DETERMINING COST-EFFECTIVE INTRUSION DETECTION APPROACHES FOR AN ADVANCED METERING INFRASTRUCTURE DEPLOYMENT USING ADVISE

BY

MICHAEL J. RAUSCH

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2016

Urbana, Illinois

Adviser:

Professor William H. Sanders

ABSTRACT

Utilities responsible for Advanced Metering Infrastructure (AMI) networks must be able to defend themselves from a variety of potential attacks so they may achieve the goals of delivering power to consumers and maintaining the integrity of their equipment and data. Intrusion detection systems (IDSes) can play an important part in the defense of such networks. Utilities should carefully consider the strengths and weaknesses of different IDS deployment strategies to choose the most cost-effective solution. Models of adversary behavior in the presence of different IDS deployments can help with making this decision as we demonstrate through a case study that uses a model created in the ADversary VIew Security Evaluation (ADVISE) formalism (which calculates metrics used to compare different IDSes). We show how these metrics give valuable insight into the selection of the appropriate IDS architecture for an AMI network.

To my family, for their love and support.

ACKNOWLEDGMENTS

I would like to thank my adviser, Professor William H. Sanders, for his insight and support of this project, and for giving me the opportunity to do research in the PERFORM group.

I would like to thank Ken Keefe for helping me to learn the Möbius framework and the ADVISE modeling formalism, and for assisting me in writing an early draft of this thesis. I would also like to thank Dr. Brett Feddersen for the engaging discussions, encouragement and technical advice. I also greatly appreciate the editorial assistance given by Jenny Applequist.

I am grateful for the advice and friendship of the members of the PERFORM group: Varun Badrinath Krishna, Atul Bohara, Carmen Cheh, Ahmed Fawaz, Mohammad Noureddine, Uttam Thakore, Benjamin Ujcich and Ronald Wright. I have learned a great deal from your example.

I am especially grateful for my loving family. Thank you for always being there for me and teaching me the most important things in life.

This thesis contains previously published material by M. Rausch, B. Feddersen, K. Keefe, and W.H. Sanders [1]. The material from [1] was published in the *Proceedings of the 13th International Conference on Quantitative Evaluation of Systems (QEST 2016)*. The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-319-43425-4_19

The work described here was performed, in part, with funding from the Department of Homeland Security under contract HSHQDC-13-C-B0014, "Practical Metrics for Enterprise Security Engineering."

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Motivation

Many utility companies are creating Advanced Metering Infrastructure (AMI) networks, which incorporate smart meters and other intelligent components into the power grid. The added functionality allows utilities to monitor and control their smart grid more effectively. The increased monitoring and control helps the utility respond to fluctuations in demand for electricity, makes some classes of energy theft attacks more difficult to attempt, and enables some new demand response schemes, to name a few benefits. Some estimate that the smart grid will provide billions of dollars per year in added benefit compared to the traditional electrical grid [2], while the United States International Trade Commission cites market observers who predict the global smart meter market will rise from $4 billion in 2011 to about $20 billion in 2018 [3].

Unfortunately, AMI networks increase the attack surface of a power grid. For example, an unscrupulous customer may compromise a single smart meter so that it sends false data to under-report electricity consumption, resulting in a lower bill. These attacks have already occurred in various locations around the world, [4] [5], and prevention methods have generated interest in the academic community [6] [7] [8].

Energy theft is not the only threat: utilities are also concerned about attacks that would affect the integrity or availability of the grid. Distributed denial of service attacks, traffic injection attacks, and Byzantine attacks are examples of new threats to these cyber-enhanced power grids [9]. As utility companies build and maintain AMI infrastructures they should be aware of the possibility of these attacks, and work to create a cost-efficient architecture to minimize the expected damage. There could be catastrophic financial consequences if these systems are not protected. Even relatively simple, low-tech attacks can be devastating, as evidenced by the Metcalf sniper attack of 2013, which caused over $15 million in damage [10]. According to a report by the insurance company Lloyd's, a cyber-attack on the United States grid could cost tens of billions of dollars in damage [11].

An effective approach to limiting the potential damage of an attack is to detect and

respond to the attack before it can cause much harm. An intrusion detection system (IDS) can help a utility company detect an attack on their cyber-enhanced networks. There are several different IDS architectures that can be deployed by a utility company as a defensive precaution [12]. Each architecture has a different cost and degree of effectiveness. A utility company must decide whether its application warrants an IDS, and if so, which would give the best protection for the best price. Making the correct decision is vital.

One common way to approach this problem is to hire security experts and have them collaborate with the system engineers to develop an appropriate security architecture. This is a good start, but the approach suffers from three limitations:

- First, assumptions of the various collaborating experts are often not made explicit. The experts may have different backgrounds, different ways of approaching the problem, and different implicit assumptions. It is necessary to have some mechanism or common language that is intuitive and easy to use to help make these assumptions explicit to facilitate communicate between the domain experts and the security experts.

- Second, decisions made using this approach are not easily auditable. If the decisions are not auditable, it will be impossible for an independent outside expert to validate whether they are correct. Even if the security expert charged with conducting the initial risk assessment is infallible, it would be useful to have a record of the expert's decision making process when he or she is no longer employed, so others may make an equally good decision when conditions in the system change.

- Finally, this approach relies heavily on the intuition and experience of the experts involved in the process. Architects in other domains (e.g. civil engineering projects) do not have to rely solely on intuition and experience. Instead, their intuition and experience is verified and complemented by science-based mathematical models. We should similarly work towards a science of security [13].

One approach for informing this critical design decision is to build a sound, state-based stochastic model of the system and the possible IDS architectures that can be applied to it. Quantitative metrics can be calculated on the models to determine which configuration provides the best cost/security balance. This thesis presents one such approach in the form a model created using the ADversary VIew Security Evaluation (ADVISE) formalism [14].

1.2 Related Work

The academic community, recognizing the importance of the topic, has done prior work comparing different security approaches in AMI, many examples of which can be found in the survey by Wang and Lu [15]. In this section we shall highlight some of the most relevant and interesting results.

The analysis in [16] proposes a method to determine the utility of adding expensive radio-frequency identification (RFID) technology to help discourage electricity theft. The authors construct a cost model which incorporates the number of new components required for the RFID scheme, and the price of each component, including installation and shipping costs. The authors also construct a benefit model which incorporates the difference in electricity stolen with and without the RFID technology. The cost model and the benefit model are then combined to create a cost-benefit model which allows a utility to calculate whether it makes financial sense to add the RFID technology. The paper does not consider integrity or availability attacks, but instead limits its scope to energy theft. In this thesis we also compare costs and benefits, but we use a stochastic modeling technique, and consider integrity and availability attacks.

The techniques proposed in [17] seek to compare and evaluate the security of different AMI IDSes. The authors use two special types of attack trees to guide penetration testing. An *archetypal attack tree* is generic and vendor-independent, while a *concrete attack tree* is vendor-specific. The authors realized that attack trees for similar systems composed of components from different vendors had many similarities, but also important differences. The use of archetypal and concrete attack trees allows one archetypal attack tree describing the commonalities to be created and reused for several systems, and the concrete attack tree could be grafted to the archetypal attack tree to specify vendor-specific differences. Once the modeler creates these attack trees they may be used to enable penetration testing to evaluate proposed systems. The authors focus on energy fraud, denial of service, and targeted disconnect attacks. These attack trees could help a modeler create an attack execution graph in an ADVISE model. In contrast to this thesis, their paper does not explicitly model the attacker's attributes or motivations in detail.

The authors of [12] provide a framework for evaluating the cost-effectiveness of different IDS architectures in an AMI network. Their framework incorporates the installation and maintenance costs of the intrusion detection system, the effectiveness of the intrusion detection system, how densely the smart meters are deployed, and several other factors to develop a model. However, the approach outlined in the paper does not explicitly account for the differences in the behaviors of adversaries when attacking the system, and considers

attacks at a very high level. In this thesis, we build upon the framework, but extend it by incorporating the adversary behavior into the model, along with a lower-level view of the attacks. We believe that the resulting model is more realistic and provides results that are more useful to those who must make security decisions.

1.3 Contributions and Organization

This thesis presents a modeling approach a utility may utilize to make sound, auditable, and informed design decisions to help secure AMI deployments with a cost-effective intrusion detection system. The metrics generated by our executed models provide rich insight into the system, adversaries that may confront the system, and the interactions between the system and the adversaries. We use an extensive case study to demonstrate the effectiveness of the approach.

We offer a multi-layered power grid example and the potential IDS implementations that can be applied to this grid. We consider three different IDS deployment strategies — centralized, dedicated, and embedded — to determine the most cost effective approach the utility may take. We then review the ADversary VIew Security Evaluation (ADVISE) [14] formalism, and introduce a new adversary decision algorithm, which has been implemented in the Möbius modeling tool [18].

We demonstrate how the formalism may be used to construct a model of attacks that may be attempted against the power grid we previously defined. We use the same formalism to construct models of several different types of adversary — insider, malicious customer, nation state, and terrorist — that may be a threat to the utility company. We then show how the attack models and the adversary models may be composed to explore interactions between the system and the adversaries.

We show how to define a variety of metrics that may be calculated using the ADVISE model we construct. Using these metrics, a utility company can make a more informed decision about how to implement an IDS on its grid. Given a particular intrusion detection deployment approach and adversary, the metrics generated by the modeling approach may be used to estimate the probability of detecting the adversary, the sequence of attack steps taken by the adversary to reach a goal, the cost to the adversary to achieve their goal, and, perhaps most importantly, the damage done to the utility due to the activity of the adversary.

However, the most important contribution is the end-to-end demonstration of a modeling method for comparing the relative security-related strengths and weaknesses of an AMI

deployment. A model that gives quantitative security metrics will give a system designer another technique to complement the advice and intuition of security experts, and moves us closer to the goal of developing a science of security.

The remainder of this thesis is organized as follows. Chapter 2 provides background on AMI networks and intrusion detection systems, and presents the example power grid that we use in our case study. Chapter 3 gives a review of the ADVISE formalism, and describes the various submodels which compose an ADVISE model, including the attack model, adversary models, and metric models. It also describes how an ADVISE model is executed to calculate the metrics of interest. Chapter 4 offers a detailed explanation of the ADVISE model that was constructed, including the adversary profiles that were considered and the metrics that were defined on the system. Chapter 5 shows our quantitative results and our interpretation of them. It also presents a thorough sensitivity analysis, to explore to what degree the model results change with different input parameters. Finally, Chapter 6 provides future directions and concludes the work.

CHAPTER 2

SYSTEM DESCRIPTION

2.1 AMI Overview

A utility company may benefit from an AMI deployment in a number of different ways. An AMI deployment enables remote meter readings, and some deployments allow the utility to remotely connect and disconnect a consumer's smart meter. Both of these features may reduce the need for expensive truck rolls which had previously been necessary to accomplish these tasks [19]. An AMI deployment enables more frequent meter readings, which may help an AMI learn to anticipate changes in consumer demand for electricity and to increase or decrease generation to properly meet electricity demand. Additionally, an AMI deployment may make it easier to detect certain classes of electricity theft attacks [20].

An AMI deployment also enables some forms of *demand response* that otherwise would have been difficult or impossible to implement [21]. The rate of electricity consumption is not constant. Occasionally there will be periods of relatively high demand. For example, there is likely to be high demand at 1 pm on a hot summer day when many consumers will be running air conditioning units at the same time. A utility that does not respond appropriately to periods of increased demand elevates the risk of brownouts or even blackouts. Traditionally, utilities could only effectively respond to increased demand by increasing supply by bringing additional generators online, which is expensive. However, demand response enabled by AMI deployments may, in real-time or close to real-time, increase the cost of electricity during times of high demand in an attempt to suppress some of the demand. In this way, demand response complements supply response, in theory limiting the number of generators that need to be brought online to meet the increased demand and saving the utility money.

Advanced Metering Infrastructure deployments offer many clear benefits to utility companies by reducing operational costs and giving the utilities greater control of their networks, which has led many utilities to upgrade their existing grids or deploy new smart grids. There are two main deployment strategies to enable connectivity between the smart meter and the utility's command and control center: a utility may either choose to use a commercial Internet connection, or build and use their own network. For example, a utility may install a

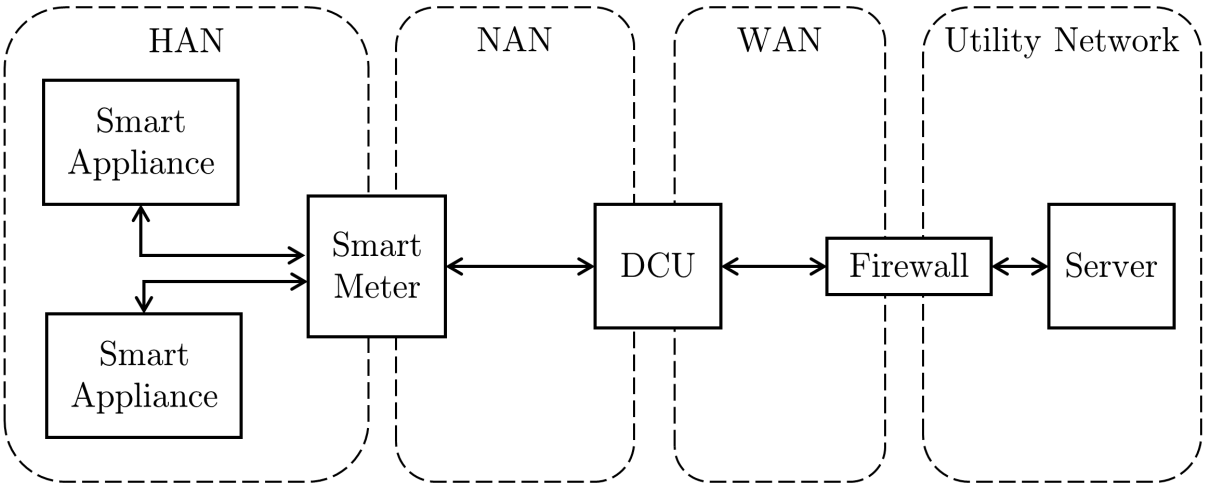

Figure 2.1: Advanced Metering Infrastructure system diagram.

smart meter in the consumer's home and communicate with it through the consumer's Internet connection, or a commercial cellular network or commercial cable network. However, some utilities prefer to use their own networks to establish the connectivity between the smart meter and the utility. The utility may choose to use their own networks so they have greater control over the network (because they may have heightened security, availability, or integrity requirements), or because the area the consumer lives in may not provide any convenient preexisting networks that a utility can leverage.

In our case study we consider a hypothetical utility which uses its own network infrastructure. We choose to only study deployments utilizing the utility company's own network, because we focus on AMI network defense from the perspective of the utility company. If the utility does not own and operate the network they are not likely to have much influence on how it is defended, and the problem becomes one of defending a general-purpose connectivity infrastructure, which is outside the scope of this thesis.

The network architecture of the AMI considered in this case study is shown in Figure 2.1. This network architecture and its variants are commonly used, and is similar to the system described in [12] and [22].

At the bottom level of the hierarchy, a smart meter forms the core of a *home area network* (HAN). The HAN may include other smart appliances, in addition to the smart meter. An example may be a charging station for an electric car which is programmed to respond to commands from the smart meter to recharge the car during times of low demand, and to cease charging during times of high demand [23]. If there are multiple devices in the HAN, the smart meter itself may act as a network gateway for the other devices.

Multiple HANs may be connected with each other and one or more data concentrator

units (DCUs) to form a *neighborhood area network* (NAN). A DCU serves as a gateway for the NAN, collecting readings from a number of meters and forwarding it to the utility, or relaying commands from the utility control center to the smart meters. The NANs do not require high bandwidth or long range, and can use shorter-range wireless. We assume in our case study that the NAN uses a wireless mesh network.

The *wide area network* (WAN) keeps the various NANs connected to the utility's control center, enabling the communication between the DCUs and the utility. The WAN usually utilizes higher bandwidth, long-range communication technologies such as long-range wireless, satellite, or power line communication. We make no assumptions about what communication technology the utility uses in the WAN.

We limit the scope of the case study to attacks on the electric distribution system, and do not consider attacks on the generation or transmission of electricity.

2.2 IDS Overview

Intrusion detection systems are intended to monitor a system for suspicious activity, to raise an alert if a security event occurs, and to log information to determine how an attack proceeded through the system. A number of different IDS deployment strategies are possible in an AMI. In this thesis, we consider centralized IDSes, dedicated distributed IDSes, and embedded distributed IDSes. These options are shown in Figure 2.2.

A *centralized* IDS deployment scheme would place an IDS at the top of the network hierarchy, in the utility company's network. The IDS would monitor all traffic flowing into and out of the utility company's command and control center LAN network, and raise an alert if it detected anything suspicious. However, it would be completely unaware of inter-meter traffic, since that would not pass through the top level of the hierarchy. This is a significant limitation. As a result, the IDS may miss evidence of an adversary's presence spreading through the smart meters in a NAN until it is too late to prevent an attack.

Alternatively, a utility could deploy a distributed set of IDSes to monitor inter-meter communication. This approach would still require a central node to coordinate the monitoring, so it would have many of the benefits of a centralized IDS approach, along with the added benefit of inter-meter traffic monitoring. Unfortunately, this approach would incur increased installation and maintenance costs associated with the additional IDSes. We consider two main distributed IDS variants: dedicated and embedded.

