hackers, or ordinary criminals. In 1997, the report of the U.S. President's Commission on Critical Infrastructure Protection identified eight critical infrastructures "whose incapacity or destruction would have a debilitating impact on our defense and economic security" [11]. In subsequent years, this list of critical infrastructures was expanded and a set of 13 critical infrastructure sectors are included in the National Strategy for Homeland Security [3]. These 13 are: agriculture, food processing, water, public health, government, emergency services, banking and finance, telecommunications, energy, transportation, the chemical industry, postal and shipping services, and the defense industrial base.

In this analysis, we focus primarily on transportation facilities, but the approach we suggest could also be used in other infrastructure contexts. For example, a similar type of analysis has been applied to information systems [2]. The objective of the analysis presented here is to provide guidance to system owners and operators regarding effective ways to reduce vulnerabilities of specific facilities. To accomplish this, we develop a Markov Decision Process (MDP) model of how an intruder might try to penetrate the various barriers designed to protect the facility. This intruder model provides the basis for consideration of possible strategies to reduce the probability of a successful attack on the facility.

We represent the system of interest as a network of nodes and arcs. Nodes represent barriers that an intruder must penetrate, and arcs represent movements between barriers that an intruder can make within the system. The adversaries first must penetrate entry points to the system, and if an attempted penetration at a particular entry node is successful, they can traverse edges from the successfully breached node to other nodes in the network that are connected to the one breached. Traversing an edge entails a risk of detection. The adversary is assumed to make the decision that maximizes the probability of successful attack.

Several previous authors have used graph-based methods to represent attackers or defenders in security analyses. Phillips and Swiler [10] introduced the concept of an "attack graph" to represent sets of system states and paths for an attacker to pursue an objective in disrupting an information system. Several subsequent papers (e.g., [4], [13], [15]) have extended these initial ideas.

A number of authors have used Markov models

0-7695-2507-5/06/$20.00 (C) 2006 IEEE

to represent uncertainties in system state in the face of attacks, especially in computer systems (e.g., [4], [7], [13], [14]). In particular, Hidden Markov Models (HMM) focus on intruder detection using indicators that indirectly reflect potential attacker activities (see, for example, [8], [14], [16]).

Jha et al. [4] introduced the idea of using Markov Decision Processes (MDP) for situations in which the intruder's path is probabilistic. By interpreting attack graphs as Markov Decision Processes they computed a probability of intruder success for each attack represented by the graph. In the current work, we also use the idea of computing the probability of a successful attack by characterizing the problem as an MDP. However, our graph structure is different from the normal attack graph structure used in information systems, and thus the underlying network over which the MDP is formulated is different from that used in [4].

Our primary attention is on a class of adversaries that is rational and well informed. By "rational," we mean that the adversaries follow a strategy that maximizes the probability of their attack being successful. By "well informed," we mean that the adversaries know the probabilities of detection, success, etc. at various stages of the attack, so they can effectively optimize their attacks.

Our focus on well-informed adversaries is useful because it leads to an estimate of the probability of successful intrusion that is likely to be an upper bound on the actual value. This, in turn, leads us to be conservative in estimating how well-protected the system is. Less well-informed intruders might also be successful, but their probabilities of success will be smaller. Further exploration of the likely strategies of less informed intruders is, however, an important area for additional work.

We first construct an HMM to represent an intruder's actions at a single node (barrier) in a system. Then we develop an aggregated representation of that single-node model for inclusion in an MDP model of intruder strategy within a network representation of the entire system.

2. Intrusion attempts at a node

An attempt to penetrate a system barrier (node) and the interaction between the intruder and the intrusion-detection system is modeled using a Hidden Markov Model (HMM). The general concept of such a model is represented in Figure 1. The intruder's actions (the lower portion of the diagram) are assumed to progress through a set of states as a Markov process. The diagram in Figure 1 shows a simplified representation in which transitions are only to sequential states, but the transition matrix used can be more general. Occupancy of various states may result in emanations that are observable by the system operator (represented by the "signals" in Figure 1). For example, the intruder may be attempting to pick the lock of a door where there is video surveillance. Picking the lock requires an uncertain amount of time, represented by transition through a series of Markov states. While the intruder occupies those states (i.e., during the time that the intruder is attempting to pick the lock), there is a probability that his/her presence will be detected by the video surveillance system. The general structure of the HMM allows considerable flexibility in defining various types of signals and resulting actions by the system operator. For example, some signals may cause an increased level of surveillance without an alarm being raised. For our current purposes, we use a straightforward definition that a recognized signal from any state constitutes detection and the end of the attempted intrusion. If the intruder reaches a "breach" state without being detected, we say that the node (barrier) has been breached, and no further emanations will cause the system to detect the intruder at that node. We also include a "retreat" state that corresponds to an unsuccessful, but undetected, attempt to penetrate the barrier. In that outcome, the intruder can withdraw without raising an alarm.

