SANDIA REPORT
SAND2006-6399
Unlimited Release
Printed October 2006

Critical Infrastructure Systems of Systems Assessment Methodology

Jennifer DePoy, James Phelan, Peter Sholander, Bryan J. Smith, G. Bruce Varnado, Gregory D. Wyss, John Darby, and Andrew Walter

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd.
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2006-6399
Unlimited Release
Printed October 2006

Critical Infrastructure Systems of Systems Assessment Methodology

Jennifer DePoy
Critical Infrastructures Systems
James Phelan
Contraband Detection
Peter Sholander
Networked Systems Survivability and Assurance
Bryan J. Smith
IO Red Team and Assessments
G. Bruce Varnado
Security Risk Assessment
Gregory D. Wyss, John Darby, and Andrew Walter
Security Systems Analysis

Sandia National Laboratories
Albuquerque, New Mexico 87185

Abstract

Assessing the risk of malevolent attacks against large-scale critical infrastructures requires modifications to existing methodologies that separately consider physical security and cyber security. This research has developed a risk assessment methodology that explicitly accounts for both physical and cyber security, while preserving the traditional security paradigm of detect, delay, and respond. This methodology also accounts for the condition that a facility may be able to recover from or mitigate the impact of a successful attack before serious consequences occur. The methodology uses evidence-based techniques (which are a generalization of probability theory) to evaluate the security posture of the cyber protection systems. Cyber threats are compared against cyber security posture using a category-based approach nested within a path-based analysis to determine the most vulnerable cyber attack path. The methodology summarizes the impact of a blended cyber/physical adversary attack in a conditional risk estimate where the consequence term is scaled by a "willingness to pay" avoidance approach.

This page intentionally left blank.

Contents

Figures

Tables

Executive Summary

Protecting critical infrastructure facilities against malevolent attacks is a major challenge for facility operators. Traditional security threats from vandals seeking to deface property or cause inconsequential damage have been managed by basic security principles, such as perimeter protection and periodic surveillance. However, an emerging adversary with a philosophical intent to destroy American society may consider more sophisticated attacks and cause widespread damage to critical infrastructures. As the critical infrastructure business practices leverage more system automation, security assessment technology must also be able to evaluate the relationship between cyber and physical security and its implications for unidentified vulnerabilities.

Most of the critical infrastructures deliver a commodity such as water, power, or natural gas to an end user. An infrastructure facility comprises "assets," such as systems, subsystems, or components that must operate properly in order for the facility to perform its intended function. The facilities include commodity delivery assets (e.g., pumps, valves, transformers, etc.) and cyber elements that can control set-points, actuation, or other operating functions for the commodity-delivery assets.

The facilities will have some form of physical protection (e.g., fences, locks, alarm systems, etc.) and some form of cyber protection (e.g., firewalls, administrative access controls, etc.). Some of the physical protection system (PPS) elements may be controlled or monitored by cyber means. Interactions between physical and cyber security are recognized in the popular media (such as the *Oceans 11* movie) and the security community. However, most risk and vulnerability assessment research has focused on either physical or cyber security. This effort first explored various historic approaches to either physical or cyber security assessment methods. Upon finding no satisfactory existing method that considered both physical and cyber attacks, a new approach was developed to evaluate "blended attacks" where the adversary makes use of both physical and cyber attack tactics.

The purpose of this LDRD project was to develop a risk assessment methodology that supports analysis of integrated physical and cyber security elements within critical infrastructure systems. The most important outcomes of this work were to achieve a better understanding of these cyber/physical interfaces and their implications for unidentified vulnerabilities and to provide decision makers with integrated and comprehensive risk results for "blended" security systems that can contain both cyber and physical elements.

Through this LDRD project, the physical security and cyber security team members researched historical approaches and retained valuable aspects of past methods, and then added new elements to develop a truly integrated cyber/physical security assessment methodology. Within this report, we have attempted to communicate the most important outcomes of this work, which were

- to achieve a better understanding of the cyber/physical interfaces and implications for unidentified vulnerabilities, and
- to provide a tool for decision makers that shows integrated and comprehensive risk results for "blended" security systems that can contain both cyber and physical elements.

The project team recognized that not all security systems are of similar sophistication, nor should they be. Security systems for low consequence impacts or where mitigation might provide adequate risk management could be evaluated with a best practices or screening analysis method. A best practices questionnaire analysis tool (CICSTART) was developed to evaluate both physical and cyber security practices. Conversely, high consequence impacts or difficult to mitigate risks often have more sophisticated security systems, which require functional or engagement style security systems analysis. This project created a functional style security assessment method (CPSAM) that integrates cyber and physical security systems as a software application.