A *dedicated* IDS deployment would have the same components as a centralized deployment, and in addition it would have a number of geographically distributed dedicated IDS devices

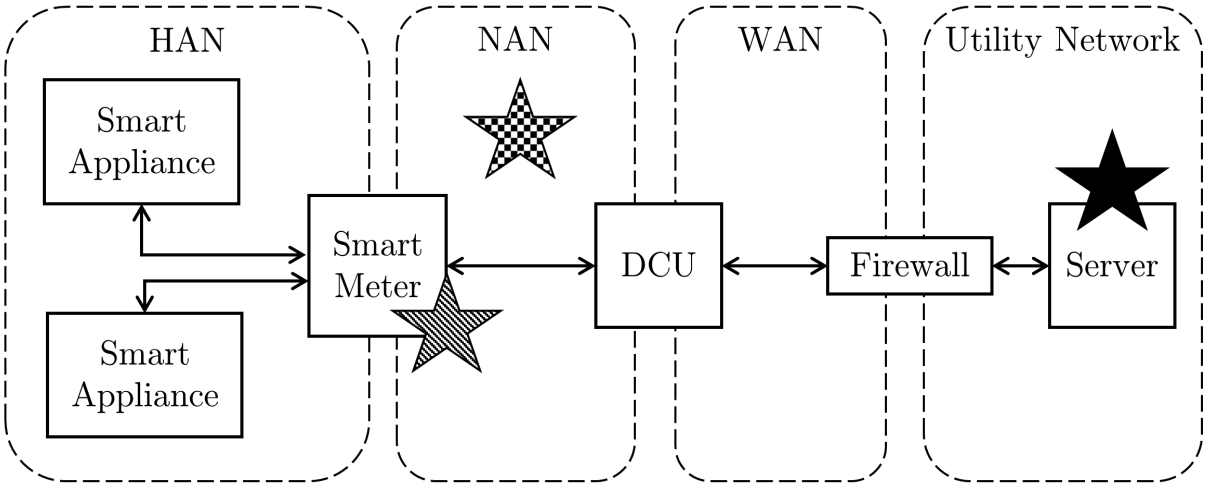

Figure 2.2: An AMI network with different intrusion detection system deployment options displayed. The location of the IDSes are denoted by stars. The solid black star denotes the location of the centralized IDS, while the location of the dedicated and embedded IDSes are denoted by checkered and striped stars, respectively.

in wireless communication with the smart meters. These IDSes would monitor AMI traffic within wireless range in an effort to detect malicious and anomalous network behavior.

An *embedded* IDS deployment incorporates intrusion detection directly into the smart meter. Like the dedicated IDS architecture, the embedded IDSes communicate and cooperate with the central IDS device.

There are a number of trade-offs to consider when evaluating these IDS designs. A centralized IDS would potentially miss large families of attacks because it is unaware of inter-meter communication. It is, however, the cheapest IDS option. An embedded or dedicated IDS scheme would be able to observe inter-meter communication, possibly allowing it to detect a larger set of attacks than a centralized scheme, but would cost more.

A dedicated architecture would cost more because many additional devices would have to be purchased and maintained, and installation would require separate permits and location sites. Technicians would require additional training to install and maintain the devices. However, the device would be able to monitor inter-meter communication in the NAN. One dedicated device could serve multiple smart meters, which could be especially advantageous in an urban setting with a dense deployment of smart meters.

An embedded system would not require separate building sites or permits, but every single meter would cost slightly more because of the added IDS capability. Given the large number of meters involved, even a small increase in price for an individual meter would potentially be very costly for a utility company. Smart meters may need to obey regulations about how

much power they can consume, and smart meters often have limited computing capabilities, which restrains the power of the IDS that may be installed. However, no new installation sites or permits would be required, and technicians would only have to be trained to install and maintain one type of device, in contrast with the distributed architecture. In addition to monitoring inter-meter communication, an embedded IDS architecture would be able to detect attacks on the meter itself, which is a capability the other IDSes we consider may not have. This means the embedded IDS option provides the greatest coverage against possible attacks among the IDSes we consider.

We do not claim that these are the only IDS deployment strategies available, and a utility could choose to use some combination of them. However, we have chosen to do an in-depth analysis of these three options, rather than a shallower analysis of more options. A utility interested in other IDS deployment strategies could evaluate them easily by extending the ADVISE models developed in Chapter 4.

CHAPTER 3

ADVISE OVERVIEW

The ADversary VIew Security Evaluation (ADVISE) method [14] is used to calculate quantitative security metrics via executable models of adversary behavior in a system [24]. At a high level, a modeler first creates an Attack Execution Graph (AEG), and defines an adversary profile. The AEG is similar to a standard attack tree [25], however, it incorporates additional details about each attack's properties (e.g. cost, time to completion, probability of success, etc.), and contains nodes that track the state of the model. The adversary profile expresses an adversary's initial capabilities to access different subsystems, knowledge relevant to accomplishing an attack, or skill in various techniques. It also defines the adversary's motivations, and to what degree the adversary values various goals. The AEG and adversary profile are then combined together with a model of the metrics of interest. This combined model is then executed to calculate the metrics. These metrics may then be used by a modeler to help make security-relevant design decisions. In our case, the metrics will help a security analyst determine the most cost-effective intrusion detection system architecture for an Advanced Metering Infrastructure deployment.

3.1 Attack Execution Graphs

An *AEG*, originally described in [14], is defined by the tuple

$$< A, R, K, S, G, C >$$

where A is the set of attack steps, R is the set of access domains available to the adversary, K is the set of information that can be known by the adversary, S is the set of skills possessed by the adversary, and G is the set of goals that the adversary attempts to achieve. The relation C defines the set of directed connecting arcs from $e \in R \cup K \cup S \cup G$ to $a \in A$, where e is a prerequisite element needed in order to attempt a. This relation also defines the set of directed connecting arcs from $a \in A$ to $e \in R \cup K \cup S \cup G$, where e is an affected element that may be changed by the performance of a. The elements R, K, S, and G are state variables

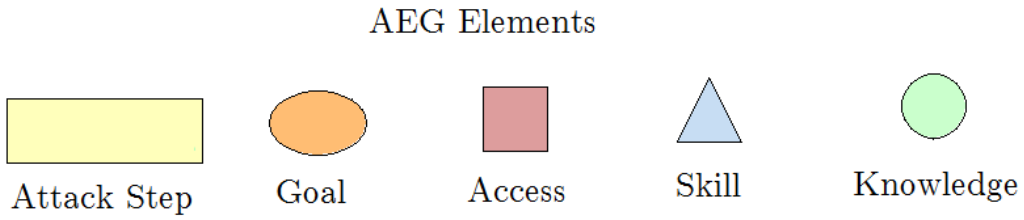

Figure 3.1: Graphical representation of elements in an Attack Execution Graph.

that hold an integer value that usually represents whether the element represented by the state variable has been obtained (1 or 0).

An *attack step*, originally described in [14], is defined by the tuple

$$< B, T, C, O >$$

where B is a Boolean precondition that indicates whether or not the attack step is currently enabled, T is the timing distribution that is sampled to determine the time it takes to complete the attack step, C is the cost to the adversary for attempting the attack, and O is the set of outcomes of the attack (such as success or failure). Each outcome contains a Pr, D, and E, which are the probability the outcome will be selected from an attack step's O, the probability of being detected for that outcome, and the effect of that outcome on the state of the model, respectively. An adversary uses the solution of a competitive Markov decision process [26] as described in [14] to select the best attack step given the adversary's characteristics, abilities and preferences.

In Möbius AEGs are represented graphically, as seen in Figure 3.1, according to the following convention: attack steps are represented by yellow rectangles, goals state variables are represented by orange ovals, access state variables are represented by red squares, skill state variables are represented by blue triangles, and knowledge state variables are represented by green circles.

3.2 Adversary Profile

The adversary profile defines the initial starting position and attributes of the adversary, how the adversary makes decisions, and what motivates the adversary to act. Specifically, there are three components to the adversary profile, (1) the description of the initial state of

the model, which defines the attacker's initial foothold in the system, the skills the attacker possesses, and the knowledge with which the attacker starts, (2) the adversary decision algorithm, which the attacker uses to select the next attack step to attempt, and (3) the goal assignment function, which assigns the payoff the attacker receives for holding goals.

Adversaries may have different characteristics, which may lead to different behavior. To express the differences in characteristics, the ADVISE formalism requires an adversary profile, in addition to the Attack Execution Graph described previously.

An *adversary profile*, described in [14], is formally defined by the tuple

$$< s_0, V, U_C, U_P, U_D, N >$$

where s_0 is the initial state of the model, V is the attack goal value function, U_C, U_P, and U_D are the preference weights for avoiding cost, obtaining payoff, and avoiding detection, respectively, and N is the planning horizon (the maximum number of steps into the future that may be considered) for the adversary.

The initial state of the model, s_0, defines the initial value of every access, skill, knowledge and goal state variable in the model. The attack goal value function, V, assigns a payoff for achieving a goal for every goal in the model. An adversary may not be equally interested in achieving every goal in the model, but rather prefer some goals to others. This idea may be precisely expressed in the adversary profile via the attack goal value function.

3.2.1 Adversary Decision Algorithm

According to LeMay [14], the adversary selects the best attack step given the state using a stationary deterministic Markovian decision rule [26],

$$\beta^N(s) = \arg\max_{a_i \in A_s} \{attr^N(a_i, s)\},$$

where s is the state, A_s is the set of actions that the adversary may attempt given state s, and N is the planning horizon. The attractiveness function *attr* returns a value representing the expected value of the attack a_i given s. Different definitions of *attr* are possible.

The case studies in [14] used the following definition for *attr*:

$$attr^N(a_i, s) = w_C \cdot U_C(C_i^N(s)) + w_P \cdot U_P(P_i^N(s)) + w_D \cdot U_D(D_i^N(s)).$$

The attack preference weights, w_C, w_P, and w_D, define the adversary's preference for avoiding cost, gaining payoff, and avoiding detection, respectively, and $w_C + w_P + w_D = 1$. The utility

functions for cost, payoff, and detection are $U_C(\cdot), U_P(\cdot)$, and $U_D(\cdot)$, respectively. Finally, C_i^N, P_i^N, and D_i^N are the recursively computed expected path cost, expected path payoff, and expected path detection, respectively.

LeMay defined a function to convert cost units, c, given on a [0-100] scale, to utility units on a [0-1] scale:

$$U_C(c) = \begin{cases} 1, & \text{when } c < 0, \\ \frac{e^2 - e^{(c/50)}}{e^2 - 1}, & \text{when } 0 \leq c \leq 100, \\ 0, & \text{when } c < 100. \end{cases}$$

Similarly, she constructed a function to convert payoff units, p, given on a [0-1000] scale, to utility units on a [0-1] scale:

$$U_P(p) = \begin{cases} 0, & \text{when } p < 0, \\ \frac{e^{10/3} - \frac{e^{10/3}}{e^{p/300}}}{e^{10/3} - 1}, & \text{when } 0 \leq p \leq 1000, \\ 1, & \text{when } p < 1000. \end{cases}$$

More simply, she also defined a function to convert detection probabilities, d, given on a [0-1] scale, to utility units on a [0-1] scale:

$$U_D(d) = \frac{1 - \frac{e^2}{e^{2d}}}{1 - e^2}.$$

We found it challenging to use these conversion functions. Ensuring that the costs and payoffs would not exceed the their respective ranges of [0-100] and [0-1000] was difficult, especially when long planning horizons were used. Furthermore, since our model defines costs and payoffs in terms of dollars, using this definition would have required us to define additional functions to convert cost in dollars to cost units in a [0-100] range, and payoff in dollars to payoff units in a [0-1000] range. The exponential nature of the functions also frequently led to unintuitive results and behavior.

For this case study, we used a simplified definition of the attractiveness function *attr*, based on the concept of expected net profit. We defined *attr* to be

$$attr^N(a_i, s) = -Cost(a_i) + \sum_{t \in Outcomes(a_i)} Pr(t)(Payoff(t) - PrDetect(t) * DetectionCost),$$

where $Cost(a)$ is a function that returns the expected cost the adversary incurs for attempting a, $Outcomes(a)$ is a function that returns the set of outcomes of a, $Pr(t)$ is a function that returns the probability that outcome t will be randomly selected if the adversary selects

a, $Payoff(t)$ returns the payoff for the adversary if the outcome occurs, $PrDetect(t)$ is a function that returns the probability of detection if the outcome occurs, and $DetectionCost$ is a constant that defines the cost the adversary incurs when detected by the defender.

This net-profit definition of the attractiveness function, which we use for all of the adversaries in this thesis, has a number of benefits. Since we are not converting cost and payoff values into utility values on a [0-1] range, we do not need to use the non-intuitive and complicated exponential functions. As a result, we do not need to set a lower or upper bound on the values the cost and payoff values may take, and it allows us to simply define one unit of payoff or one unit of cost to be equal to one dollar. These benefits greatly simplify the modeling process.

3.3 Reward Variable Model

Theoretically, ADVISE supports more than one kind of reward model for expressing metrics. However, in this case study, metrics are defined using rate- and event-based performance variables [27]. Possible metrics include:

- **Accumulated Adversary Payoff**
  The total payoff for the model being in its current state from the point of view of the adversary at a specific discrete time point.
- **Average Number of Attempts at Specific Attack Step**
  The mean number of times an attack step is attempted in a specific time interval.
- **Cost to Adversary**
  The total cost accumulated by the adversary for its behavior in the model.
- **Number of Goals Achieved**
  The number of goals held by an adversary at a specific discrete point of time.
- **Probability State Variable Held by Adversary**
  The probability a particular ADVISE state variable (access, skill, knowledge, or goal) is held by the adversary at a specific discrete time point.
- **Probability of Detecting Adversary**
  The probability that the adversary will be detected sometime in a specific time interval.
- **Specific Attack Step Outcome Rate**
  The number of times a particular outcome occurs during a specific interval of time.

- **Total Damage Caused by Adversary**
  The total damage from the perspective of the defender in the current state of the model at a discrete time point.

- **Value of Access, Knowledge, Skill or Goal State Variable**
  The number held by a particular ADVISE state variable at a specific discrete time point.

This is not an exhaustive list of possible metrics. Indeed, the performance variable model gives a modeler the power and flexibility to create custom metrics tailored to their specific use case. We describe the particular metrics we use in this case study in Chapter 4.

3.4 Solving the Models

The ADVISE model, once defined, may be executed to calculate the defined metrics. An executable ADVISE model consists of the initial model state and the functions that may change the model state. There are two primary classes of these functions: the adversary attack decision function (which we have described previously), and the attack outcome selection functions.

The adversary attack decision function takes the set of all attack steps and the current model state as input and returns an attack step, the attack step that the adversary will attempt next. The adversary attack decision function first determines all of the attacks for which the precondition expression is true, as these are the only attacks that an adversary may choose to attempt. Then the function calculates the attractiveness of each of these remaining attack steps, and selects the most attractive attack step given the model state. Once the adversary decides upon a particular attack step one of the outcomes of the attack step is randomly selected according to the probabilities associated with each outcome of that attack step, and the effect of the outcome is applied to the model state. Then the process starts over from the beginning. The most attractive attack for the adversary given the new model state is found, one of the outcomes of the attack is randomly selected, the model state is updated. This cycle repeats until the simulation ends. This hybrid of game theory and simulation is used to calculate the security metrics of interest, which may be used by security analysts to help make informed design decisions.

CHAPTER 4

DESCRIPTION OF MODEL

In this case study, we consider a hypothetical utility company with an urban deployment of an AMI network, as shown in Figure 2.1 and described in Chapter 2. We have based our system on the system described in [12], following it in detail whenever possible. In this network, zero or more smart appliances connect to a smart meter at each home and together form a HAN. Multiple HANs are connected to one another and one or more gateways via a wireless mesh network to form a NAN. Multiple NANs are connected to one another and to the utility command and control center network via a WAN.

The utility wishes to supply power to consumers, protect their equipment, ensure the integrity of communication in the AMI network, and ensure the confidentiality of communication in the AMI network.

The utility company in this scenario wishes to analyze four classes of potential adversaries: disgruntled insider employees who wish to cause as much monetary damage as possible in retribution for a perceived wrong, unscrupulous customers who wish to under-report their electricity consumption to unfairly lower their bill, sophisticated, well-funded nation-states who wish to interrupt the delivery of power and cause as much damage as possible, and terrorists who wish to gain notoriety for their cause by publicly compromising the availability and integrity of the power grid.

An adversary may choose from a variety of attacks to achieve a goal. We utilized the literature search conducted in [9] to compile a list of attacks for inclusion in our model. An adversary may install a node to jam the wireless communication in the NAN or perform various routing attacks. Alternatively, the adversary may choose to compromise one or more smart meters either by physical means (such as an optical port compromise) or through a remote compromise (possibly through a compromised device in the HAN that is also connected to the Internet) to gain control of the component(s). Once compromised, these device(s) can be used to launch Byzantine attacks, black-hole attacks, denial of service attacks, man-in-the-middle attacks, and other attacks. In addition, an attacker may choose to take physical rather than cyber action.

The utility company wishes to compare the cost-effectiveness of various proposed IDS

Label	Attack Step Name
1	Install Long Range Jammer
2	Install Short Range Jammer
3	Install Malicious Smart Meter
4	Physical Smart Meter Exploit
5	Mass Meter Compromise
6	Remote Smart Meter Exploit
7	Collect Crypto Keys
8	Create Botnet
9	Analyze Traffic
10	Gain Routing Capability
11	Major Jamming Attack
12	Minor Jamming Attack
13	Physical Attack
14	Minor Routing Attack
15	Major Routing Attack
16	Byzantine Attack
17	Resource Exhaustion Attack

Table 4.1: Attack step labels and their corresponding names from Figure 4.1.

architectures. In particular it wishes to compare the centralized IDS solution with the two distributed IDS solutions: embedded and dedicated. The utility can easily obtain the estimated installation and maintenance costs of an IDS from vendors. However, estimating the expected benefit of implementing the IDS is much more difficult. We attempt to make such an estimate with an ADVISE model.

We used the ADVISE formalism as implemented in Möbius to create a model that was detailed enough to calculate the quantitative security metrics of interest, while minimizing the number of assumptions that a more detailed model would have forced us to make. We were primarily interested in four key model results: the attack path used by the adversary, the average cost the adversary incurs while attacking the system, the estimated probability of detecting an adversary, and the estimated damage to the utility company due to adversary behavior.