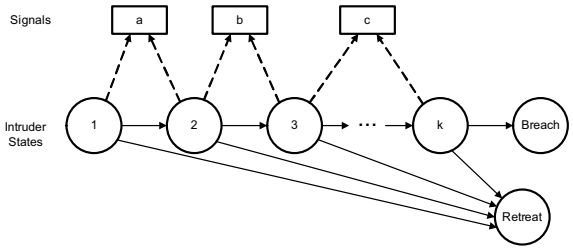

Figure 1. A hidden Markov model characterizing an attack at a system node.

We use a discrete-time, discrete-state HMM characterized by the following equations:

$$X_{n+1} = A^T X_n \tag{1}$$

$$Y_n = BX_n \tag{2}$$

for transition steps n = 1, 2, …, ∞. The state of the system (i.e., presence of the intruder in some node in

the lower portion of Figure 1) is represented by the (column) probability vector, *X*. The dynamics of the system are governed by (1), where *A* is a transition matrix (i.e., it satisfies the properties $a_{ij} \geq 0$ and $\sum_j a_{ij} = 1$.) The states of the system are not observed directly. The process *Y* is observed, which is a function of the state of the underlying Markov process, *X*. Each column of *B* specifies a conditional probability distribution over the possible observations, given that the underlying (hidden) system is in a particular state. The estimated values for B in a given application should reflect any efforts that might be taken by an intruder to reduce the likelihood of detection (e.g., attempting to defeat sensors, create diversions, etc.).

For our purposes, we assume that *A* and *B* are known (or have been estimated). We want to use the estimated HMMs at various nodes as the basis for a network-level model of intruder strategy. In large networks, it is useful to abstract the HMM at node *v* to a simpler representation, as shown in Figure 2. An intruder enters an "Attempt" state for that barrier (node). The intruder continues to occupy that state until the attempted penetration is detected (and an alarm is raised), the penetration is successful and the barrier is breached, or the intruder retreats.

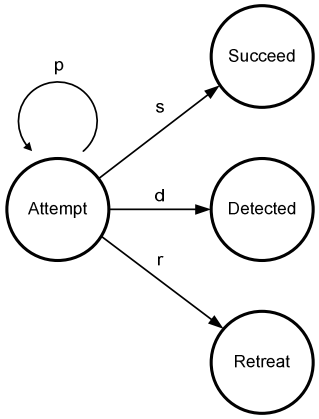

Figure 2. Aggregated abstraction of the HMM at a node.

To make the abstraction in Figure 2 useful, we must be able to derive the transition probabilities *p*, *s*, *d* and *r* from the underlying *A* and *B* matrices of the HMM. The transition probabilities *s, d* and *r* are specified so that the probabilities of detection, successful breach and retreat match those from the original HMM. The transition probability *p* is specified so the expected length of residence in the "attempt" state matches the duration of the attempted penetration in the original HMM. In the interests of space, the details are not given here, but they are provided in [5].

The value of the aggregated representation is that it allows us to construct a Markov Decision Process (MDP) of the intruder's strategy at the system level, without carrying along all the detail of states within each node. This is the focus of the following section.

3. Expanding to the system level

At the system level, we represent a network of barriers and potential movements as shown in the simple example in Figure 3. Each node can be expanded using a representation like the one in Figure 2.

If the intruder is successful at breaching a particular barrier, he/she has choices about where to go next (which arc to cross). Crossing arc *ij* entails a probability of detection δ_{ij}, and this is represented in the transition matrix.

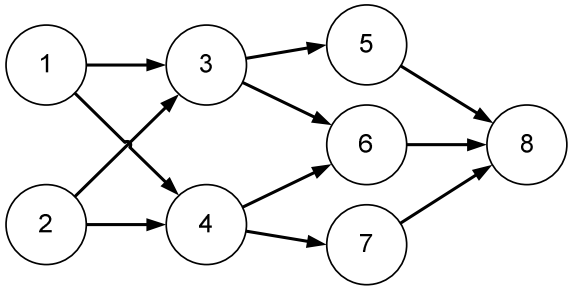

Figure 3. Simple system-level network.