The CPSAM functional risk assessment methodology combines the fundamentals of physical protection systems (e.g., detect, delay and respond) with cyber protection system primitives that are based on opportunistic pathway analysis. The methodology begins with a fundamental risk principle where the analyst selects specific consequences of concern (CoC) so resources are not wasted looking at inconsequential impacts. Specific key asset failures that could lead to those consequences are identified by external analysis methods as these are often tailored to the complexities of the specific infrastructure. The capabilities of the adversary attacking the facility are contrasted with the protective features at the facility to estimate the likelihood of adversary success. The key cyber-physical security integration step occurs in the portion of the vulnerability assessment model, where the performance of protection elements that are cyber-controlled are turned off to account for the likelihood that an attacker could penetrate the cyber protection system. Since there is insufficient data to support a probabilistic approach to cyber security assessment, a novel application of a broader mathematical tool (Belief and Plausibility) was developed to support vulnerability estimates for attacks that include cyber elements.

While the CPSAM is an operational alpha-version software product, additional developments of the methodology and software features were identified. These include: 1) development of user interfaces to elicit data, 2) development of graphics to display differences in risk values, 3) development of multiple target applications, 4) links to engineering process models to automate target set identification, 5) improved methods to assess mitigation, and 6) improved techniques to evaluate cyber protective system effectiveness.

Acronyms

ASME	American Society of Mechanical Engineers
BATLE	Brief Adversary Threat Loss Estimator
BIT	Built-In Test
BMS	Balanced Magnetic Switch
C	Consequence
CAP	Cyber Access Point
CARA	Critical Asset Risk Assessment
CD	Compact Disk
CISOSAM	Critical Infrastructure System of Systems Assessment Methodology
CoC	Consequence of Concern
CPEI	Cyber Protection Effectiveness Index
CPS	Cyber Protection System
CPSAM	Cyber Physical Security Analysis Methodology
CR	Conditional Risk
CSP	Cyber Security Posture
DoD	Department of Defense
DOE	Department of Energy
EASI	Estimate of Adversary Sequence Interruption
EPM	Engineering Process Model
ETA	Event Tree Analysis
FMEA	Failure Modes and Effects Analysis
FMECA	Failure Modes, Effects, and Criticality Analysis
FTA	Fault Tree Analysis
GUI	Graphical User Interface
H	High
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
ISP	Internet Service Provider
L	Low
LAN	Local Area Network
LDRD	Laboratory Directed Research and Development
LMA	Logging/Monitoring/Auditing
M	Medium
MLE	Markov Latent Effects
NAC	Network Access Control
NRIC	Network Reliability and Interoperability Council

NSA	National Security Agency
NTP	Network Time Protocol
OBEST	Object-Based Event Scenario Tree
OS	Operating System
P_I	Probability of Interruption
PMD	Protective Measures Definer
P_N	probability of neutralization
POST	perform power-on self test
PPS	physical protection system
PRA	probabilistic risk assessment
$P_{S\|A}$	Probability of Success given an Attack
PSTN	Public Switched Telephone Network
R	Risk
R&D	research and development
RAM-W	Risk Assessment Methodology for Water Utilities
RBAC	roles-based access control
RPC	Remote Procedure Calls
SCADA	Supervisory Control and Data Acquisition
SNAC	Systems and Network Attack Center
SNM	Special Nuclear Material
SP	Security Primitive
SSL	Secure Sockets Layer
TD	Threat Definer
UAC	User Access Control
VA	Vulnerability Assessment or Vulnerability Analyst
VPN	virtual private network
VSAT	Vulnerability Self Assessment Tool
WEP	Wired Equivalent Privacy

1. Introduction

1.1 Overview

Protecting critical infrastructure facilities against malevolent attacks is a major challenge for facility operators. Traditional security threats from vandals seeking to deface property or cause inconsequential damage have been managed by basic security principles, such as perimeter protection and periodic surveillance. However, an emerging adversary with a philosophical intent to destroy American society may consider more sophisticated attacks to cause widespread damage to critical infrastructures. As the critical infrastructure business practices leverage more system automation, security assessment technology must also be able to evaluate the relationship between cyber and physical security and its implications for unidentified vulnerabilities.