4.1 Attack Execution Graph Model

The attack execution graph, which is shown in Figure 4.1, contains three main adversary goals; three auxiliary goals; seventeen attack steps that an adversary may attempt when

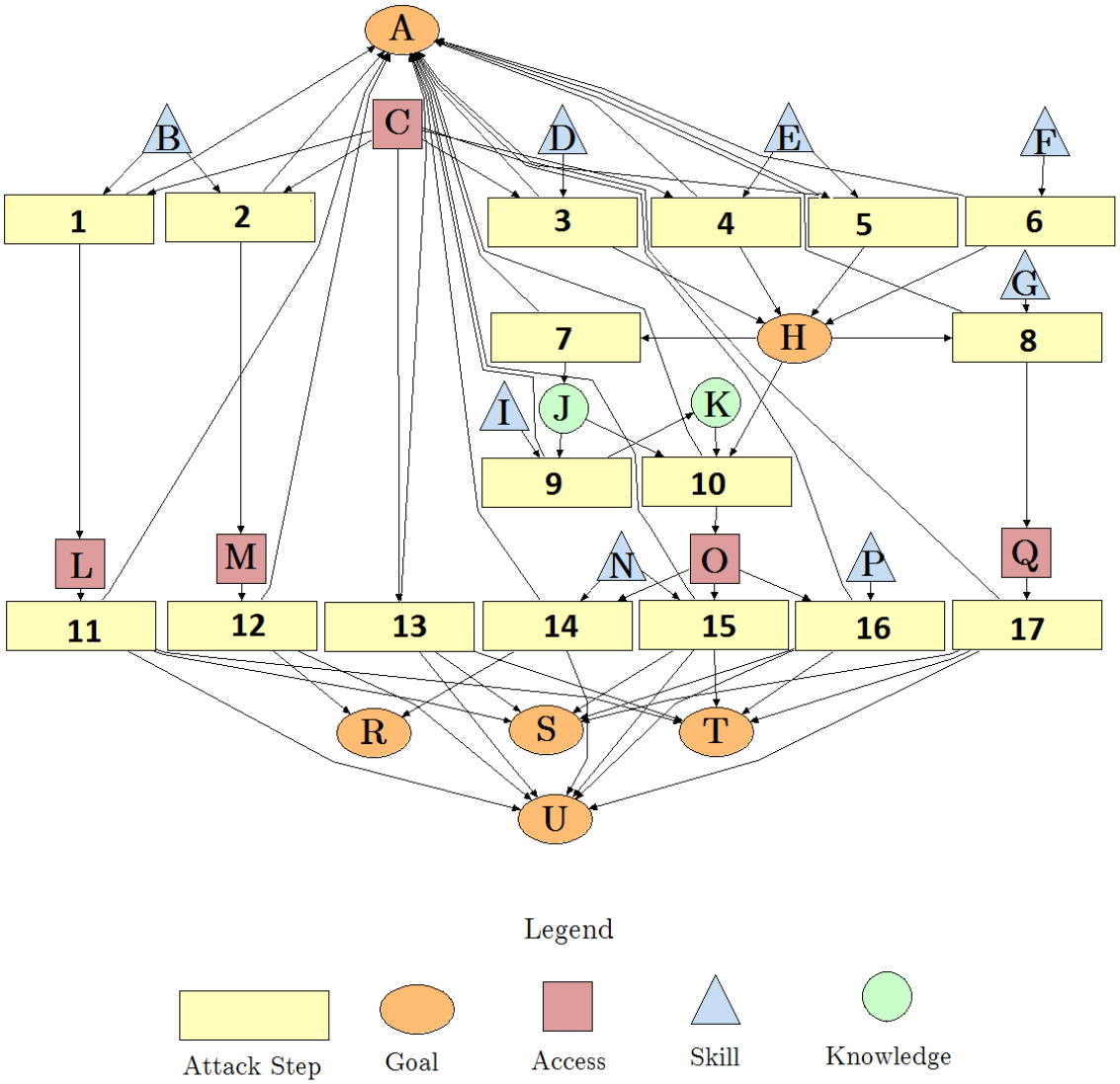

Figure 4.1: Attack Execution Graph of ADVISE Model. The names associated with each attack step label can be found in Table 4.1, and the names associated with each state variable can be found in Table 4.2.

Label	State Variable Name
A	Undetected Goal
B	Node Installation Skill
C	Physical Access
D	Smart Meter Installation Skill
E	Physical Smart Meter Exploit Skill
F	Remote Smart Meter Exploit Skill
G	Botnet Shepherd Skill
H	Compromised Smart Meters
I	Traffic Analysis Skill
J	Crypto Keys
K	Traffic Knowledge
L	Long Range Jammer Access
M	Short Range Jammer Access
N	Routing Attack Skill
O	Routing Capability
P	Byzantine Attack Skill
Q	Botnet Access
R	Steal Energy Goal
S	Interrupt Service Goal
T	Damage Equipment Goal
U	Goal Achieved

Table 4.2: State variable labels and their corresponding names from Figure 4.1.

trying to reach a goal; a number of supporting access, knowledge, and skill state variables that may help an adversary satisfy the preconditions for attempting a particular attack; and arcs that connect the attack steps to particular accesses, knowledge, skills, and goals and signify the relationships between them. The access, skill, knowledge, and goal state variables in our model hold a value of zero if they are not held by the adversary, and a positive integer otherwise.

The set of goals desired by the adversary drives his or her behavior and are therefore one of the most important components of the model. Cheating the company by under-reporting electricity consumption, interrupting the delivery of electric power, and damaging the utility's equipment are the three most important adversary goals. Those goals are represented in Figure 4.1 by Goals R, S, and T, respectively. In addition, the adversary wishes to remain undetected; this goal is represented by Goal A, the *Undetected* goal. Goal H is the supporting goal of acquiring compromised smart meters. Finally, Goal U represents the goal of achieving at least one of the three primary goals described previously. Usually goal state variables initially hold a value of zero; and the value is incremented on the successful conclusion of an attack. The notable exception is the *Undetected* goal, which initially holds a value of 1 and is decremented to 0 if an attack fails and the adversary is detected.

In the model, the adversary is not allowed to attack after being detected. For this reason, every attack step in the graph is connected by an arc to the *Undetected* goal, which is a prerequisite that must be satisfied before attempting any attack step. If the *Undetected* goal holds a value of zero, the adversary may not attempt any attack, with the exception of the unique *DoNothing* attack step [14].

Attack steps can be attempted by an adversary either to achieve a goal directly or to change the model state to make it easier to achieve a goal later. An attack step must have at least one outcome. In this particular ADVISE model every attack step outcome results either in the certain detection of an adversary, or the adversary remaining undetected. In other words, if p_o is the probability of detection associated with an outcome o, then $p_o = 0$ or $p_o = 1$, but $p_o \notin (0, 1)$.

Attack Step 1 in the AEG diagram is an *Install Long Range Jammer Attack*. This attack step requires the adversary to be undetected, to be in reasonably close proximity to the smart meters, and to have skill in installing wireless jammers. It would result in the adversary's having access to a long-range wireless jammer that incapacitates the wireless mesh network in the NAN. The adversary must hold several prerequisites to attempt this attack, including the *NodeInstallationSkill*, represented by Skill B, the *PhysicalSmartMeterAccess*, represented by Access C, and the *Undetected* goal. At the successful conclusion of the attack, the adversary gains the *LongRangeJammerAccess*, Access L, whose value is incremented from 0 to 1. Attack

Step 2, *Install Short Range Jammer*, is very similar, but its purpose is to gain access to a short-range wireless jammer that blocks communication in a HAN rather than a NAN, so Attack 2 is connected to *ShortRangeJammerAccess*, Access M. Attack 2's prerequisites are identical to Attack 1's prerequisites.

Any one of Attack Steps 3, 4, 5, and 6 may be attempted by an adversary in an effort to obtain the *NumCompromisedSmartMeters* goal (Goal H), which would give the adversary control of smart meters in the AMI network. Attack Step 3, *InstallMaliciousSmartMeter*, aims to accomplish this goal by installing a new meter (controlled by the adversary) that tricks the AMI network into accepting it as one of its own smart meters. Attack Step 3 requires physical access to the AMI network and skill in installation as prerequisites, and so is connected to the *PhysicalAccess* access and the *SmartMeterInstallationSkill* skill, shown as Access C and Skill D, respectively. Attack Step 4, *PhysicalSmartMeterExploit*, represents an adversary attempt to physically tamper with the smart meter to gain control of it. An adversary must have physical access to the smart meters and skill in this exploit to attempt the attack step, so Attack Step 4 is connected to Access C and Skill E, which are the *PhysicalAccess* access and the *PhysicalSmartMeterExploitSkill* skill, respectively. Attack Step 5, *MassMeterCompromise*, is very similar to Attack Step 4, with the major difference being that 50 smart meters are compromised if this attack step is achieved instead of just one. Finally, Attack Step 6 also compromises 50 smart meters, but it requires the adversary to have the appropriate skill (*RemoteSmartMeterExploitSkill*, shown as Skill F) and does not require the adversary to have physical access to the smart meters or other power grid components.

Attack Steps 7, 9, and 10 are related because their sequence leads to Access O, the *RoutingCapability* access, which is a prerequisite for Attack Steps 14, 15, and 16. Attack Step 7, *CollectCryptoKeys*, represents the adversary's attempt to collect cryptographic keys from the compromised smart meters. The adversary must have access to compromised smart meters to attempt the attack, and if the attack step is successful, it leads to the acquisition of knowledge of the cryptographic keys, which is represented by Knowledge Item J. The *AnalyzeTraffic* attack step (Attack Step 9) requires the adversary to hold Knowledge Item J and Skill I (I being the *TrafficAnalysis* skill) in order to attempt the attack step. If successful, the adversary gains sufficient knowledge of the traffic in the network to launch sophisticated routing and Byzantine attacks. This knowledge is represented by the *TrafficKnowledge* state variable, which is Knowledge Item K. Finally, the adversary may attempt Attack Step 10, *GainRoutingCapability*, if he or she has knowledge of the keys and traffic and at least one compromised smart meter. If the prerequisites have been satisfied, the adversary will successfully execute the attack step and gain the *RoutingCapability* access.

Attack Step 8, *CreateBotnet*, gives the adversary the *BotnetAccess* access, depicted as Access Q, which is a prerequisite for launching resource exhaustion attacks such as DDoS attacks. To attempt the attack step, the adversary must hold Skill G, the *BotnetShepherd* skill, as well as at least 50 smart meters, represented by a value greater than or equal to 50 in Goal H.

There are seven attack steps that directly achieve at least one of the three most significant goals (Attacks 11-17). First, Attack Step 11, the *Major Jamming Attack*, requires the adversary to have access to a long range jammer; it results in a significant interruption of service in the NAN, and also damages equipment, since important commands for coordinating the network are not delivered. Attack Step 12, which is the *Minor Jamming Attack*, requires access to a short-range jammer: it does not result in loss of power or damage to equipment but may be utilized to help an unscrupulous customer give a false power reading to under-report electricity consumption. Attack Step 13, *PhysicalAttack*, represents a major physical, non-cyber attack on the equipment of the utility company. This attack requires only physical access to the equipment, and causes a significant blackout and major damage to the equipment. It has a relatively high probability of detection, but requires only minimal prerequisites to attempt. Attack Steps 14 and 15, *MinorRoutingAttack* and *MajorRoutingAttack*, respectively, are similar in that they have the same prerequisites, *RoutingAttack* skill and *RoutingCapability* access (Skill N and Access O, respectively), but have different intended goals. The *MinorRoutingAttack* under-reports the electricity consumption of one customer. The *MajorRoutingAttack*, in contrast, leads to interrupted service and damage to the AMI network equipment. Attack Step 16, *ByzantineAttack*, requires that the adversary hold the *RoutingCapability* access and the *ByzantineAttack* skill (Access O and Skill P, respectively), and a successful outcome for the adversary leads to damaged equipment and interrupted service. Finally, Attack Step 17, the *ResourceExhaustion* attack, requires *BotnetAccess* and results in damaged equipment and interrupted service.

In addition, there is one attack step not shown in the diagram, the *DoNothing* attack step, which an adversary may attempt at any time and has no effect on the model state, costs nothing, and will never lead to the detection of the adversary. This attack step may be attempted by an adversary when the expected payoff for attempting any other attack step does not justify the risk of detection and the cost of attempting the attack step.

The same AEG is used to model every IDS approach studied: none, centralized, dedicated and embedded. The IDS used in a particular simulation run is modeled as a global variable [18]. The cost to attempt the attack step, the probability that an attack step will lead to a successful outcome for an adversary, as well as the effect an outcome will have on the system, may be adjusted based on the IDS approach being modeled. The following description of

the *CreateBotnet* attack will illustrate how this may be accomplished.

Each attack step contains detailed information about the probability of success, probability of detection, cost to attempt, effect on the system, duration, and other information. We present one attack step as an example, the *CreateBotnet* attack, which is Attack Step 8 in the diagram. It is defined to cost the adversary $1,000 to attempt if no IDS is present in the system, and $5,000 otherwise. If attempted, the attack step takes 8 hours to complete. If the attack is to be attempted, the *Undetected* goal state variable and the *BotnetShepherdSkill* skill state variable must both contain a positive value, the *NumCompromisedSmartMeters* goal state variable must hold a value greater than or equal to 50, and finally the adversary must not already possess the *Botnet* access. If these conditions are not met, the attack step cannot be attempted. If the attack step is attempted, one of three outcomes, $FailureUndetected$, $FailureDetected$, or $Success$, is randomly chosen according to their probabilities of occurrence. The $FailureUndetected$ outcome represents the event in which the adversary attempts the attack and fails, but remains undetected. It has no effect on the state of the model. The $FailureDetected$ outcome represents the event in which the adversary attempts the attack, fails, and is detected. If this outcome is randomly selected by the simulation, it modifies the model state by changing the value of the *Undetected* goal from 1 to 0, disabling any future attack, and depriving the adversary of any reward he or she may otherwise have received for remaining undetected. Finally, the $Success$ outcome represents the successful completion of the attack. It has the effect of giving the adversary access to a botnet of smart meters, which is represented by changing the value of the *BotnetAccess*, State Variable Q, from 0 to 1. The probabilities of randomly selecting the $FailureUndetected$, $FailureDetected$, and $Success$ outcomes varies based on the adversary's level of expertise in the *BotnetShepherdSkill* and the type of IDS in the system. The probability of selecting the $FailureDetected$ outcome is set to 0.05. The probability of selecting the $Success$ outcome is set to be $0.4 + 0.5 \cdot BotnetShepherdSkill$, so the adversary will be more likely to succeed if he or she is more skilled. The probability of the $FailureUndetected$ outcome is set to be $1 - prob(FailureDetected) - prob(Success)$. As a result, $prob(Success) + prob(FailureDetected) + prob(FailureUndetected) = 1$. However, if an embedded IDS is installed in the system, the attack is less likely to succeed. In that case, the probability of the $Success$ outcome being chosen is scaled down, and the probability of the $FailureDetected$ outcome is scaled up, in such a way that $prob(Success) + prob(FailureDetected) + prob(FailureUndetected)$ remains 1. All the other attack steps in the model have a similar level of detail.

Initial State	Customer	Insider	Terrorist	Nation-State
BotnetAccess		1		
LongRangeJammer				
PhysicalAccess	1	1	1	
RoutingCapability		1		
ShortRangeJammer				
CryptoKeys		1		
TrafficKnowledge		1		1
BotnetShepherdSkill		7	6	10
ByzantineAttackSkill	2	8	5	10
NodeInstallationSkill	8	10	9	10
PhysicalSmartMeterExploitSkill	7	9	7	10
RemoteSmartMeterExploitSkill		7	7	10
RoutingAttackSkill	5	10	5	10
SmartMeterInstallationSkill		10	7	9
TrafficAnalysisSkill	3	9	5	10
CompromisedMeters		51		

Table 4.3: Initial state values for each adversary. Skills are expressed in a range from 0 to 10.

4.2 Attacker Model

In addition to a model of attacks against the system, we need a model of the adversary, since different adversaries have different goals, and different strengths and weaknesses. These differences can lead to very different behaviors when the attackers are confronted by the same system. Table 4.3 shows the state variables initially held by each adversary, which describes their initial foothold in the system, their knowledge of system properties, and the skills that will help them accomplish their attacks.

In ADVISE models an adversary has a specified preference for avoiding cost, avoiding detection, and gaining payoff [14]. Different adversaries may weigh these preferences differently. For example, a well-funded nation-state may have substantial resources to launch an attack but also a great desire to avoid detection. On the other hand, a terrorist may be unable to afford an expensive attack but is very tolerant of the possibility of being detected and caught. In previous ADVISE models adversary preferences were expressed by varying the weights on the attack decision parameters [14]. For this case study, all adversaries weigh cost and payoff equally, since we use the net-profit attractiveness function, as explained in

	Customer	Insider	Nation-State	Terrorist
Undetected Goal	$2000	$80000	$10000000	$0
CompromisedSmartMeters	$0	$0	$0	$0
Steal Energy Goal	$2000	$0	$0	$0
Interrupt Service Goal	$0	$100	$10000	$10
Damage Equipment Goal	$0	$1000	$20000	$100
Goal Achieved	$0	$0	$0	$0

Table 4.4: The per-unit payoff assigned to each goal.

Chapter 3.

The adversary's desire to avoid detection is expressed in the attack execution graph as a goal, the *Undetected* goal. Every adversary initially holds this goal and if the adversary is detected the adversary will not gain the payoff associated with this goal. If one adversary, (call her Alice) has a higher payoff associated with the *Undetected* goal than another adversary (call her Eve), it means that Alice has a higher preference for remaining undetected than Eve. Unlike previous ADVISE models (e.g. the case studies in [14]), we choose to model the preference to remain undetected as a goal in the AEG, in an effort to make the adversary's behavior easily understandable.

Table 4.4 gives the per-unit payoff assigned to each goal. The total payoff the adversary receives at the end of the attack is the value of the goal times the per-unit payoff. For example, if the malicious insider has a per-unit payoff of $100 for the *Damage Equipment* goal, and the integer value of the goal at the end of the simulation is 7, the adversary receives $\$100 \cdot 7 = \$700$ of total payoff. For this reason, the adversaries are motivated to achieve as high a value as possible for each goal to maximize payoff.