We can pose the problem of finding the intruder's optimal strategy as an MDP over an infinite horizon. We define the expected "reward" to the intruder as a value associated with reaching the "success" state of a goal node (such as node 8 in the example in Figure 3), which represents an undetected exit from the system after accomplishing a desired action (such as placing a bomb, etc.). If we define this reward value as 1, then the expected rewards calculated at all earlier nodes in the network can be interpreted as probabilities of success, given that the intruder has reached that node.

We assume that the objective of the intruder is to maximize his/her expected reward (probability of successful attack), and we examine the problem of finding the optimal strategy for this objective. Solving this problem positions us to adopt the perspective of the system operator and consider the actions that can have the largest impact on reducing the probability of successful intrusions.

If the intruder is in state i and chooses action a_i, we denote the expected value of the future stream of rewards by $w(i,a_i)$. Each possible action a_i implies a change in the transition probabilities that govern the process. We denote the elements of the transition matrix resulting from choosing action a_i as $P_{ij}(a_i)$. The MDP we define for this problem is positive bounded, and we can find the optimal policy through either policy iteration or linear programming.

From a computational standpoint, policy iteration is generally preferable to linear programming for finding solutions, but the linear programming formulation can yield insights that are significant for our current purposes. Puterman [12] describes the linear programming formulation for positive bounded expected total reward models. The formulation seeks the decision policy (choice of a_i) that maximizes the expected value of the reward stream, $w(i,a_i)$. We denote the resulting optimal expected value as $w^*(i)$.

As [12] describes in detail, the set of $w^*(i)$ is the smallest set of values of $w(i)$ for which the following inequalities hold for all states, i:

$$w(i) \geq R_i(a_i) + \sum_j P_{ij}(a_i) w(j) \qquad (3)$$

where $R_i(a_i)$ is the immediate reward for selecting action a_i when the system state is i. In our application, $R_i(a_i) = 0$ for all states i other than the goal state, g, and $R_g(a_g) = 1$ for the dummy action, a_g, after achieving the goal state.

If we then introduce an arbitrary set of positive scalars, β_i, with the requirement that $\sum_i \beta_i = 1$, the linear program can be written as follows:

$$\min \sum_i \beta_i w(i) \qquad (4)$$

subject to:

$$w(i) - \sum_j P_{ij}(a_i) w(j) \geq R_i(a_i) \qquad \forall\, i, a_i \qquad (5)$$

$$w(i) \geq 0 \qquad \forall\, i \qquad (6)$$

This linear program has a dual that can be expressed as follows:

$$\max \sum_i \sum_{a_i} R_i(a_i) x_i(a_i) \qquad (7)$$

subject to:

$$\sum_{a_i} x_i(a_i) - \sum_j \sum_{a_i} P_{ij}(a_i) x_i(a_i) \leq \beta_i \qquad \forall i \qquad (8)$$

$$x_i(a_i) \geq 0 \quad \forall i, a_i \qquad (9)$$

In our case, because all but one of the $R_i(a_i)$ values are zero, the dual objective function can be simplified to:

$$\max \quad x_g(a_g) \qquad (7')$$

The primal linear program has many more constraints than variables, so it is more effective to solve the dual problem. In addition, it can be shown (see [12]) that in an optimal solution to the dual problem (7)–(9), there is no more than one non-zero $x_i(a_i)$ for each state i. The a_i for which $x_i(a_i)$ is non-zero indicates the optimal action a_i^* for each i. The shadow prices on the dual constraints (8) are the values of $w^*(i)$, indicating the probability of successful attack, given that the intruder has reached state i.

4. An illustrative application

As an example of system-level analysis for a specific infrastructure facility, consider an intruder who is attempting to place an explosive device aboard an aircraft while it is sitting at an airport gate, with the intent that it will explode later after the aircraft is in flight. A simplified representation of the barrier network and possible intruder actions is shown in Figure 4 (the network structure is the same as in Figure 3, but the nodes and links have now been labeled as specific barriers and movements).

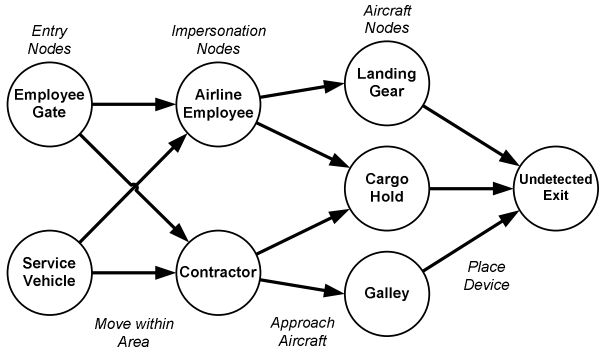

Figure 4. Illustrative network for analyzing an attempted placement of an explosive device on an aircraft.