Most of the critical infrastructures deliver a commodity such as water, power, or natural gas to an end user. An infrastructure facility comprises "assets," such as systems, subsystems, or components that must operate properly in order for the facility to perform its intended function. The facilities include commodity-delivery assets (e.g., pumps, valves, transformers, etc.) and cyber elements that can control set-points, actuation, or other operating functions for the commodity-delivery assets.

The facilities will have some form of physical protection (e.g., fences, locks, alarm systems, etc.) and some form of cyber protection (e.g., firewalls, administrative access controls, etc.). Some of the physical protection system (PPS) elements may be controlled or monitored by cyber means. Interactions between physical and cyber security are recognized in the popular media (such as the *Oceans 11* movie) and the security community. However, most risk and vulnerability assessment research has focused on either physical or cyber security. This effort first explored various historical approaches to either physical or cyber security assessment methods. Upon finding no satisfactory existing method that considered both physical and cyber attacks, a new approach was developed to evaluate "blended attacks" where the adversary makes use of both physical and cyber attack tactics.

1.2 Purpose

The purpose of this LDRD project was to develop a risk assessment methodology that supports analysis of integrated physical and cyber security elements within critical infrastructure systems.

The most important outcomes of this work were to achieve a better understanding of these cyber/physical interfaces and their implications for unidentified vulnerabilities and to provide decision makers with integrated and comprehensive risk results for "blended" security systems that can contain both cyber and physical elements.

1.3 Approach

Physical and cyber security have fundamentally distinct foundations, which challenged the team first in understanding each domain, and then in finding a general construct where both physical and cyber security assessment methods could be functionally integrated. Physical security

assessment is founded on a time sequence race (called "detect, delay, and respond") for the adversary to defeat security systems (e.g., fences, locks, etc.) and overcome a response force. Physical protection systems include perimeter defense, active detection technology, and access controls to allow privileged entry. Cyber protection systems also include "perimeter" defense such as firewalls and access controls. However, they typically do not rely on active detection technology to summon a response force because once a perimeter is penetrated, an attack typically proceeds faster than a cyber response force can act. Cyber security assessment methods are typically either checklist-based to compare current capabilities with best practices or use a red team approach that actively engages the cyber security system with custom-designed exploits.

To address this problem, this research developed a functional risk assessment methodology that combined the fundamentals of physical protection systems (e.g., detect, delay, and respond) with cyber protection system primitives that are based on an opportunistic pathway analysis [Young et al. 2004]. The methodology begins with a fundamental risk principle where the analyst selects specific consequences of concern (CoC) so resources are not wasted by evaluating inconsequential impacts. Specific key asset failures that could lead to those consequences are identified by external analysis methods as these are often tailored to the complexities of the specific infrastructure. The capabilities of the adversary attacking the facility are contrasted with the protective features at the facility to estimate the likelihood of adversary success. The key cyber-physical security integration step occurs in the vulnerability assessment portion of the model, where the performance of protection elements that are cyber-controlled are turned off to account for the likelihood that an attacker could penetrate the cyber protection system. Since there are insufficient data to support a probabilistic approach to cyber security assessment, a novel application of a broader mathematical tool (Belief and Plausibility) was developed to support vulnerability estimates for attacks that include cyber elements. Appendix A provides a short tutorial on the theory of evidence that underlies this assessment approach.

1.3.1 Attack Types

Attacks may be physical or cyber, or some combination of the two. This research considered four types of attacks against critical infrastructure assets:

- physical-only
- cyber-enabled physical
- cyber-only
- physically enabled cyber

Each of these attack types is defined below and in Figure 1.

Physical-only attacks — The attacker gains physical access to the asset under attack in order to damage or disable it. In this case, asset failure is induced by actions taken at the asset location. The two general types of physical attacks are physical-only attacks and cyber-enabled physical attacks. Physical-only attacks involve no cyber activities.

Cyber-enabled physical attacks — The attacker uses cyber means to facilitate physical access to the asset. In a cyber-enabled physical attack, the attacker disables cyber-controlled elements of

the PPS (for example, magnetic locks or detection systems) that can be turned off by cyber control. Cyber-enabled physical attacks can occur if (and only if) one or more elements of the PPS are cyber controlled.