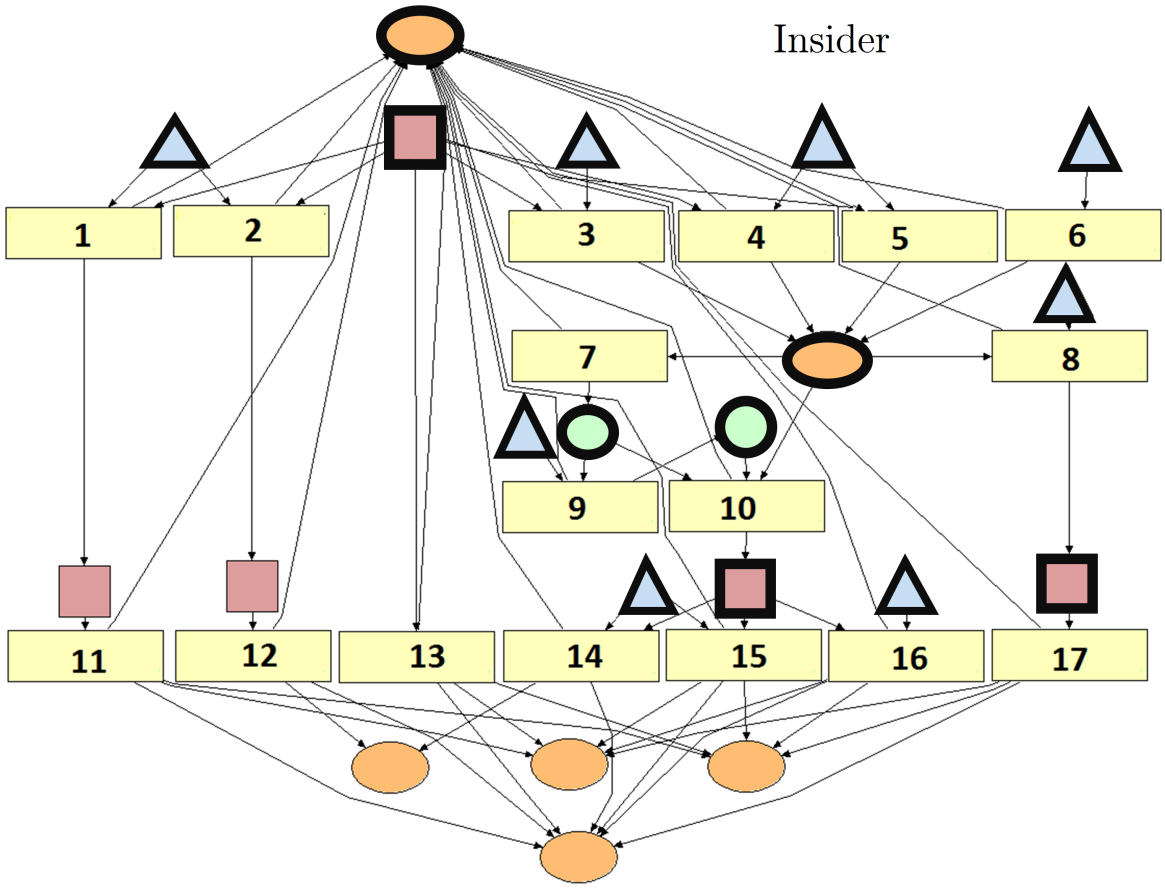

Figure 4.2: Initial state of insider adversary. Elements with a bold outline denote state variables initially held by the adversary.

4.2.1 State Variables Held by Insider

The insider adversary is assumed to be a disgruntled employee or former employee of the utility company. The insider would have intimate familiarity with details of the inner workings of the utility company, and many skills learned on the job. The insider adversary is in some ways the most powerful adversary, because the insider starts with the most access, knowledge, and skills of any adversary considered, and in addition is the only adversary assumed to start with a number of compromised smart meters. See Table 4.3 and Figure 4.2 for a complete listing of state variables initially held by the adversary. This adversary initially starts from a powerful position, but is constrained by a relatively high desire to avoid detection, as can be seen in Table 4.4. The insider wishes to cause as much monetary damage as possible to the utility company without being detected.

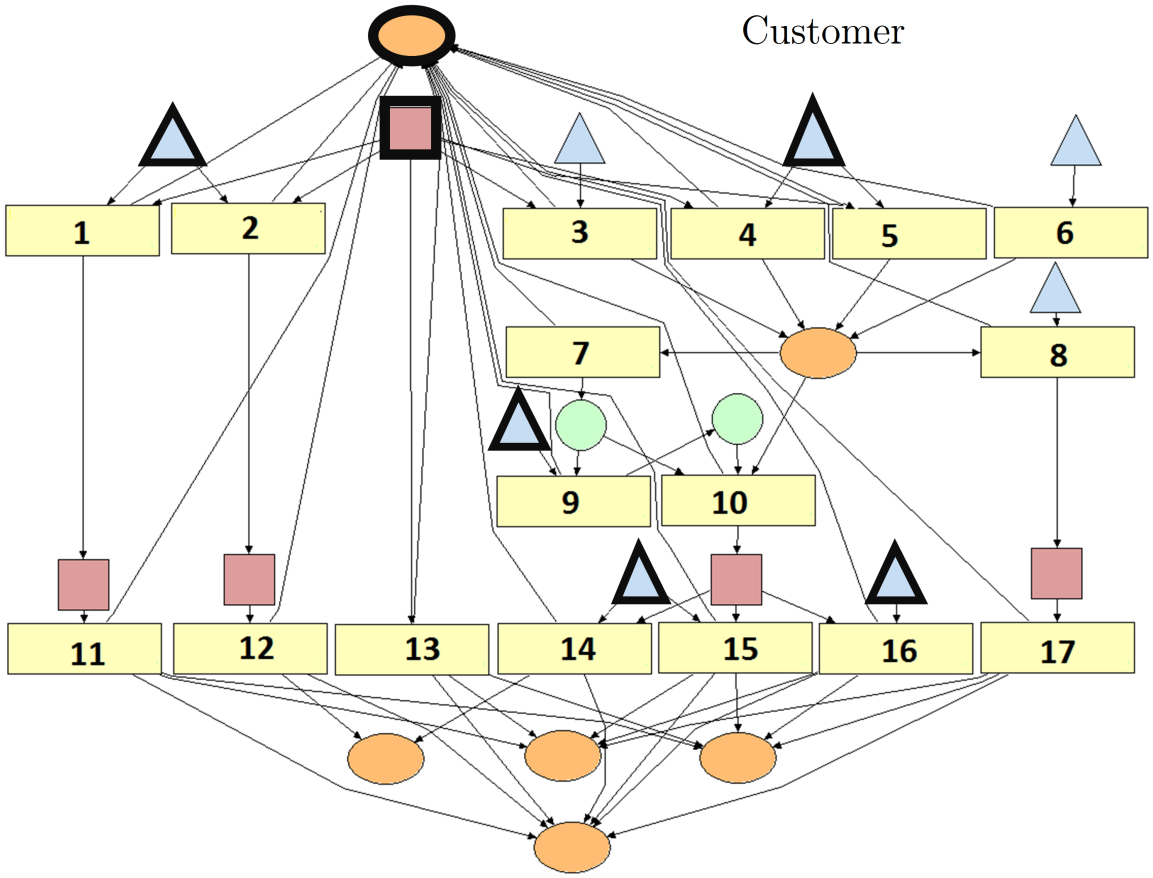

Figure 4.3: Initial state of malicious customer adversary. Elements with a bold outline denote state variables initially held by the adversary.

4.2.2 State Variables Held by Malicious Customer

The malicious customer adversary is assumed to be a customer of the utility company who wishes to steal energy from the utility company by under-reporting energy usage. As can be seen from Table 4.3, and from Figure 4.3, the malicious customer adversary is assumed to have access to a physical smart meter (his or her own) and some skill in various attacks, perhaps obtained via compromises published on the Internet. The customer wants to achieve the goal of under-reporting electricity consumption to lower his or her bill. This unfairly deprives the utility of compensation for providing power to the customer.

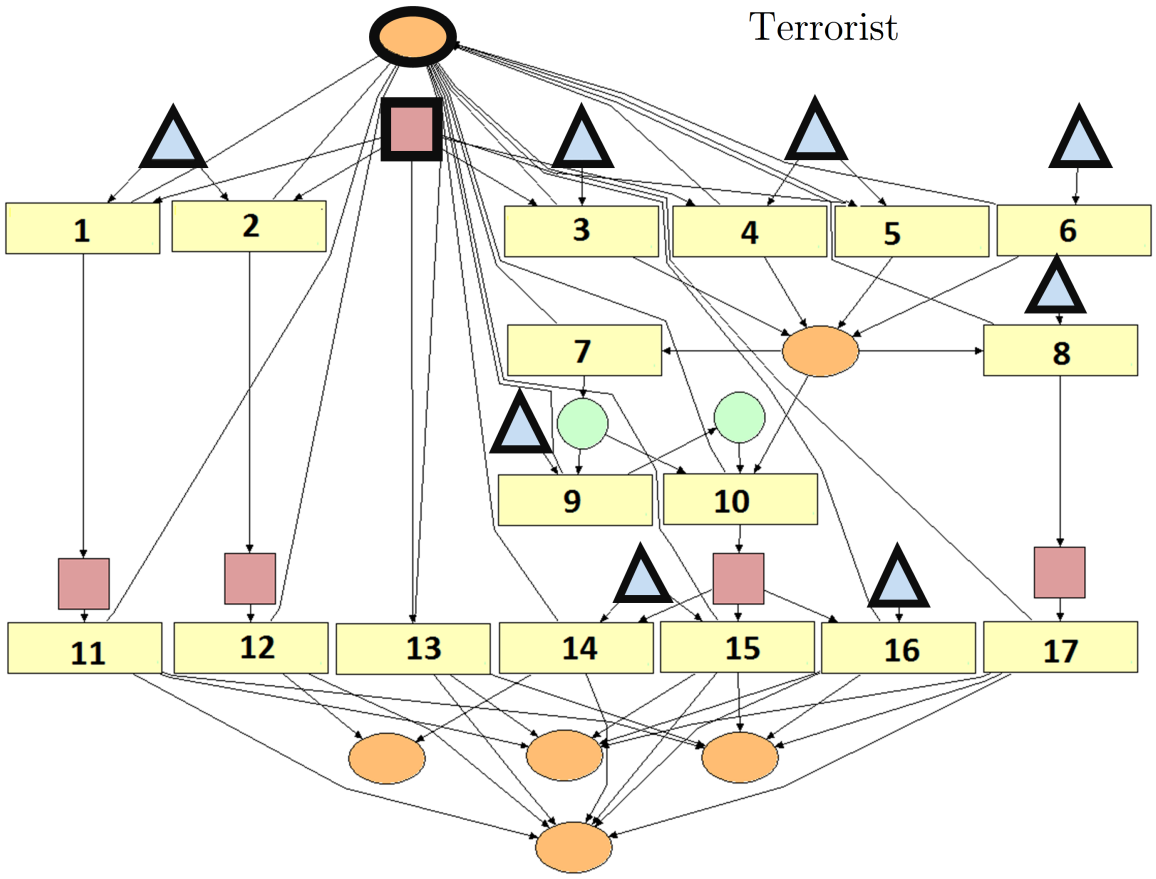

Figure 4.4: Initial state of terrorist adversary. Elements with a bold outline denote state variables initially held by the adversary.

4.2.3 State Variables Held by Terrorist

The terrorist adversary represents an individual or group that wishes to damage the equipment of the utility company and cause a service disruption, perhaps to draw publicity to a cause or as a demonstration of power. The terrorist adversary has fewer initial forms of access, knowledge, and skill than the insider, as can be seen in Table 4.3. Consulate Figure 4.4 for a visual representation of the state variables initially held by this adversary. Table 4.4 shows that the terrorist adversary has the goal of causing the utility company as much monetary damage as possible by interrupting the delivery of power and damaging equipment. The terrorist is assumed to be unconcerned with the possibility of being detected and apprehended, unlike the insider. This means that the terrorist is much more likely to try risky attacks than the insider. This relatively high tolerance for the possibility of detection is expressed in the model by letting the *Undetected* goal have a payoff of zero.

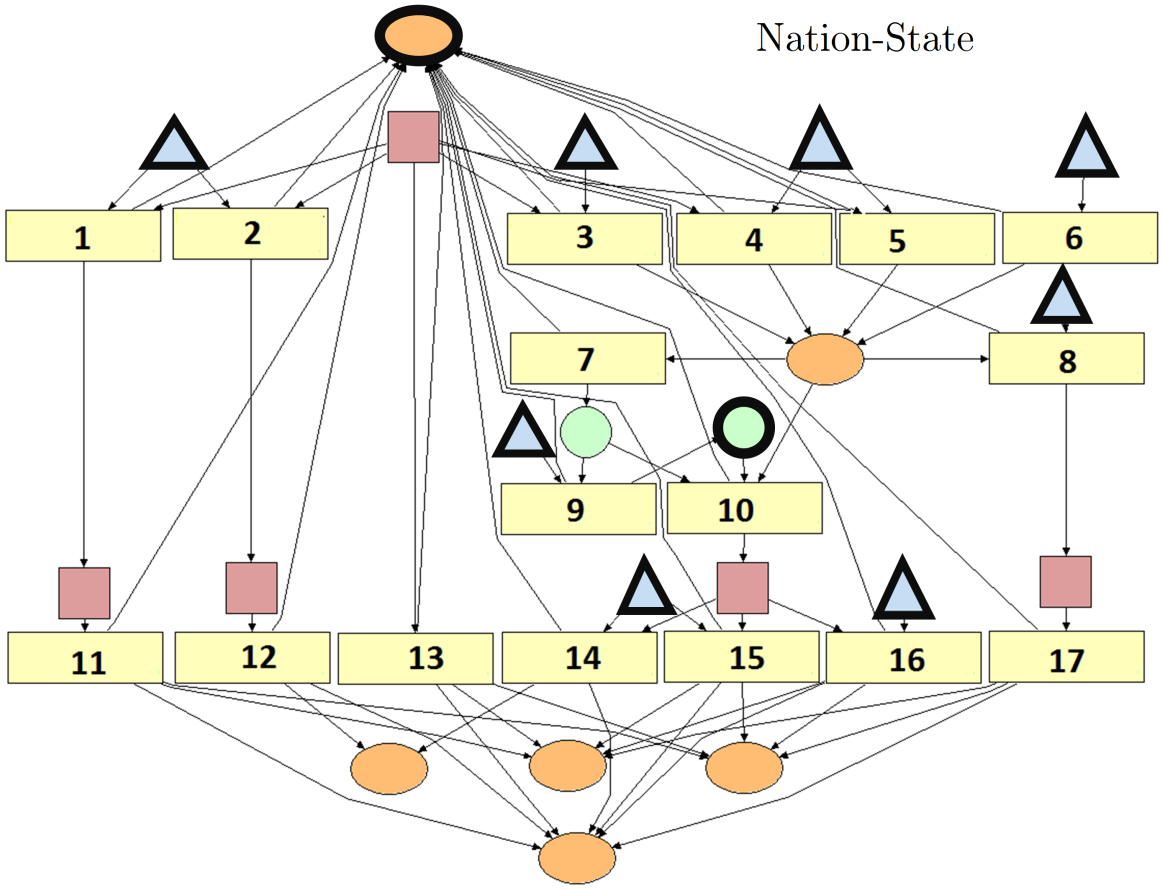

Figure 4.5: Initial state of nation-state adversary. Elements with a bold outline denote state variables initially held by the adversary.

4.2.4 State Variables Held by Nation-State

Finally, the nation-state adversary models a sophisticated, well-funded, malicious, but physically distant state actor that may wish to attack the utility to damage equipment and cause major service disruptions during a conflict. In this model, the nation-state adversary has no initial forms of access into the system. In particular, this adversary does not have physical access to the system, which is a necessary prerequisite for several attack steps. However, this adversary does start with all of the available skills in the AEG, plus knowledge of how traffic is routed. A visual representation of the initial state of the nation-state adversary may be found in Figure 4.5. As can be seen in Table 4.4, the nation-state adversary wishes to damage the utility's equipment and interrupt service, and is strongly motivated to remain undetected.

4.3 Metrics

We use the ADVISE model described above to calculate a number of useful metrics and forecast the adversary's behavior. All results are determined through the creation of performance variables [27] calculated by simulation in Möbius. We took the cross-product of the adversaries {Insider, Customer, Nation State, Terrorist} and the IDS approaches {None, Central, Dedicated, Embedded}, and ran a simulation for every element of this set. We estimated the mean of every performance variable with a 0.95 confidence level and each was run until a 10% relative width for the interval was reached.

The first result we consider is the attack path taken by the adversary through the AEG during the course of the simulation. We determined this result by constructing a set of interval-of-time impulse-reward variables, one reward variable for every attack step in the AEG. If any outcome of a particular attack step is selected during the course of the simulation the performance variable associated with that attack step accumulates a reward. By examining this set of metrics it becomes possible to determine precisely which attacks an adversary attempted during the course of the simulation. This information allows the modeler to determine the paths taken by the adversary through the attack execution graph. The modeler may use this metric to determine commonly exploited weaknesses in the system. A system designer may use this insight to help place defenses in positions where they will do the most good.

The next result, the cost for the adversary to attempt the attack, was also constructed as a set of interval-of-time impulse-reward variables. Any time an outcome of an attack step was attempted between the beginning and end of the simulation run, the cost to attempt that attack step was added to a running total of the accumulated cost. This metric helps a security analyst to determine whether different design decisions result in a system that requires more or less effort for an adversary to successfully attack.

The third metric, the probability that the adversary will remain undetected through the end of the attack, was constructed as an instant-of-time rate-reward variable that returned the value of the *Undetected* goal variable at the end of the simulation. At the beginning of the simulation, the *Undetected* goal variable holds a value of one. Almost every attack step in the AEG has an outcome that represents the event that the adversary is detected if it is executed. If that outcome occurs at some point during the course of the simulation, one of its effects is to set the value of the *Undetected* goal variable to 0. If no outcome representing the detection of the adversary is chosen during the course of the simulation, the value of the *Undetected* goal variable remains 1. In that way we determine whether the adversary was detected during one run of the simulation. Multiple runs of the simulation show the

probability that the adversary will remain undetected through the duration of the attack.

The final and perhaps most important metric, the expected monetary loss incurred by the utility in the event of an attack by an adversary, was also calculated by an instant-of-time rate-reward variable. The integer values held in the *StealElectricity*, *InterruptService*, and *DamageEquipment* goal state variables represent units of damage done by the adversary. We let one unit of *StealElectricity* equal \$600 of loss, one unit of *InterruptService* equal \$10,000 of loss, and one unit of *DamageEquipment* equal \$100,000 of loss. Initially these goal state variables hold a value of 0, but the value can be increased at the successful conclusion of certain attacks. The metric will help an analyst determine the relative benefits of the various intrusion detection systems, by forecasting how much damage the utility sustains when monitored by each IDS deployment configuration.

These results give a security analyst a broad view of the performance of the system under attack and the likely adversary behavior. These quantitative metrics are auditable and reproducible, and help analysts approach the problem of planning and forecasting security consequences in a scientific manner.

CHAPTER 5

RESULTS

The ADVISE model, which we described in Chapter 4, may be executed to calculate a number of security-focused metrics and gain insight into the adversary's behavior. In this chapter, we describe the results we obtained from executing the ADVISE model. We shall first describe the results we obtained from executing the ADVISE security model, and then we shall discuss a sensitivity analysis we conducted.

5.1 Model Execution Results

We executed the ADVISE model to generate four results: the sequence of attack steps the adversary takes to achieve his or her goals, the cost the adversary incurs for attempting an attack on the system; the probability that the utility company will detect the attack; and finally, and perhaps most importantly, the expected monetary loss the utility sustains that occurs due to the activity of the adversary. We shall examine each of these results in turn.