The intruder must first gain access to the apron area of the terminal. We postulate that this can occur either by gaining illicit access through the employee gate (e.g., by stealing an employee ID and using it to enter the area), or by entering in a service vehicle at a gate (e.g., in a catering truck). If the intruder is successful in getting access to the area, he/she must then impersonate a legitimate worker in the aircraft gate area – either an airline employee or a service contractor. The "cross-over" arcs between "entry" and "impersonation" in Figure 4 indicate that even if the intruder gains access to the apron area using an employee ID, he/she may switch ID's and impersonate a service contractor within the area (or vice versa). This impersonation must be successful for the period of time required to get from the entrance to the aircraft itself.

Approaching the aircraft carries a risk of detection, and the approachable areas on the aircraft if the intruder is impersonating an employee may be different from those that are approachable if he/she is impersonating a service contractor. For example, a person who appears to be an airline maintenance employee might not attract attention approaching the under-wing area around the landing gear, whereas a person who appears to be a catering contractor would. For purposes of this example, we consider in Figure 4 three areas of the aircraft where an explosive device might be hidden – inside the wing around the landing gear, in the cargo hold, or in the catering supplies delivered to the galley.

If access to the aircraft is gained, the device must be placed without arousing suspicion. This is represented by the arcs connecting the aircraft area nodes to the exit node. Each of these arcs has a probability of detection.

Finally, if the intruder succeeds in gaining access to the aircraft and placing the device, he/she must exit without detection, and this represents the last barrier. Our modeling premise is that if the intruder is detected after placing the device, it will trigger a thorough search of the aircraft and the device will be discovered, so that the attempted attack will be foiled.

Table 1 summarizes the node data used for the example analysis, and Table 2 shows the probabilities of detection used for the arcs in the example network. These data are all inputs to the analysis and the values shown in Tables 1 and 2 are strictly hypothetical. In practice, these input values would likely be a mixture of estimates based on testing specific elements of the system and subjective estimates (i.e., expert judgment).

Table 1. Example data for network nodes.

Node (see Figure 4)	Expected Time for Attempted Breach (min)	Prob. of Success	Prob. of Detection	Prob. of Retreat
Employee Gate	1	0.6	0.25	0.15
Service Vehicle	2	0.4	0.4	0.2
Impersonate Employee	10	0.5	0.3	0.2
Impersonate Contractor	15	0.7	0.2	0.1
Landing Gear	5	0.6	0.3	0.1
Cargo Hold	3	0.7	0.25	0.05
Galley	15	0.55	0.4	0.05
Undetected Exit	10	0.8	0.2	0

Table 2. Probability of detection for possible moves.

Arc	Prob. of Detection
Empl. Gate – Impersonate Employee	0
Empl. Gate – Impersonate Contractor	0
Service Vehicle – Impersonate Empl.	0
Service Vehicle – Impersonate Contr.	0
Impersonate Empl. – Landing Gear	0.3
Impersonate Empl. – Cargo Hold	0.2
Impersonate Contr. – Cargo Hold	0.5
Impersonate Contr. – Galley	0.1
Landing Gear – Exit	0.4
Cargo Hold – Exit	0.2
Galley – Exit	0.3

In the example data, we assume there is no retreat at the stage of exiting after placing the device – at that stage either the attack is successful or it is detected. Also note that the probability of detection on the arcs leading to the "impersonation" nodes is zero. This is because we are treating impersonation process (and time) as a barrier (node), so the probability of detection is lumped at the nodes, rather than on the arcs.

For this set of input data, the solution for the optimal intruder strategy can be summarized as shown in Figure 5. To the left of each node is the probability of successful attack, given that the intruder is "arriving at" that barrier. To the right of each node is the probability of success, given that the intruder has successfully negotiated that barrier. There is only one value shown for the exit node (i.e., the "approaching" probability), because once that node is successfully negotiated, the attack has been a success, by definition.

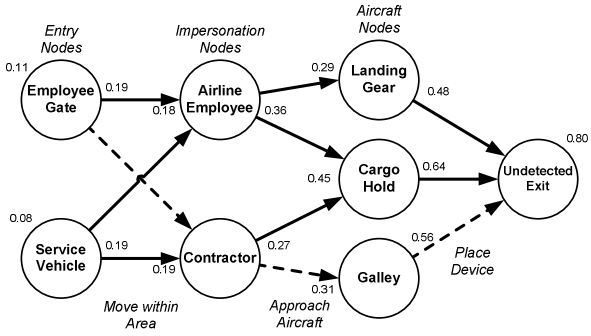

Figure 5. Summary of intruder strategy and probability of success.