5.1.1 Adversary Attack Path

We shall first examine the sequence of attack steps each adversary (insider, malicious customer, nation-state, and terrorist) uses given each IDS configuration (none, centralized, dedicated, and embedded), as calculated by our simulation in Möbius.

To begin, we shall consider the case in which the utility is not defended by any intrusion detection system, as shown in Figure 5.1. When no IDS is present, the insider will immediately attempt a major routing attack to cause massive damage and service disruptions. They can attempt this attack since they initially start with the skill and access necessary to attempt this attack. The malicious customer uses a longer sequence of attacks to achieve his or her goal. The customer first attempts to install a short range jammer. Once this is accomplished, the customer may use their access to this jammer to try a short range jamming attack to achieve the goal of stealing electricity by underreporting electricity consumption.

The nation-state has the longest attack path of any adversary we consider. As a first step, the nation-state attempts a remote meter compromise to gain control of a number of smart meters, and then tries to extract the cryptographic keys. Once this has been accomplished, the nation-state will gain the ability to route traffic at will in the AMI, and will use this ability to attempt a major routing attack. If the attack is successful, the nation-state will damage the utility's equipment and ability to deliver power. Finally, the terrorist will use his or her physical access to the system to attempt a massively damaging physical attack, which, if successful, results in damage to the system and interrupts service.

Next, we show the attack paths that are used by the various adversaries when confronted by a centralized intrusion detection system, as shown in Figure 5.2. When compared to Figure 5.1, we see that every adversary uses the same attack path, whether they face a system guarded by no IDS or a centralized IDS, with one exception. The risk of detection outweighs the potential benefits for the malicious customer if a centralized IDS is present, so the malicious customer does not attempt any attack at all in this case. All of the other adversaries use the same attack path as they would if faced by no IDS. However, as we shall see later in the chapter, the centralized IDS puts an additional burden on the adversaries and, in some cases, prevents them from doing as much damage to the system, or requires more effort on the part of the adversary to achieve the same amount of damage.

Finally, we present the results of the attack paths chosen by the adversaries if a dedicated IDS or an embedded IDS is used in the system. The attack paths are shown in Figure 5.3 and Figure 5.4. In this case, no adversary attempts to attack the system, except the terrorist. The high probability of detection and the low probability of success discourages most adversaries from even attempting an attack. Unlike other adversaries, the terrorist is not penalized for being detected, and the intrusion detection systems do not have any effect on the probability of success for the physical attack (since they are used to detect cyber attacks), so the terrorist continues to attack despite the presence of the powerful intrusion detection systems.

A modeler may use these attack path results as a guide when considering the placement of system defenses. For example, given the model results, if the utility company chooses to use the centralized intrusion detection system, it should also consider additional measures to make it more difficult for a malicious nation-state to compromise meters remotely.

It would be useful to complement these results with metrics that estimate the cost the adversary incurs for attempting an attack, and the probability that the adversary is detected during the course of the attack. For example, an adversary may attempt the same attack path given a different system configuration, but have a different cost or probability of completing the attack undetected. These metrics give deeper insight into the security properties of the

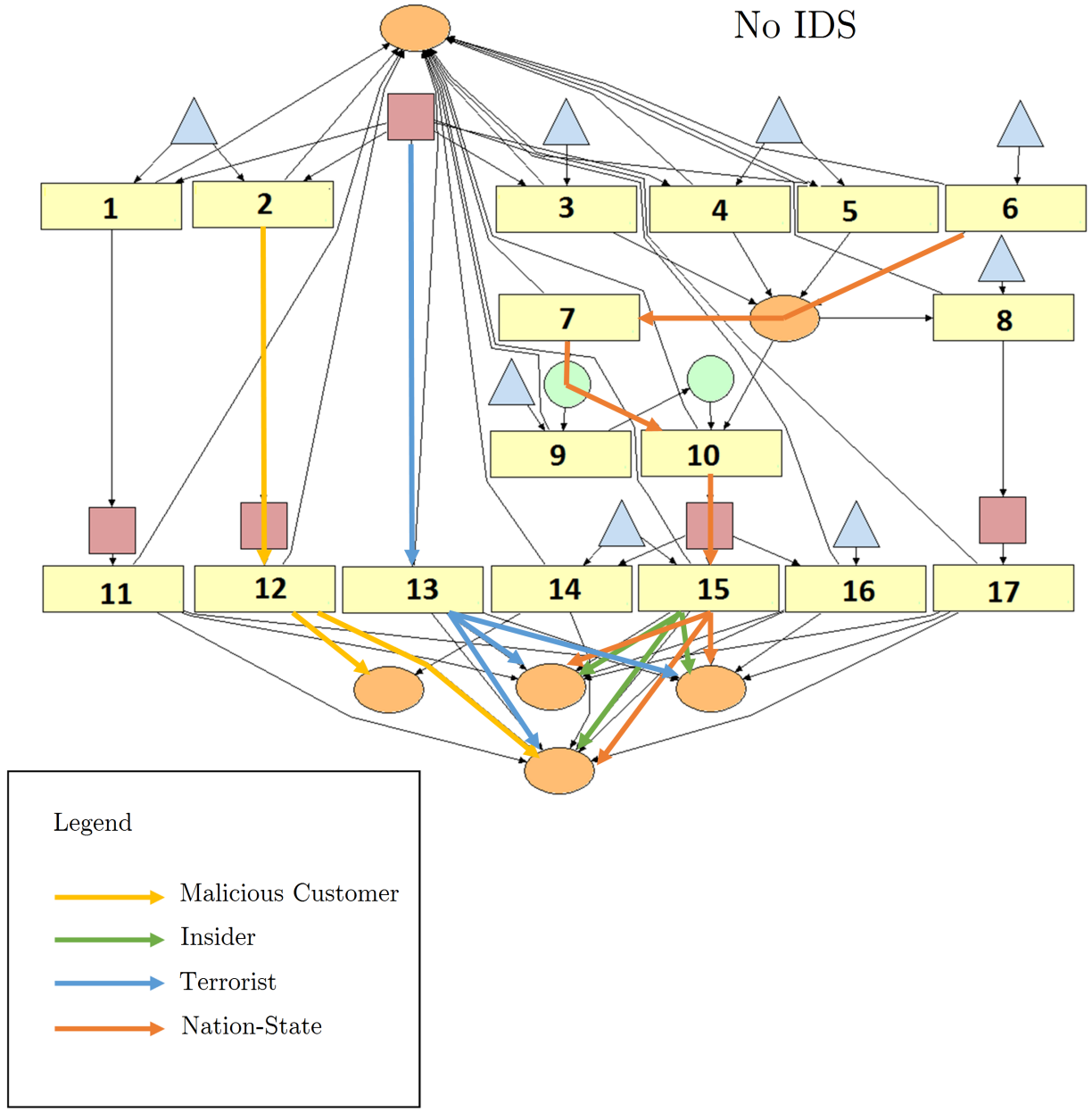

Figure 5.1: Adversary paths through the Attack Execution Graph if the utility company does not use an IDS. If the adversary did not attempt an attack, no path is shown for that adversary.

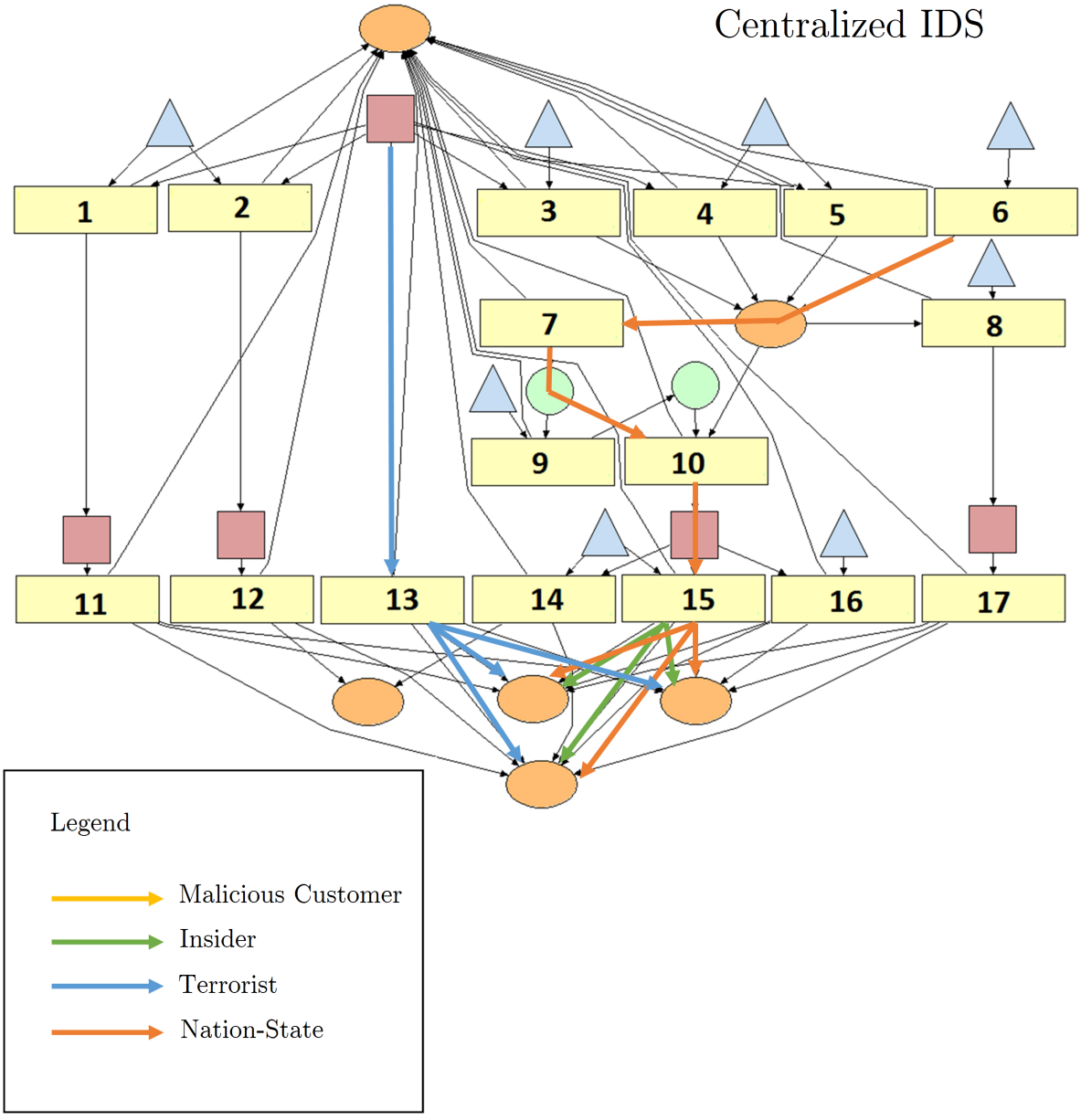

Figure 5.2: Adversary paths through the Attack Execution Graph if the utility company uses a centralized IDS. If the adversary did not attempt an attack, no path is shown for that adversary. Note that for this IDS configuration the malicious customer adversary does not attempt an attack.

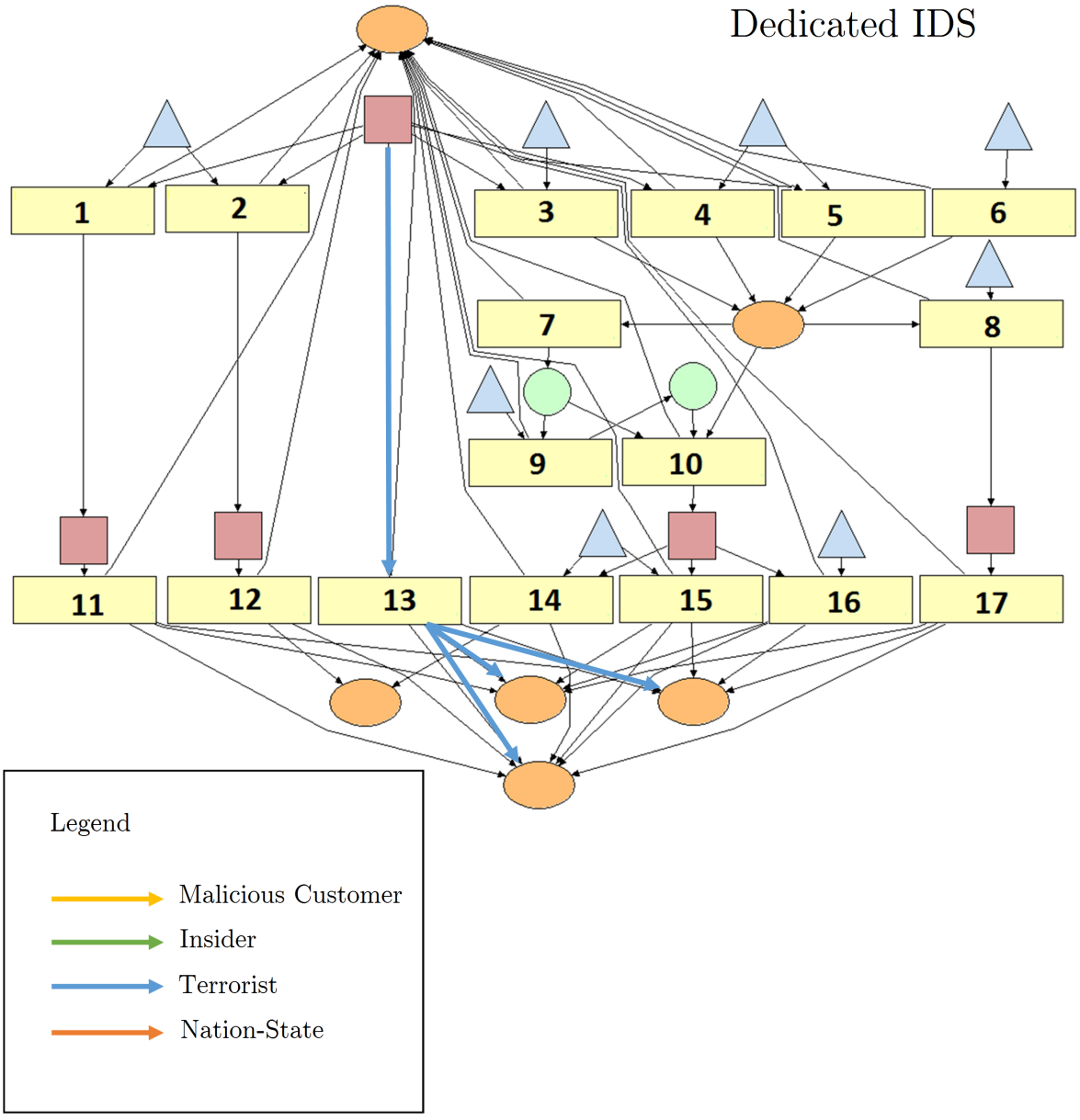

Figure 5.3: Adversary paths through the Attack Execution Graph if the utility company uses a dedicated IDS. If the adversary did not attempt an attack, no path is shown for that adversary. Note that for this IDS configuration only the terrorist adversary attempts an attack.

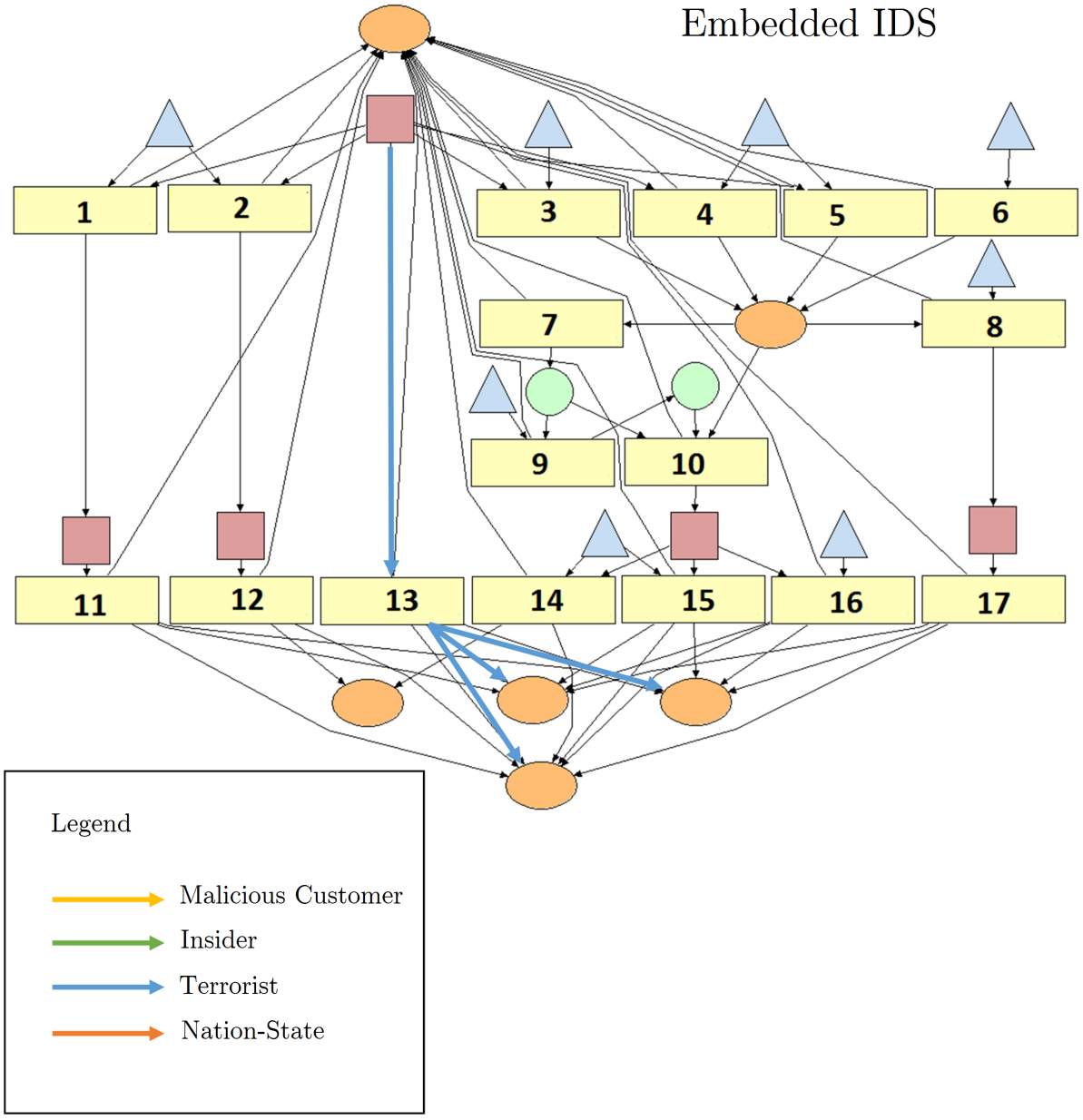

Figure 5.4: Adversary paths through the Attack Execution Graph if the utility company uses an embedded IDS. If the adversary did not attempt an attack, no path is shown for that adversary. Note that for this IDS configuration only the terrorist adversary attempts an attack.

system, and we consider them next.