The dashed line indicates the optimal path for an intruder (i.e., the path that maximizes the probability of success). This is the path of greatest vulnerability to the system. In our simple example, we would compute a probability of successful attack of 0.11 for an intruder whose strategy is to gain entry to the apron area through the employee gate, then switch ID's and impersonate a contractor (probably a catering service worker) to access the aircraft galley and place the device there before exiting.

The existence of this strategy does not mean that all intruders will always proceed in exactly the way indicated. It does mean that if all intruders were rational and well informed (in the sense described at the beginning of the paper), this would be a strategy through which they could maximize the probability of a successful attack. The actual probability of successful attack is likely to be less than this maximum value because intruders will have less-than-complete information and may not optimize their strategy. The solution to the MDP model also provides useful information on the conditional probability of success for an attacker that reaches a certain point in the network, regardless of whether or not he/she followed the optimal strategy. For example, if an intruder succeeds in reaching the cargo hold of the aircraft (despite the fact that this is not an optimal strategy), the probability of a successful attack from that point on is 0.45.

This information can be extended to represent a "vulnerability tree" as shown in Figure 6. This tree indicates the optimal strategy for continuing an attack by an intruder who reaches a given node, regardless of how he/she arrived there. This information adds value to system security studies over and above the identification of the single most vulnerable path for a system intruder.

Having established a base-case vulnerability assessment for the system, we can proceed to a series of "what if" analyses to examine the impact of potential changes to improve security. For example, what if an attempt were made to reduce the likelihood of successful attack along the most vulnerable path by more carefully checking contractors moving in the aircraft gate area and delivering food to the galley? We will represent this change in operational policy by increasing the probability of detection of someone impersonating a contractor moving in the gate area to 0.5 (and correspondingly decreasing the probability of successful impersonation to 0.4). We will represent the effect of increasing the vigilance on contractors entering the galley area of the aircraft by increasing the probability of detection on that access arc to 0.3.

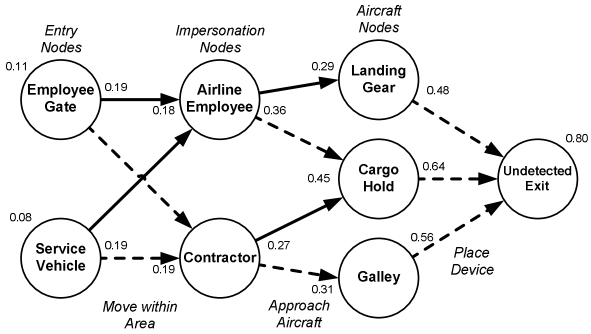

Figure 6. Vulnerability tree.

Figure 7 summarizes the results of those changes. The well-informed intruder adapts by changing

his/her strategy, and now impersonates an airline employee, making an attempt to place the explosive device in the cargo hold of the aircraft rather than in the galley. The overall probability of success has declined, but only marginally, to 0.105. Of course, the change might have somewhat greater short-term effectiveness (i.e., before the potential intruder can learn of it and change strategy), but it is unlikely to produce very significant improvements in security over a longer period.

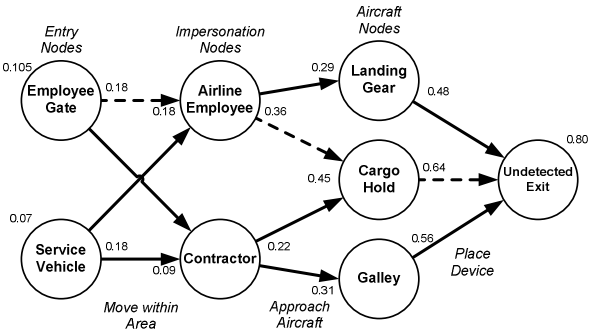

Figure 7. Revised intruder strategy after increases in monitoring levels for contractors.

One strategy for achieving greater long-term improvement in security is to focus on cut sets in the intrusion network. This idea is illustrated in Figure 8, which shows a cut set constructed across the arcs representing access to the aircraft. If simultaneous improvements in detection rates for intruders are made in all arcs of the cut set, it is more difficult for the intruder to change strategy to avoid the higher-security paths because all paths must cross the cut set.

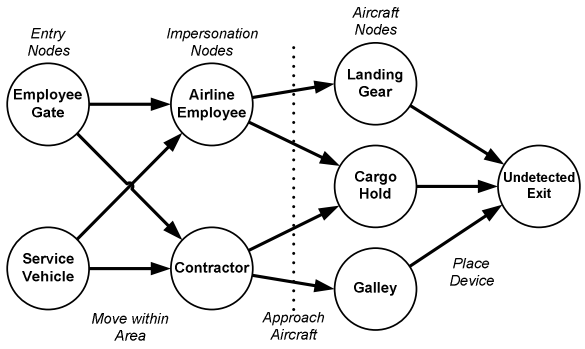

Figure 8. Illustration of cut set.

As an example, suppose that instead of focusing just on contractors, as in our first experiment, the probability of detection were increased to 0.6 on all arcs in the cut set shown in Figure 8. The resulting solution for intruder strategy is shown in Figure 9. The optimal intruder strategy has shifted from the galley to the cargo hold in response to this change, and the overall probability of successful attack has decreased to 0.075, a 32% decrease from the original value of 0.11.

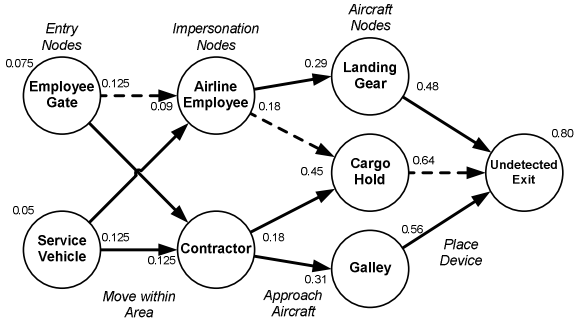

Figure 9. Intruder strategy and probability of success after increasing detection probability on cut set arcs to 0.6.

The model structure developed here can also be used to answer a variety of other questions. For example, suppose we were to focus our attention on the cut set in Figure 8. We have seen that an increase in the detection probability on those arcs to 0.6 results in a noticeable reduction in overall success probability for the intruder. How high would the detection probability on those cut set arcs have to be in order to reduce the overall intrusion success probability to 0.01? We can determine that the required detection probability is 0.97.

We can also use the model to examine combinations of strategies. For example, suppose we thought it would be feasible to increase the detection rate on the aircraft access arcs to 0.9, but not to 0.97. If 0.9 were achieved on those arcs, how much better would the detection probability have to be at the impersonation nodes preceding those arcs in order to achieve an overall success probability of no more than 0.01? We can do a quick search with the model and determine that the answer to this question is 0.68. That is, we would have to be able to maintain a 68% chance of detection of impersonators (of both employees and contractors), along with a 90% chance of detection of intruders approaching an aircraft, in order to reduce the probability of a successful attack to 0.01.

5. Optimizing resource allocation for security improvement

The illustrative analysis in Section 4 leads us to an obvious question: If it were possible to estimate a cost function for changes within the network that would reduce the likelihood of a successful intrusion, could we identify the most effective (i.e., minimum cost) way of achieving a desired (small) probability of successful intrusion? This question can be answered using a bi-level optimization formulation. At the "upper level" we have an optimization that determines changes at nodes and arcs in the network so as to minimize cost, subject to a constraint that the resulting probability of successful attack is no greater than a specified value. However, the probability of successful attack is determined as the solution to a "lower level" optimization (optimizing the intruder's strategy, given the characteristics of the network he/she is facing).

To be more specific about this optimization, consider again the model of the intruder's strategy expressed in equations (4)-(6). There are at least five ways that the system operator (or "defender") can act to reduce the likelihood that the intruder will be successful:

- Increase the probability of detection at barrier (node) *i* ; this might be accomplished either by increasing the sensitivity of the detection process, or by increasing the time required to penetrate the barrier, allowing the existing detection mechanisms more time to be effective.
- Increase the probability of detection on movement arcs *ij* between nodes.
- Add new barriers that must be negotiated; this is represented by a new node in the network, with reconnection of existing arcs to force some (or all) intruders' paths to go through the new node.
- Remove existing arcs in the network; this represents some additional constraints (either physical or virtual) on movement within the system.
- Reduce the level of information that potential intruders have about the system structure and detection probabilities, creating additional uncertainty for the intruders, and perhaps some level of "disinformation" that would lead them to make poor choices in their attack strategy.

From the standpoint of the model we have defined, the third and fourth strategies listed can be considered to be special (extreme) cases of the first two strategies (for more detailed discussion of this, see [5]). The fifth strategy is quite different from the first two, and needs to be analyzed in a separate way. This is described further in the following section as an extension of the work in the current paper.