5.1.2 Cost to Adversary

It often costs an adversary resources to launch an attack, and we seek to estimate the expense to the adversary given each intrusion detection system configuration we consider. This will give deeper insight into the effectiveness of the intrusion detection system. The monetary value of the resources expended by each adversary when faced by each intrusion detection system, which we obtained by executing the ADVISE model, is given in Table 5.1.

The insider adversary spends about $15,000 when faced by no IDS, about $14,000 when faced by a centralized IDS, and nothing when faced by a dedicated IDS or an embedded IDS. The insider spends slightly more when there is no IDS than when there is a centralized IDS because he or she will be more aggressive in attacking a nearly-defenseless system, and attack more frequently, which costs more. The insider spends nothing when confronted by a dedicated IDS or an embedded IDS because he or she will not attack at all in those cases.

When the utility does not utilize an intrusion detection system, the malicious customer adversary spends about $200 on average to attempt an attack. When an intrusion detection system is present in the system, the customer will not attempt any attack at all, and thus incur no costs.

The nation-state will spend about $10,000 on average if the utility does not install an intrusion detection system. However, the average cost quadruples if the nation-state must attempt to defeat a centralized intrusion detection system. This shows that centralized IDS may be useful for deterring some nation-state adversaries. The dedicated and embedded intrusion detection systems discourage the nation-state adversary from attempting any attack and thus the nation-state does not incur the costs associated with the attack.

Finally, the average cost to the terrorist to attempt an attack does not vary at all based on the intrusion detection system. For each of the IDS configurations, the adversary will spend about $1,000 to attack the utility. In this model, the IDS does not make it more or less difficult to attempt a physical attack, as an IDS is meant to prevent cyber attacks, so the terrorist will not deviate from attempting this attack regardless of the IDS approach chosen.

5.1.3 Probability Adversary Remains Undetected Through Attack

One useful measure of the effectiveness of an intrusion detection system is the probability that it will successfully detect an adversary. We calculated these metrics using the ADVISE

IDS	Adversary	Cost	Error
None	Insider	$15.3K	+/- $330
	Customer	$200	+/- $0.01
	Nation-State	$9.35K	+/- $55
	Terrorist	$1K	+/- $0
Centralized	Insider	$13.6K	+/- $217
	Customer	$0 *	+/- $0
	Nation-State	$39K	+/- $105
	Terrorist	$1K	+/- $0
Dedicated	Insider	$0 *	+/- $0
	Customer	$0 *	+/- $0
	Nation-State	$0 *	+/- $0
	Terrorist	$1K	+/- $0
Embedded	Insider	$0 *	+/- $0
	Customer	$0 *	+/- $0
	Nation-State	$0 *	+/- $0
	Terrorist	$1K	+/- $0

Table 5.1: Cost incurred by the adversary, given a particular IDS. Costs of 0 with asterisks (*) occur as a result of the adversary choosing not to attack.

model we implemented in Möbius, and present them in Table 5.2.

We first calculated a baseline: the probability of detecting each adversary if no intrusion detection system is present in the system. The customer is quite likely to remain undetected if no IDS is present in the system, while the other adversaries have a lower probability of remaining undetected since the attacks they are attempting are risky and complicated. When faced by a centralized intrusion detection system and compared to the baseline, the insider has a slightly lower chance of being detected, because the insider acts more aggressively and attacks more frequently when no IDS is present. When compared to the baseline, all the other adversaries have the same chance of being detected or do not choose to attack when faced by a centralized IDS. When faced by a dedicated IDS or an embedded IDS, no adversary (with the exception of the terrorist) attempts any attack and thus has no probability of being detected. The probability of detection does not change for the terrorist, regardless of the intrusion detection system used, because the terrorist always attempts a physical attack which is not affected by intrusion detection systems.

These results initially appear nonintuitive (because no IDS seems to increase the likelihood of detecting the adversary) but can be easily explained. The intrusion detection system can be so effective that the adversary will not even attempt an attack, because the risk of detection does not outweigh the potential reward. When this happens, the adversary

IDS	Adversary	Probability	Error
None	Insider	0.382	+/- 0.021
	Customer	0.947	+/- 0.005
	Nation-State	0.641	+/- 0.009
	Terrorist	0.200	+/- 0.012
Centralized	Insider	0.401	+/- 0.015
	Customer	1 *	+/- 0
	Nation-State	0.632	+/- 0.009
	Terrorist	0.200	+/- 0.012
Dedicated	Insider	1 *	+/- 0
	Customer	1 *	+/- 0
	Nation-State	1 *	+/- 0
	Terrorist	0.200	+/- 0.012
Embedded	Insider	1 *	+/- 0
	Customer	1 *	+/- 0
	Nation-State	1 *	+/- 0
	Terrorist	0.200	+/- 0.012

Table 5.2: The probability that the adversary will remain undetected by the end of the attack. Results with asterisks (*) denote that the adversary went undetected because they did not attempt to attack the system.

does not attack and remains undetected. If the adversary had attempted the attack, the probability of detecting the adversary would have risen dramatically.

5.1.4 Expected Cost to Utility

The monetary loss that each adversary inflicts on a utility given each intrusion detection system is one of the most important metrics for successfully completing a cost-benefit analysis. The monetary loss includes the loss of equipment, service disruptions, and loss of revenue due to energy theft. The results we obtained from the execution of our ADVISE model may be found in Table 5.3.

We see that adversaries cause a great deal of damage when no intrusion detection system is present in the AMI. Every adversary, except the terrorist, is predicted to do much less damage when the centralized IDS is utilized in the AMI, and no damage when the dedicated IDS or embedded IDS are used. The terrorist does the same amount of damage, regardless of the intrusion detection system configuration used, because the terrorist favors physical attacks, which the intrusion detection system cannot detect.

A utility company can use these metrics to compare intrusion detection approaches. The

IDS	Adversary	Monetary Damage	Error
None	Insider	$11.6M	+/- $312K
	Customer	$379	+/- $1.84
	Nation-State	$6.3M	+/- $60K
	Terrorist	$1.02M	+/- $93K
Centralized	Insider	$2.9M	+/- $54K
	Customer	$0 *	+/- $0
	Nation-State	$1.58M	+/- $16K
	Terrorist	$1.02M	+/- $93K
Dedicated	Insider	$0 *	+/- $0
	Customer	$0 *	+/- $0
	Nation-State	$0 *	+/- $0
	Terrorist	$1.02M	+/- $93K
Embedded	Insider	$0 *	+/- $0
	Customer	$0 *	+/- $0
	Nation-State	$0 *	+/- $0
	Terrorist	$1.02M	+/- $93K

Table 5.3: Cost incurred by the utility company as a result of the actions of each adversary, given a particular IDS. Results with asterisks (*) denote that the adversary did no damage because they did not attempt to attack the system.

expected monetary loss sustained by a utility company, M, for an IDS configuration, $i \in IDS$, can be calculated with Equation 5.1

$$M_i = \sum_a N_a * D_a \tag{5.1}$$

where N is the expected number of attack attempts and D_a is the expected monetary damage to the system, D, per adversary, $a \in Adversaries$.

Consider a hypothetical utility that estimates 1,000 attack attempts by unscrupulous customers, 5 attack attempts by an insider, 5 attack attempts by a nation-state, and 2 attack attempts by a terrorist over a 20-year period.

Using Equation 5.1 and the numbers in Table 5.3, we calculate the results shown in Table 5.4. The utility can use Table 5.4, along with information about installation and maintenance costs provided by vendors, to help determine the most cost-effective architecture for its system.

IDS Approach	Monetary Loss
None	$91,919,000
Centralized	$16,540,000
Dedicated	$2,040,000
Embedded	$2,040,000

Table 5.4: Estimated monetary loss by IDS approach over a 20-year period.

5.2 Sensitivity Analysis

Models built using the ADVISE formalism utilize many input parameters. It may be difficult to obtain accurate estimates for some of these input parameters. For example, it may be next to impossible to precisely determine the payoff an adversary may achieve for accomplishing a goal, or the costs to attempt certain attack steps, or the effectiveness of an intrusion detection system which has not yet been implemented. The metrics that interest the security analyst may be sensitive or insensitive to the input parameters. If the metric is insensitive to the input parameter, it suggests that the security analyst need not be concerned with obtaining an accurate estimate for it. If, on the other hand, the metric of interest is sensitive to the input parameter, the modeler may have to accept uncertainty regarding the results. We present such a sensitivity analysis to explore the stability of the model we develop in this case study.

5.2.1 Sensitivity Analysis Methodology

We use two different techniques in the sensitivity analysis: one that analyzes individual parameters one at a time at high resolution, and one that analyzes combinations of parameters at a lower resolution.

Parameters Studied in Isolation: To begin, we analyze the various cost, payoff, and effectiveness of IDS parameters individually, one at a time. This gives a detailed, low-level view of the model parameters in isolation.

To begin, we define 10 cost scaling factors. Each of these factors scales a cost on an individual attack step, except one factor which scales the cost on two attack steps. The correspondence between cost scaling factors and attack steps is shown in Table 5.5. We do 6 simulations for each cost scaling factor, assigning it to each of the following values in turn: 0.2, 0.5, 1.0, 1.5, 2.0, and 5.0. This, in turn, scales the original cost of the attack step by the corresponding amount. We can then estimate the overall monetary damage incurred by the utility given the scaled cost of the attack step. This allows us to analyze whether an attack

would do more damage if the attack were cheaper, or less damage if the attack was more expensive. For example, if we examine Table 5.7, we see that the insider adversary causes the same amount of loss of availability and integrity when *costScaleFactor 15* (the cost scale factor for the *Major Routing* attack) is set to a value of 0.2 as it does when it is set to the default value of 1.0. However, the insider does about a tenth as much damage if it is set to 5.0 as opposed to 1.0. This result can be interpreted to mean that the insider will do no more damage if the *Major Routing Attack* is substantially cheaper, but will do much less damage if it is substantially more expensive.

Next, we define 5 payoff scaling factors. Each of these factors scales the payoff an adversary obtains for achieving a particular goal. Table 5.6 shows the goal that each payoff factor scales. We vary each payoff scaling factor by assigning it to each of the following values in turn: 0.2, 0.5, 1.0, 1.5, 2.0, and 5.0. This allows us to examine how much the motivation of an adversary changes given different payoff values.

Finally, we let the *IDSMultiplier*, which scales the effectiveness of the intrusion detection system, take one of six different values: 0.0001, 0.001, 0.01, 0.05, 0.1, and 0.25. The *IDSMultiplier* has an effect on the probability of success and failure for various attack steps. Often, the effect is to simply multiply the probability of success by the *IDSMultiplier* and adjust the probability of failure in a corresponding way (so that the sum of the probabilities total to 1), though sometimes the *IDSMultiplier* is used in a slightly more complicated way. Since it scales the probability of success for the adversary on an attack step, a lower value for the *IDSMultiplier* is better for the defender.

Each time a scaling factor was varied, whether it was associated with cost, payoff, or the *IDSMultiplier*, an experiment was run 16 times, one for each of the cross-product of the adversaries and the IDSes. There were 10 cost scaling factors, 5 payoff scaling factors, and 1 *IDSMultiplier* scaling factor, for a total of 16 scaling factors. Each of the 16 scaling factors was assigned to one of six values in turn. Therefore, a total of $16 * 16 * 6 = 1536$ experiment runs were conducted for this phase of the sensitivity analysis.

Parameters Studied in Combination: Next, we analyze the consequences of varying the cost, payoff and effectiveness of IDS parameters in combination.

First, we define a cost scaling factor, *CostScaleFactor*. This scaling factor scales each attack cost (and the *Undetected* goal payoff value) at the same time. We have defined the *Undetected* goal payoff as a cost, rather than a payoff, because the adversary always begins with the payoff from this goal, and can only maintain or lose it, but never increase it. By an equivalent definition, the *CostScaleFactor* is the union of all the cost scale factors in Table 5.5 and the *payoffScaleFactor1* defined in Table 5.6. As the cost scaling factor grows larger the cost for the adversary to perform any chain of actions increases (or remains the same, in

Scale Factor Name	Corresponding Attack Step(s)
costScaleFactor1	Install Long Range Jammer Install Short Range Jammer
costScaleFactor3	Install Malicious Smart Meter
costScaleFactor4	Physical Smart Meter Exploit
costScaleFactor5	Mass Meter Compromise
costScaleFactor6	Remote Smart Meter Exploit
costScaleFactor8	Create Botnet
costScaleFactor13	Physical Attack
costScaleFactor15	Major Routing Attack
costScaleFactor16	Byzantine Attack

Table 5.5: The scaling factors in the left column scale the costs of the attack steps in the right column.

Scale Factor Name	Corresponding Goal
payoffScaleFactor1	Undetected Goal
payoffScaleFactor2	Compromised Smart Meters Goal
payoffScaleFactor3	Steal Energy Goal
payoffScaleFactor4	Interrupt Service Goal
payoffScaleFactor5	Damage Equipment Goal

Table 5.6: The scaling factors in the left column scale the payoffs of the goals in the right column.

the case of attacks that have no cost). We vary the *CostScaleFactor* in 10 increments of 0.2 between 0.2 and 2.0.

Similarly, we define a payoff scaling factor, *PayoffScaleFactor*. This scaling factor scales the payoff for each goal at the same time, with the exception of the *Undetected* goal. We may define the *PayoffScaleFactor* as the union of all the payoff scale factors (except *payoffScaleFactor1*) in Table 5.6. A larger value for the *PayoffScaleFactor* will result in a greater adversary payoff for achieving one or more goals, while a smaller value for the *PayoffScaleFactor* will result in a smaller adversary payoff for achieving one or more goals, except, of course, for the case when goals have a payoff of zero. We vary the *PayoffScaleFactor* in increments of 0.2 between 0.2 and 2.0, just as we did for the *CostScaleFactor*.

Lastly, we used the *IDSMultiplier* we defined previously. This scale factor is allowed to take one of the following four values: 0.001, 0.01, 0.1 and 0.2.

5.2.2 Insider

The first adversary we consider is the insider. After examining Table 5.7 and Table 5.8 it becomes clear that this adversary's effect on the system is most sensitive to scale factors *costScaleFactor15*, *payoffScaleFactor1*, and *payoffScaleFactor5*. These factors scale the cost of performing the *Major Routing Attack*, the payoff of the *Undetected* goal, and the payoff of the *DamageEquipment* goal, respectively. Scaling the other cost, payoff and detection factors according to the scheme we proposed in our methodology did not affect the total monetary loss sustained by the utility.

These results are intuitive. The insider already has a strong foothold in the system, as can be seen in Table 4.3, and so is uninterested in attempting attacks that would increase his or her knowledge of or access to the AMI network. In addition, the insider is not interested in any attack performed with goal of stealing electricity. Therefore, it is reasonable that the insider is insensitive to most of the individual attack costs. Furthermore, as shown in Table 4.4, the adversary is not interested in stealing electricity or compromising smart meters, and receives a relatively low payoff for compromising the availability of the AMI. This explains why varying the factors that scale the corresponding payoffs has no discernible effect on the model. The insider's effect on the system is most sensitive to the scale factors corresponding to the two goals with the highest payoffs, and the attack which the adversary uses to maximize payoff.

If we examine Figure 5.5, we see that the insider will cause the same amount of loss to the utility, when there is no IDS present, for almost every combination of the payoff scale factor and cost scale factor. For this system configuration, the insider will cause no damage to the

system only when the payoff is set very low and the cost is high. In every other cost/payoff combination, the insider will do massive damage to the system. We see more variability in Figure 5.6, which displays the monetary damage done by the insider when a centralized IDS is present in the AMI. However, a clear majority of cost/payoff scale factor combinations give results identical to the default cost/payoff scale factor combination. This gives some confidence that this configuration of the model may tolerate small to moderate errors in the estimates of the insider's payoffs and costs. In contrast, Figures 5.7 and 5.8 show that the distributed and embedded configurations are quite sensitive to changes to the estimates of the insider's payoffs and costs. For example, keeping the payoff scale factor set to its default value of 1.0 and reducing the cost scale factor to 0.8 results in around $1,000,000 in damage, compared to $0 in damage when both factors are set to their default values of 1.0.

The results of this sensitivity analysis may help a utility make more effective and informed design and policy choices. The sensitivity analysis suggests that the utility company should focus on (a) decreasing the payoff an insider may receive from performing an attack by monitoring and improving employee job satisfaction, and (b) increasing the penalty an insider incurs when caught performing the attack. In addition, the sensitivity analysis warns that the utility should take care to not overestimate the effectiveness of the distributed and embedded intrusion detection systems, since small increases in the adversary's payoff or small decreases in the adversary's cost may be enough to motivate the adversary to attempt massively damaging attacks. The utility's modelers should be aware that small errors in their estimates of these values may cause drastic differences in the results given in the model.

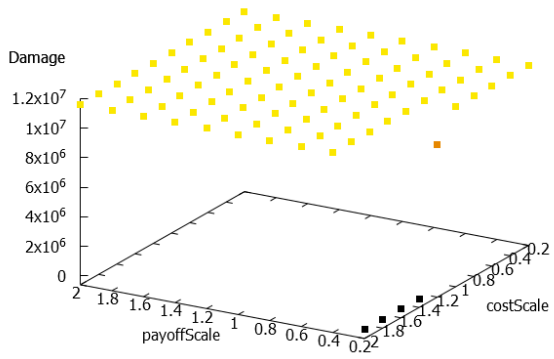

Figure 5.5: Monetary damage done by the insider adversary and no IDS.