For our current analysis, we will focus on the first two strategies for reducing the vulnerability of the system (implicitly including the third and fourth as well). Suppose that the initial detection probability at node *i* is denoted d_i^0, and the increase in that probability is denoted Δ_i, so that the actual detection probability in effect is $d_i = d_i^0 + \Delta_i$.

Similarly, we will assume that the initial detection probability on arc *ij* is δ_{ij}^0, and the increase in that probability is γ_{ij}, so the actual detection probability in effect is $\delta_{ij} = \delta_{ij}^0 + \gamma_{ij}$.

Increases in the detection probabilities are assumed to require expenditures $C_i(\Delta_i)$ and $K_{ij}(\gamma_{ij})$. In the current formulation, the cost functions are separable by node and arc, but a more general cost function could be used without changing the structure of the bi-level optimization formulation.

We will use *E* to denote the set of entry nodes to the system network, and then express the "upper level" problem as follows:

$$\text{Min} \sum_i C_i(\Delta_i) + \sum_{ij} K_{ij}(\gamma_{ij}) \qquad (10)$$

subject to:

$$w^*(i) \le W^* \qquad \forall\, i \in E \qquad (11)$$

$$d_i = d_i^0 + \Delta_i \qquad \forall\, i \qquad (12)$$

$$\delta_{ij} = \delta_{ij}^0 + \gamma_{ij} \qquad \forall\, ij \qquad (13)$$

$$\Delta_i \ge 0 \qquad \forall\, i \qquad (14)$$

$$\gamma_{ij} \ge 0 \qquad \forall\, ij \qquad (15)$$

In (11), the $w^*(i)$ values are the optimal solution to the "lower level" problem, specified as follows:

$$\min \sum_i \beta_i w(i) \tag{16}$$

subject to:

$$w(i) - \sum_j P_{ij}(a_i \mid d_i, \delta_{ij}) w(j) \geq 0 \qquad \forall\, i \neq g, a_i \tag{17}$$

$$w(g) - \sum_j P_{gj}(a_g \mid d_g, \delta_{gj}) w(j) \geq 1 \qquad \forall\ a_g \tag{18}$$

$$w(i) \geq 0 \qquad \forall\, i \tag{19}$$

In (17) and (18), the transition matrix is written as $P_{ij}(a_i \mid d_i, \delta_{ij})$ to reflect the fact that it depends on the values of d_i and δ_{ij} determined in the upper problem. The lower problem in (16)-(19) is the same problem as in (4)-(6), but is re-written to reflect the specific knowledge of $R_i(a_i)$ values that relevant to this problem, and to emphasize its connection to the upper problem in (10)-(15).

A solution procedure for this bi-level optimization searches over possible values of Δ_i and γ_{ij}, and for each set of values, solves the lower problem to find $w^*(i)$ (after translating the d_i and δ_{ij} values into a new transition matrix $P_{ij}(a_i \mid d_i, \delta_{ij})$). A general issue (which is endemic to bi-level models) is that it is difficult to guarantee convergence of solution algorithms to true optimal solutions in the upper model. Bard [1] describes this general difficulty.

6. Extensions

Several extensions to the model described here are possible and desirable. In addition to further development of the bi-level optimization ideas discussed in the previous section, there are two extensions that seem particularly important. First, it is useful to incorporate imperfect information on the part of the intruders. This allows us to begin exploration of the fifth "defender" strategy mentioned in section 5. One very direct way to do this is to embed the MDP model in a simulation where uncertainty in the perceptions of the detection probabilities is reflected. This is one type of limitation on the information assumed to be available to the attackers. Variations in the perceptions of the detection probabilities can lead to different strategies for different intruders, and the effect (from the system operator's perspective) is that potential attacks appear to be following a mixed (or randomized) strategy. This form of simulation is a step in the general direction of considering the system to be a partially observable Markov decision process (POMDP) from the perspective of the intruder. The simulation approach can also be used to analyze other types of imperfect information on the part of intruders – for example, imperfect knowledge of what arcs exist in the network for movement among nodes, or even imperfect information as to what nodes exist.

A second useful extension is to create semi-Markov models for the processes of attempted penetration of barriers. This would allow more accurate representation of the uncertain time required to penetrate a given barrier, as well as offer a broader range of opportunities for modeling various types of time-dependent detection probabilities. This extension could improve the range of applicability of the model.

7. Conclusions

The objective of the analysis presented here is to provide guidance to system owners and operators regarding effective ways to reduce vulnerabilities of specific infrastructure facilities. To accomplish this, we have developed a Markov Decision Process (MDP) model of how an intruder might try to penetrate the various barriers designed to protect the facility. The solution to this MDP model provides insight into the level of vulnerability of the facility (the probability of successful intrusion) and indicates where the vulnerabilities are (the most likely paths for the intruder).