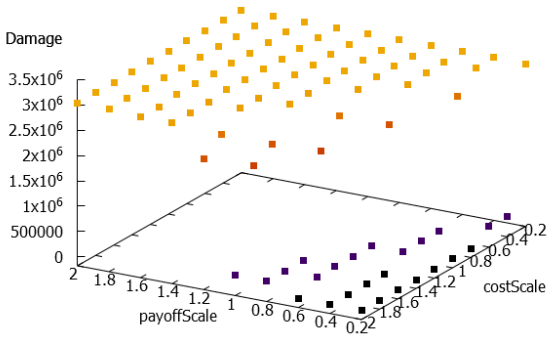

Figure 5.6: Monetary damage done by the insider adversary with $ScaleFactorIDS = \{0.001, 0.01, 0.1, 0.2\}$ and centralized IDS.

Variable	IDS Type	**0.2**	**0.5**	**1.0**	**1.5**	**2.0**	**5.0**
costScaleFactors 1,3,4,5,6,8,9,13, and 16	None	11600	11600	11600	11600	11600	11600
	Centralized	3038	3038	3038	3038	3038	3038
	Distributed	0	0	0	0	0	0
	Embedded	0	0	0	0	0	0
costScaleFactor 15	None	11600	11600	11600	11600	11600	11600
	Centralized	3038	3038	3038	2996	2414	229
	Distributed	0	0	0	0	0	0
	Embedded	0	0	0	0	0	0
payoffScaleFactor 1	None	11600	11600	11600	11600	11600	11600
	Centralized	3038	3038	3038	3038	2996	0
	Distributed	1051	1019	0	0	0	0
	Embedded	1051	1019	0	0	0	0
payoffScaleFactors 2,3, and 4	None	11600	11600	11600	11600	11600	11600
	Centralized	3038	3038	3038	3038	3038	3038
	Distributed	0	0	0	0	0	0
	Embedded	0	0	0	0	0	0
payoffScaleFactor 5	None	11600	11600	11600	11600	11600	11600
	Centralized	229	1848	3038	3038	3038	3038
	Distributed	0	0	0	1019	1019	1051
	Embedded	0	0	0	1019	1019	1051

Table 5.7: Cost incurred by the utility company due to an insider attack, for each scaling factor value. Values are in thousands of dollars.

Variable	IDS Type	**0.0001**	**0.001**	**0.01**	**0.05**	**0.1**	**0.25**
IDSMultiplier	Centralized	3038	3038	3038	3038	3038	3038
	Distributed	0	0	0	0	0	0
	Embedded	0	0	0	0	0	0

Table 5.8: Cost incurred by the utility company due to an insider attack, for each scaling factor value. Values are in thousands of dollars.

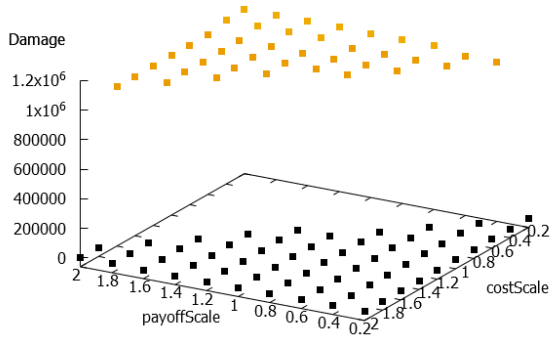

Figure 5.7: Monetary damage done by the insider adversary with $ScaleFactorIDS = \{0.001,0.01,0.1,0.2\}$ and dedicated IDS.

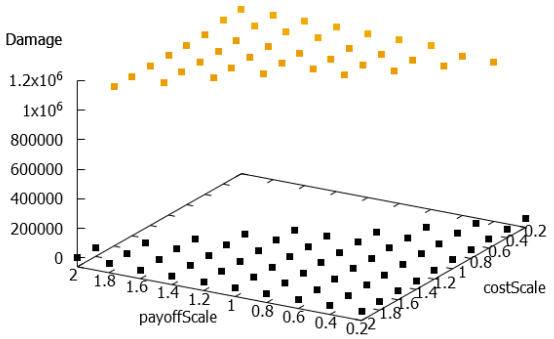

Figure 5.8: Monetary damage done by the insider with $ScaleFactorIDS = \{0.001,0.01,0.1,0.2\}$ and embedded IDS.

5.2.3 Malicious Customer

We turn to an analysis of the criminal customer, who wishes to steal electricity (as defined in Table 4.4). The adversary has access to smart meters, but is not particularly skilled in attacks or knowledgeable about the system in general (as shown in Table 4.3).

We begin with an analysis of the various individual cost, payoff and detection scaling factors. To begin, we can see from Table 5.9 that the adversary's effect on the system is not sensitive to any of the individual cost scaling factors, nor is it sensitive to the factors which scale the payoff of the *CompromiseSmartMeter* goal, the *DamageEquipment* goal, or the *InterruptService* goal. The customer obtains no payoff for compromising smart meters, damaging equipment, or interrupting service, so we would expect that varying the corresponding scaling factors would have no effect on the model. It is more surprising that the adversary's effect on the system is so insensitive to cost, but the costs of the attacks are small compared to the payoff of either stealing electricity or remaining undetected, which explains the result.

Continuing the analysis of Table 5.9, along with Table 5.10, the malicious customer causes more damage to the AMI with a centralized IDS than he or she otherwise would have if the payoff for the *Undetected* goal is half or less of its original value. This means that if the penalty for getting caught is lessened the customer is tempted to try the attack on the centralized IDS. The customer also causes more damage to the AMI with a centralized IDS if the payoff of the *StealEnergy* goal is increased, or if the centralized IDS is assumed to be very weak. However, when confronted by the dedicated or embedded IDS, the customer will

not attempt an attack even if the penalty for getting caught is very low, or the payoff for stealing electricity is very high, or the IDS is assumed to be relatively ineffective.

We next consider the aggregated payoff, aggregated cost, and detection scale factors. If we examine Figure 5.9, we see that if there is no intrusion detection system the malicious customer will be motivated to attempt the attack and succeeds in stealing electricity except in the cases when the payoff is quite low and at the same time the cost is relatively high. In contrast, Figures 5.10 and 5.11 show that small changes to the overall cost or payoff estimates can have drastic effects on the output of the model. Figures 5.12 and 5.13 show that the estimate monetary loss for the utility given the dedicated and embedded IDS deployment approaches is not sensitive to changes in estimates of the adversary costs and payoffs.

The utility should carefully review the information contained in this sensitivity analysis. The model of the centralized approach is very sensitive to small decreases in cost and increases in payoff for the malicious customer adversary. The initial results presented in the previous section suggested that the centralized IDS would stop all attack attempts by the malicious customer adversary, but this sensitivity analysis suggests that this may not be the case, unless the payoff and cost estimates are very accurate. If more accurate estimates are not available, a conservative approach would suggest that the utility should either assume that at least some percentage of malicious customers will successfully reach the goal. This may lead the utility to consider more effective intrusion detection approaches.

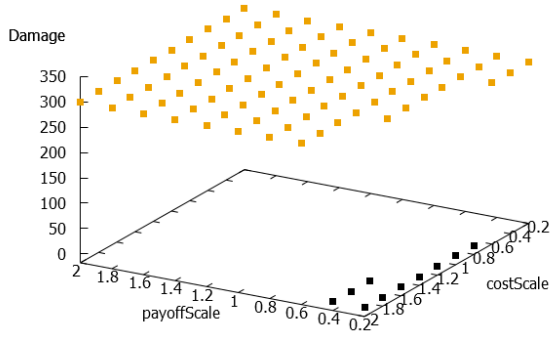

Figure 5.9: Monetary damage done by the malicious customer adversary and no IDS.

Variable	IDS Type	**0.2**	**0.5**	**1.0**	**1.5**	**2.0**	**5.0**
costScaleFactors 1,3,4,5,6,8,9,13, 15 and 16	None	301	301	301	301	301	301
	Centralized	0	0	0	0	0	0
	Dedicated	0	0	0	0	0	0
	Embedded	0	0	0	0	0	0
payoffScaleFactor 1	None	301	301	301	301	301	301
	Centralized	167	167	0	0	0	0
	Dedicated	0	0	0	0	0	0
	Embedded	0	0	0	0	0	0
payoffScaleFactors 2,4, and 5	None	301	301	301	301	301	301
	Centralized	0	0	0	0	0	0
	Dedicated	0	0	0	0	0	0
	Embedded	0	0	0	0	0	0
payoffScaleFactor 3	None	0	301	301	301	301	301
	Centralized	0	0	0	167	167	167
	Dedicated	0	0	0	0	0	0
	Embedded	0	0	0	0	0	0

Table 5.9: Cost incurred by the utility company due to a malicious customer attack, for each scaling factor value. Values are in dollars.

Variable	IDS Type	**0.0001**	**0.001**	**0.01**	**0.05**	**0.1**	**0.25**
IDSMultiplier	Centralized	0	0	0	0	0	221
	Dedicated	0	0	0	0	0	0
	Embedded	0	0	0	0	0	0

Table 5.10: Cost incurred by the utility company due to a malicious customer attack, for each scaling factor value. Values are in dollars.

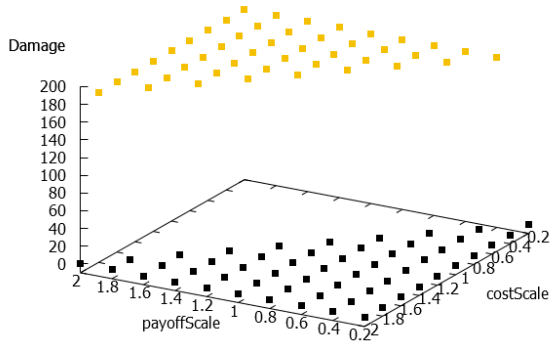

Figure 5.10: Monetary damage done by the malicious customer adversary with $ScaleFactorIDS = \{0.001, 0.01, 0.1\}$ and centralized IDS.

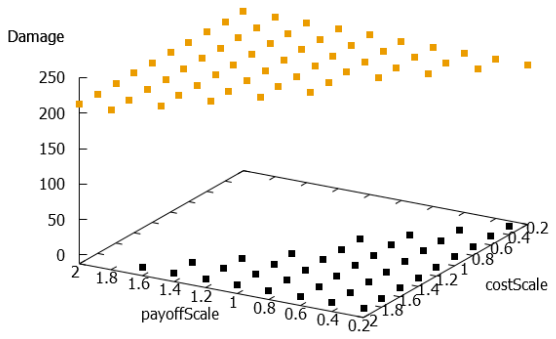

Figure 5.11: Monetary damage done by the malicious customer adversary with $ScaleFactorIDS = 0.2$ and centralized IDS.

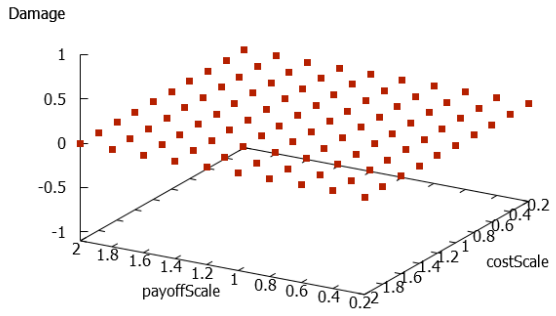

Figure 5.12: Monetary damage done by the malicious customer adversary with $ScaleFactorIDS = \{0.001, 0.01, 0.1\}$ and IDS=$\{$dedicated, embedded$\}$.

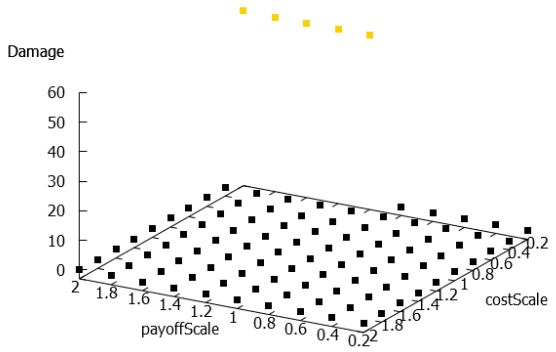

Figure 5.13: Monetary damage done by the malicious customer adversary with $ScaleFactorIDS = 0.2$ and IDS = $\{$dedicated, embedded$\}$.

5.2.4 Nation-State

The nation-state adversary wishes to damage equipment and interrupt service at the utility, but pays a high cost if detected (this is shown in Table 4.4). The adversary is highly skilled in various attacks and fairly knowledgeable about the system, but has minimal access (as can be seen in Table 4.3).

We first consider the individual scale factors. Examining Table 5.11 and Table 5.12 we see that the adversary's effect on the system only changes when the cost scaling factor

associated with the *Remote Smart Meter Compromise* attack is varied, or when the payoff scaling factor that scales the payoff of the *Undetected* goal is varied. This result is sensible: the nation-state adversary is not nearly as constrained by cost as the other adversaries, since it is well-resourced. They are slightly sensitive to the relatively large cost of doing a massive remote smart meter compromise. The adversary is not very sensitive to attack step costs, but the nation-state risks massive reputational damage, possible sanctions, and perhaps even war if they are identified as an attacker. Therefore, their behavior is sensitive to changes in the cost of being detected.

Next, we consider the aggregated cost, payoff and detection scale factors. Figure 5.14 shows us that the nation-state will always cause the same amount of damage irrespective of any of the cost and payoff levels considered when faced with an AMI that employs no IDS. Similarly, Figures 5.15 and 5.16 show that the adversary faced by the centralized IDS will do massive damage almost irrespective of cost, payoff, or the IDS scale factor levels; the only exceptions are cases when the payoff is very low and the cost is very high, and in these cases the adversary does no damage. Figures 5.17 and 5.18 show that nation-state adversary will do no damage to the system when confronted by a dedicated or embedded IDS for any combination of cost, payoff, or IDS detection level considered, with one exception. The one exception is when the payoff is set to double its normal value, the cost is a fifth of its normal value, and the IDS multiplier is set to its weakest level, in which case the adversary is motivated to attack and causes a small amount of damage.

The modeler may use these results to inform their design decisions. The most immediately apparent result is the critical importance of preventing the smart meters from being remotely compromised en masse. Both the embedded and dedicated intrusion detection systems seem to offer the defender a very good way to accomplish this goal. Even the centralized intrusion detection system offers some improvement over no intrusion detection system in this respect. The sensitivity analysis also makes clear that the adversary's estimated impact on the system is relatively insensitive to cost, payoff, and detection parameters. This may give the modeler increased confidence that the model results accurately reflect reality even if the input parameters were not estimated with high precision.

5.2.5 Terrorist

The terrorist adversary has the goal of damaging equipment and interrupting service, but is not concerned with the possibility of being detected by the defender (these preferences may be seen in Table 4.4). It becomes clear after examining Table 4.3 that the adversary is not very highly skilled, and has limited knowledge of and access to the AMI.

Variable	IDS Type	**0.2**	**0.5**	**1.0**	**1.5**	**2.0**	**5.0**
costScaleFactors 1,3,4,5,8,9,13, 15, and 16	None	6243	6243	6243	6243	6243	6243
	Centralized	1601	1601	1601	1601	1601	1601
	Dedicated	0	0	0	0	0	0
	Embedded	0	0	0	0	0	0
costScaleFactor 6	None	6243	6243	6243	6243	6243	6243
	Centralized	1601	1601	1601	1601	1601	0
	Dedicated	5	5	0	0	0	0
	Embedded	0	0	0	0	0	0
payoffScaleFactor 1	None	6243	6243	6243	6243	6243	6243
	Centralized	1601	1601	1601	1601	1601	1601
	Dedicated	0	0	0	0	5	5
	Embedded	0	0	0	0	0	0
payoffScaleFactors 2,3,4, and 5	None	6243	6243	6243	6243	6243	6243
	Centralized	1601	1601	1601	1601	1601	1601
	Dedicated	0	0	0	0	0	0
	Embedded	0	0	0	0	0	0

Table 5.11: Cost incurred by the utility company due to a nation-state attack, for each scaling factor value. Values are in thousands of dollars.

Variable	IDS Type	**0.0001**	**0.001**	**0.01**	**0.05**	**0.1**	**0.25**
IDSMultiplier	Centralized	1604	1604	1596	1601	1598	1621
	Dedicated	0	0	0	0	0	0
	Embedded	0	0	0	0	0	0

Table 5.12: Cost incurred by the utility company due to a nation-state attack, for each scaling factor value. Differences in the result are within the confidence interval. Values are in thousands of dollars.

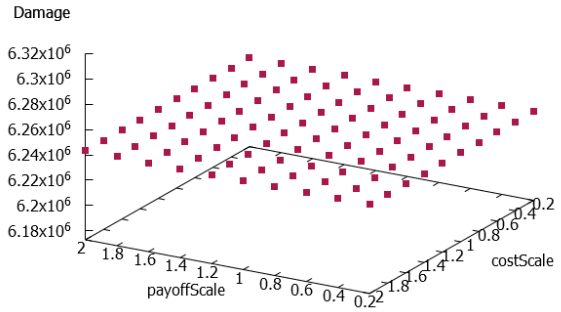

Figure 5.14: Monetary damage done by the nation-state adversary and no IDS.

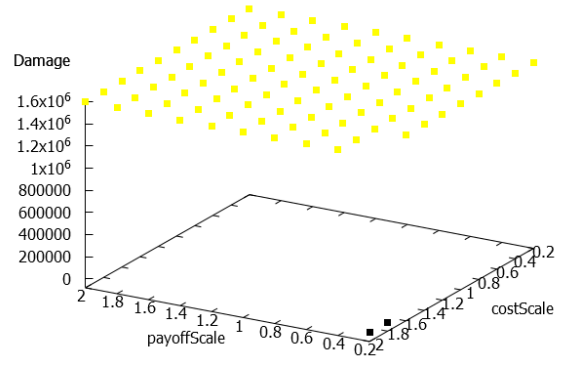

Figure 5.15: Monetary damage done by the nation-state adversary with *ScaleFactorIDS*={0.001,0.01,0.1} and centralized IDS.

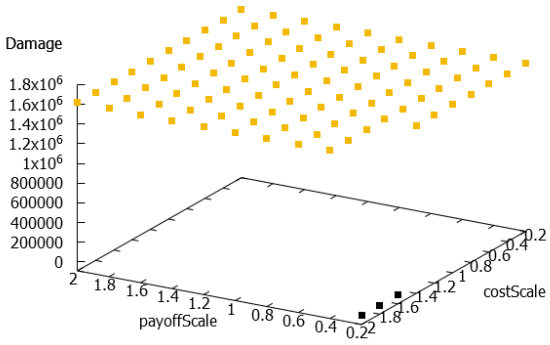

Figure 5.16: Monetary damage done by the nation-state with *ScaleFactorIDS*=0.2 and centralized IDS.