The intruder model also provides the basis for consideration of possible strategies to reduce the probability of a successful attack on the facility. Illustrations of using the model in this way are provided in the case study analysis in section 4. The process of searching for cost-effective strategies to reduce system vulnerability can be formally cast as a bi-level optimization problem, as discussed in section 5. This provides a promising direction for further work.

Successful implementation of the model described in this paper depends very directly on two important tasks: 1) constructing large-scale networks that represent the various barriers and movement possibilities in a system; and 2) estimating the various probabilities embedded in the *A* and *B* matrices that are elements of the HMM's at each network node. Quite clearly, if the constructed network does not reflect accurately the barriers to intrusion and possible

paths for intruders, the resulting computations from the model will be flawed. Constructing an accurate network representation requires significant system knowledge and also the ability to "think like an attacker." Estimating the probabilities is also a challenging task. There are tools that have been created for estimating HMM matrices in other application contexts, and the experience gained in those other contexts should provide important insight for this task.

The process of testing, implementing and enhancing the model is an ongoing one, with the expectation that this approach will become an important new tool for the protection of critical infrastructure facilities.

References

[1] Bard, J.F., "Some Properties of the Bilevel Programming Problem," *Journal of Optimization Theory and Applications*, 68:2, 1991, 371-378.

[2] Carlson, R.E., Turnquist, M.A. and Nozick, L.K., *Expected Losses, Insurability and Benefits from Reducing Vulnerability to Attacks*, Report SAND2004-0742, Sandia National Laboratories, Albuquerque, NM, 2004.

[3] Executive Office of the President, *National Strategy for Homeland Security*, July 2002, available on line at http://www.dhs.gov.

[4] Jha, S., Sheyner, O., and Wing, J.M. "Two Formal Analyses of Attack Graphs," *15th IEEE Computer Security Foundations Workshop*, June 2002, Cape Breton, NS, Canada, 49-63.

[5] Jones, D.A., Turnquist, M.A. and Nozick, L.K., *Physical Security and Vulnerability Modeling for Infrastructure Facilities*, Report SAND2005-xxxx,Sandia National Laboratories, Albuquerque, NM, 2005.

[6] Katsikas, S.K., Gritzalis, D., and Spirakis, P., "Attack Modelling in Open Network Environments," *Communications and Multimedia Security II*, 1996, 268-277.

[7] Katsikas, S.K., Spyrou, T., Gritzalis, D., and Darzentas, J., "Model for Network Behaviour under Viral Attack," *Computer Communications,* 19:2, 1996, 124-132.

[8] Ourston, D., Matzner, S., Stump, W., and Hopkins, B., "Applications of Hidden Markov Models to Detecting Multi-stage Network Attacks," *36th Hawaii International Conference on Systems Science*, IEEE Computer Society, Hawaii, 2003, CD-ROM, 10p.

[9] Ourston, D., Matzner, S., Stump, W., and Hopkins, B. "Coordinated Internet Attacks: Responding to Attack Complexity," *Journal of Computer Security*, 12:2, 2004, 165-190.

[10] Phillips, C.A., and Swiler, L.P., "A Graph-Based System for Network Vulnerability Analysis," *Proceedings of the 1998 New Security Paradigms Workshop*, Association for Computing Machinery, 1998, 71-81.

[11] President's Commission on Critical Infrastructure Protection, *Critical Foundations: Protecting America's Infrastructures*, The White House, Washington, DC, 1997.

[12] Puterman, M.L. *Markov Decision Processes*. Wiley, New York, 1994.

[13] Sheyner, O., Haines, J., Jha, S., Lippmann, R., and Wing, J.M., "Automated Generation and Analysis of Attack Graphs," *Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy*, Berkeley, CA, May 2002, 273-284.

[14] Soh, B.C., and Dillon, T.S. "Setting Optimal Intrusion-Detection Thresholds," *Computers & Security*, 14:7, 1995, 621-631.

[15] Swiler, L.P., Phillips, C.A., Ellis, D., and Chakerian, S., "Computer Attack Graph Generation Tool," *Proceedings of the 2nd DARPA Information Survivability Conference and Exposition*, 2001, 307-321.

[16] Warrender, C., Forrest, S. and Pearlmutter, B. "Detecting Intrusions Using System Calls: Alternative Data Models," *Proceedings of the 1999 IEEE Symposium on Security and Privacy*, 1999, 133-145.