Tables 5.13 and 5.14 show that the behavior of this adversary is very sensitive to the cost of the *Physical* attack and to the payoff of the *Damage Equipment* goal, but is very insensitive to the other payoff, cost, and detection values. The data shows us that if the cost of the *Physical* attack is increased or the payoff of the *Damage Equipment* goal is decreased the adversary does no damage to the system. The terrorist does no harm to the system in this case because it does nothing, since all the other actions have a negative expected payoff.

Figures 5.19 and 5.20 show that the terrorist is not very sensitive to the type or strength of the intrusion detection system, but is quite sensitive to the aggregated costs and payoffs relating to the attack. This result is not surprising, since the intrusion detection system does

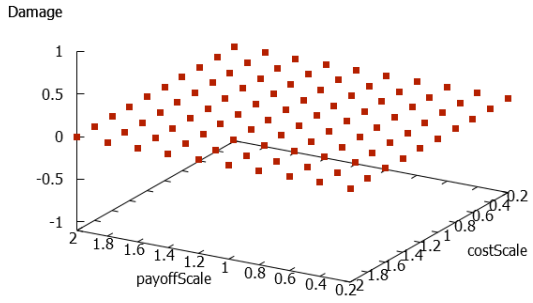

Figure 5.17: Monetary damage done by the nation-state adversary with *ScaleFactorIDS*={0.001,0.01,0.1} and dedicated IDS.
Also represents damage done by the nation-state adversary with *ScaleFactorIDS*={0.001,0.01,0.1,0.2 } and embedded IDS.

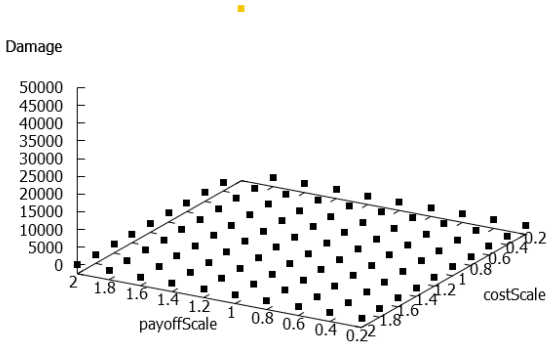

Figure 5.18: Monetary damage done by the nation-state adversary with *ScaleFactorIDS*=0.2 and dedicated IDS.

not change the efficacy of the physical attack, which the terrorist favors.

The utility may use the results of this sensitivity analysis to help make security investment decisions. For example, the sensitivity analysis confirms that the different kinds of intrusion detection systems have a low impact on the terrorist. If the utility is concerned with terrorist attacks it would be reasonable for it to invest more money into things that would make physical attack more difficult (such as fences, security cameras, physical patrols) rather than investing in a powerful and expensive intrusion detection system. The sensitivity analysis also shows that the degree to which the terrorist values the *Damage Equipment* goal has a large effect on the model results, from the perspective of the defender. If the value of this goal decreases slightly it will demotivate the terrorist so much that it will not even attempt the attack, while a large increase in the value of the goal will barely effect the results. This gives the modeler some confidence in the estimate for this goal value: if the estimate is just slightly too high it is likely that the adversary won't even try the attack, but if the estimate if low (even grossly low) the adversary likely won't do much more damage than they would if the estimate were true.

Variable	IDS Type	**0.2**	**0.5**	**1.0**	**1.5**	**2.0**	**5.0**
costScaleFactor 1	None	1051	1019	1019	1019	1019	1019
	Centralized	1051	1019	1019	1019	1019	1019
	Dedicated	1051	1019	1019	1019	1019	1019
	Embedded	1051	1019	1019	1019	1019	1019
costScaleFactors 3,4,5,6,8,9,15, and 16	None	1019	1019	1019	1019	1019	1019
	Centralized	1019	1019	1019	1019	1019	1019
	Dedicated	1019	1019	1019	1019	1019	1019
	Embedded	1019	1019	1019	1019	1019	1019
costScaleFactor 13	None	1019	1019	1019	0	0	0
	Centralized	1019	1019	1019	0	0	0
	Dedicated	1019	1019	1019	0	0	0
	Embedded	1019	1019	1019	0	0	0
payoffScaleFactors 1,2,3, and 4	None	1019	1019	1019	1019	1019	1019
	Centralized	1019	1019	1019	1019	1019	1019
	Dedicated	1019	1019	1019	1019	1019	1019
	Embedded	1019	1019	1019	1019	1019	1019
payoffScaleFactor 5	None	0	0	1019	1019	1019	1051
	Centralized	0	0	1019	1019	1019	1051
	Dedicated	0	0	1019	1019	1019	1051
	Embedded	0	0	1019	1019	1019	1051

Table 5.13: Cost incurred by the utility company due to a terrorist attack, for each scaling factor value. Values are in thousands of dollars.

Variable	IDS Type	**0.0001**	**0.001**	**0.01**	**0.05**	**0.1**	**0.25**
IDSMultiplier	Centralized	1019	1019	1019	1019	1019	1019
	Dedicated	1019	1019	1019	1019	1019	1019
	Embedded	1019	1019	1019	1019	1019	1019

Table 5.14: Cost incurred by the utility company due to a terrorist attack, for each scaling factor value. Values are in thousands of dollars.

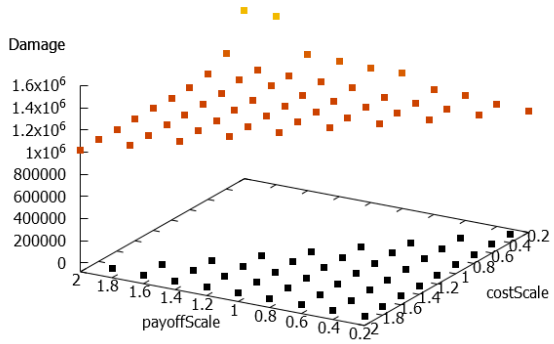

Figure 5.19: Monetary damage done by the terrorist and no IDS.

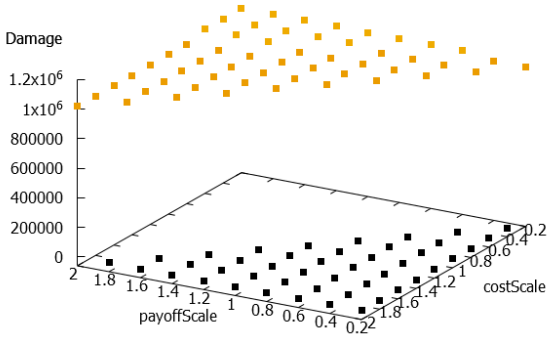

Figure 5.20: Monetary damage done by the terrorist adversary with *ScaleFactorIDS*= {0.001,0.01,0.1,0.2} and IDS = {centralized, dedicated, embedded}.

CHAPTER 6

CONCLUSIONS

In this thesis we demonstrated that the ADVISE state-based stochastic modeling approach may be used to calculate security metrics that are relevant in comparing different IDS deployment configurations in an AMI network through the use of a case study. These security metrics may be used by analysts to build more secure and defensible AMI deployments.

We argue that the scientific approach ADVISE offers for security evaluation is a useful complement to a common method of estimating the relative effectiveness of different security approaches: consultation of one or more security experts, who rely on intuition and experience. In contrast, the metrics calculated by ADVISE are easily auditable by other parties and assumptions are explicitly stated, which allows multiple security experts with different backgrounds to use the ADVISE formalism as a modeling language to collaboratively analyze different system designs.

6.1 An Argument for Quantitative Security Metrics

Some are concerned that models that produces quantitative security metrics (like ADVISE) may be misused to the detriment of the security of the system, in part because people may place too much faith in the accuracy of the metrics [28]. However, we believe that the alternative of having no formal security model is worse. Security analysts will almost always form an informal mental model of the system, which are limited in many of the same ways as a formal security model, while also suffering from its own set of limitations. The ADVISE formalism gives a modeler the ability to convert this mental model into a formal, concrete, rigorous, auditable model. The quantitative metrics produced by these models will contribute towards the development of a science of security.

6.2 Future Work

In the future, we plan to make the ADVISE formalism easier to use, and extend it to increase the modeling power of the approach. First, we will explore an approach to convert a high-level system diagram into an Attack Execution Graph, which will allow a modeler to quickly and easily generate large, sophisticated security models. Then, we shall consider ways to extend the ADVISE formalism so that a modeler may also analyze the defender behavior in the system of interest, in addition to the adversary's behavior, as well as the interactions between the defender and the adversary.

6.2.1 Automatic Generation of AEG

Though many aspects of the ADVISE formalism are intuitive and easy to use, there is room for improvement. It can be challenging, time-consuming, and error-prone to create effective Attack Execution Graphs by hand, even for an expert in security and modeling. Those without security or modeling background may have even greater difficulty learning and using the ADVISE formalism. In the future, we would like to explore an approach for automatically generating Attack Execution Graphs from high level system models.

Using this approach, a modeler would construct a graphical model of the system of interest. The modeler would use elements taken from a preexisting *system component library* to build the system model. The system component library would consist of system components and the relationships that may be formed among the components. For example, a system model may consist of networks, smart meters, intrusion detection systems, and the relationships among them. Once the security analyst constructs the system model, it may be used by a model generation algorithm to create an Attack Execution Graph. The model generation algorithm would utilize a predefined ontology which would map elements in the system component library to elements of an AEG.

To determine whether or not the approach is effective, we intend to create a custom system component library and ontology for the AMI case study, and use a generation algorithm similar to the one proposed in [29] to automatically convert a high-level system model into an Attack Execution Graph. We would then compare the automatically-generated AEG with the hand-generated AEG, to determine their similarity. If successful, this model generation approach could dramatically reduce the time needed to create a functional ADVISE model.

6.2.2 Extending ADVISE

Currently, it is difficult to model the defender's response to an adversary's actions. In this case study, the defender response is modeled implicitly in the Attack Execution Graph through the use of global variables. This is not a good general approach, because it makes it difficult to calculate some defender-specific metrics and specify some defender behaviors. Several other ADVISE case studies [14] [30] use stochastic activity networks (SANs) [31] to model the defender behavior in the system. In these case studies, the defender SAN models and the adversary ADVISE models are composed together to calculate relevant security metrics. This approach is powerful and flexible, but difficult to use. It requires the modeler to learn two formalisms (ADVISE and SANs), and SANs are a very general-purpose modeling language and are not particularly well suited to create security models. We propose extending the ADVISE formalism so it could natively support defender models in addition to adversary models to overcome this limitation. The security models created using this extended formalism would be a very powerful aid to security analysts who seek to make good design choices when building or improving cyber systems.

REFERENCES

[1] M. Rausch, B. Feddersen, K. Keefe, and W. H. Sanders, *A Comparison of Different Intrusion Detection Approaches in an Advanced Metering Infrastructure Network Using ADVISE*. Cham: Springer International Publishing, 2016, pp. 279–294. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-43425-4_19

[2] "Estimating the costs and benefits of the smart grid: A preliminary estimate of the investment requirements and the resultant benefits of a fully functioning smart grid," Electric Power Research Institute, Tech. Rep., March 2011.

[3] L. Alejandro, C. Blair, L. Bloodgood, M. Khan, M. Lawless, D. Meehan, P. Schneider, and K. Tsuji, "Global market for smart electricity meters: Government policies driving strong growth," 2014. [Online]. Available: https://www.usitc.gov/publications/332/id-037smart_meters_final.pdf

[4] Cyber Intelligence Section, "Smart grid electric meters altered to steal electricity," 2010. [Online]. Available: http://krebsonsecurity.com/wp-content/uploads/2012/04/FBI-SmartMeterHack-285x305.png

[5] M. Ward, "Smart meters can be hacked to cut power bills," October 2014. [Online]. Available: http://www.bbc.com/news/technology-29643276

[6] V. Badrinath Krishna, G. A. Weaver, and W. H. Sanders, "PCA-based method for detecting integrity attacks on advanced metering infrastructure," in *International Conference on Quantitative Evaluation of Systems*. Springer International Publishing, 2015, pp. 70–85.

[7] V. Badrinath Krishna, R. K. Iyer, and W. H. Sanders, "Arima-based modeling and validation of consumption readings in power grids," in *Critical Information Infrastructures Security*. Springer International Publishing, 2016, pp. 199–210.

[8] V. Badrinath Krishna, K. Lee, G. A. Weaver, R. K. Iyer, and W. H. Sanders, "F-deta: A framework for detecting electricity theft attacks in smart grids," *46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2016*.

[9] D. Grochocki, J. H. Huh, R. Berthier, R. Bobba, W. H. Sanders, A. A. Cardenas, and J. G. Jetcheva, "AMI threats, intrusion detection requirements and deployment recommendations," in *Smart Grid Communications (SmartGridComm), 2012 IEEE Third International Conference on*, Nov 2012, pp. 395–400.

[10] J. Thornton, "Transformed," *Mechanical Engineering*, vol. 137, no. 3, p. 36, 2015.

[11] Lloyd's, "Business blackout," *Lloyd's Emerging Risk Report*, May 2015.

[12] A. A. Cardenas, R. Berthier, R. B. Bobba, J. H. Huh, J. G. Jetcheva, D. Grochocki, and W. H. Sanders, "A framework for evaluating intrusion detection architectures in advanced metering infrastructures," *Smart Grid, IEEE Transactions on*, vol. 5, no. 2, pp. 906–915, March 2014.

[13] S. Stolfo, S. M. Bellovin, and D. Evans, "Measuring security," *IEEE Security & Privacy*, vol. 9, no. 3, pp. 60 – 65, 2011.

[14] E. LeMay, "Adversary-driven state-based system security evaluation," Ph.D. dissertation, University of Illinois at Urbana-Champaign, Urbana, Illinois, 2011.

[15] W. Wang and Z. Lu, "Survey cyber security in the smart grid: Survey and challenges," *Comput. Netw.*, vol. 57, no. 5, pp. 1344–1371, Apr. 2013. [Online]. Available: http://dx.doi.org/10.1016/j.comnet.2012.12.017

[16] B. Khoo and Y. Cheng, "Using RFID for anti-theft in a Chinese electrical supply company: A cost-benefit analysis," in *Wireless Telecommunications Symposium (WTS), 2011*, April 2011, pp. 1–6.

[17] S. McLaughlin, D. Podkuiko, S. Miadzvezhanka, A. Delozier, and P. McDaniel, "Multi-vendor penetration testing in the advanced metering infrastructure," in *Proceedings of the 26th Annual Computer Security Applications Conference*, ser. ACSAC '10. New York, NY, USA: ACM, 2010. [Online]. Available: http://doi.acm.org/10.1145/1920261.1920277 pp. 107–116.

[18] Möbius team, *Möbius Documentation*, University of Illinois at Urbana-Champaign, Urbana, IL, 2014. [Online]. Available: https://www.mobius.illinois.edu/wiki/

[19] United States Department of Energy, "Operations and maintenance savings from advanced metering infrastructure - initial results," 2012. [Online]. Available: "http://energy.gov/sites/prod/files/AMI_Savings_Dec2012Final.pdf"

[20] M. Anas, N. Javaid, A. Mahmood, S. Raza, U. Qasim, and Z. A. Khan, "Minimizing electricity theft using smart meters in AMI," in *P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2012 Seventh International Conference on.* IEEE, 2012, pp. 176–182.

[21] J. Seo, J. Jin, J. Y. Kim, and J.-J. Lee, "Automated residential demand response based on advanced metering infrastructure network," *International Journal of Distributed Sensor Networks*, vol. 2016, pp. 1:1–1:1, Jan. 2016. [Online]. Available: http://dx.doi.org/10.1155/2016/4234806

[22] D. R. Grochocki, "Deployment considerations for intrusion detection systems in advanced metering infrastructure," M.S. thesis, University of Illinois at Urbana-Champaign, Urbana, Illinois, 2013.

[23] A. Robinson, P. Blythe, M. Bell, Y. Hbner, and G. Hill, "Analysis of electric vehicle driver recharging demand profiles and subsequent impacts on the carbon content of electric vehicle trips," *Energy Policy*, vol. 61, pp. 337 – 348, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0301421513004266

[24] E. LeMay, M. D. Ford, K. Keefe, W. H. Sanders, and C. Muehrcke, "Model-based security metrics using ADversary VIew Security Evaluation (ADVISE)," in *Proceedings of the 8th International Conference on Quantitative Evaluation of SysTems (QEST 2011)*, Aachen, Germany, Sept. 5–8, 2011, pp. 191–200.

[25] C. Salter, O. S. Saydjari, B. Schneier, and J. Wallner, "Toward a secure system engineering methodolgy," in *Proceedings of the 1998 Workshop on New Security Paradigms*, ser. NSPW '98. New York, NY, USA: ACM, 1998. [Online]. Available: http://doi.acm.org/10.1145/310889.310900 pp. 2–10.

[26] R. Bellman, *Dynamic Programming*, 1st ed. Princeton, NJ, USA: Princeton University Press, 1957.

[27] W. H. Sanders and J. F. Meyer, "A unified approach for specifying measures of performance, dependability, and performability," in *Dependable Computing for Critical Applications, Vol. 4 of Dependable Computing and Fault-Tolerant Systems*, A. Avizienis, H. Kopetz, and J. Laprie, Eds. Springer-Verlag, 1991, pp. 215–237.

[28] V. Verendel, "Quantified security is a weak hypothesis: A critical survey of results and assumptions," in *Proceedings of the 2009 Workshop on New Security Paradigms Workshop*, ser. NSPW '09. New York, NY, USA: ACM, 2009. [Online]. Available: http://doi.acm.org/10.1145/1719030.1719036 pp. 37–50.

[29] M. G. Ivanova, C. W. Probst, R. R. Hansen, and F. Kammüller, *Transforming Graphical System Models to Graphical Attack Models*. Cham: Springer International Publishing, 2016, pp. 82–96. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-29968-6_6

[30] R. Wright, K. Keefe, B. Feddersen, and W. H. Sanders, "A case study assessing the effects of cyber attacks on a river zonal dispatcher," in *Proceedings of the 11th International Conference on Critical Information Infrastructures Security (CRITIS)*, 2016.

[31] W. H. Sanders and J. F. Meyer, "Stochastic activity networks: Formal definitions and concepts," in *Lectures on Formal Methods and Performance Analysis*, ser. Lecture Notes in Computer Science, E. Brinksma, H. Hermanns, and J. P. Katoen, Eds., vol. 2090. Berg en Dal, The Netherlands: Springer, 2001, pp. 315–343.