Cyber-only attacks — The attacker induces failure of the commodity delivery asset without gaining physical access to that asset. Cyber attacks are of concern only if: 1) the physical asset is cyber controlled; and 2) failure of the asset can be caused by cyber manipulation. In this context, the failure of the commodity delivery asset means it is damaged, disabled, or manipulated in a way that leads to a "Consequence of Concern" (CoC). The fact that a commodity delivery asset is cyber-controlled does not necessarily mean that it can be caused to "fail" (e.g., cause a physical action that leads to a CoC) by cyber means. A cyber-only attack is launched without gaining physical access to the facility at which the commodity delivery asset is located or controlled. For example, an attack may be launched from a "Cyber Access Point" (CAP) that is attached to the public Internet. In that case, the system owner has no control over the physical protection at the CAP.

Physically enabled cyber attacks — The attacker gains physical access to an on-site location from which the cyber attack is then launched. This type of attack involves a physical attack to gain access to an on-site location. A cyber attack is then launched from that physical location, which is under the control of the facility operator. The difference from the cyber-only case is that the attacker must first circumvent the physical protection for that CAP before the attacker can then use that physical location's cyber elements (e.g., network terminal or control system element, for example) to launch a cyber attack. A simple example would be a malicious insider who has access to the industrial process control room. A more complex example would be a terrorist team comprised of several large, well-trained paramilitary forces and one cyber specialist assaulting a physically protected control room.

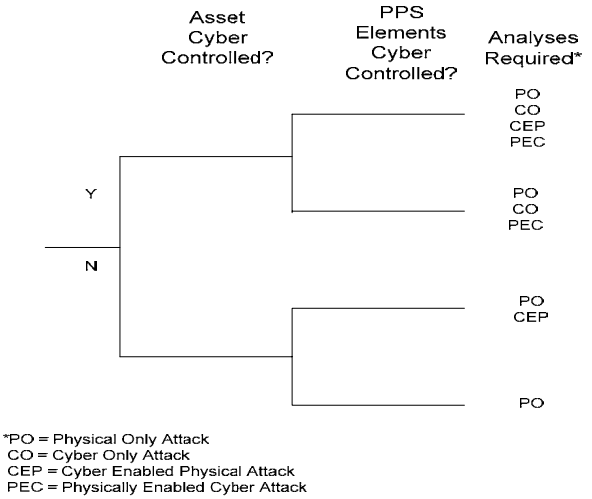

Figure 1. Functional Breakdown of Singular and Blended Physical and Cyber Attacks

2. Assessment Method Review

2.1 Existing Methods

In the initial stages of this effort, the project team recognized that many risk assessment methods had been developed. These methodologies provided certain insights into the complex phenomena of risk analysis, which the team sought to understand and then determine their potential for analysis of a blended cyber and physical attack. The following sections provide a brief description of select methods categorized into inductive methods, deductive methods, and other types of logical models.

2.1.1 Inductive Methods

Inductive risk assessment methods start with the definition of potential risk scenarios followed by identification of risks or consequences that might occur as a result of that scenario. The risk scenarios are identified through both formalized methodologies and imaginative thinking, and are based on component failures, subsystem failures, human actions, and/or natural and human-made phenomena. While there are many inductive methods, this section focuses on two classes: failure modes and effects analysis (FMEA) and event tree analysis.

The most important limitations of inductive techniques are related to a reliance on "bottom-up" problem-solving method – that is, the sources of risk are identified at the *beginning* of the analysis, instead of being inferred by a systematic deductive "top-down" approach. If the analyst does not envision a particular scenario, then that scenario will remain unanalyzed because a list of scenarios is the starting point for an inductive analysis, *not* its result.

2.1.1.1 Failure Modes and Effects Analysis

The FMEA technique [McCormick 1981), along with its close cousins, failure modes, effects and criticality analysis (FMECA), and HAZOP [Greenberg and Cramer 1991), are generally the first systematic risk and reliability analysis techniques applied to a system. The purpose of an FMEA is to examine individual components and assess the effect of their failure on the system in which they are used (and on other systems and subsystems). FMEA is a qualitative method that is typically documented in a tabular format. To accomplish an FMEA, the analyst examines the components of a system one by one, and for each component, considers every known failure mode individually. The analyst writes a description of the failure mode, the method by which that failure would be detected, the effect of the failure, and the expected response of operators or automatic controls to the situation.

A HAZOP study is related to an FMECA in that it assesses predefined scenarios to determine their probable causes, consequences, and possible remediation actions. The HAZOP method focuses on qualitative deviations of key system operating parameters from their nominal, normal, or design values. The fundamental philosophy here is that normal operations are generally safe, and deviations from these normal operations are the source of unexpected or unrecognized problems. The objective is to find the "weak link" in the system, and to provide a basis for developing procedural or engineering controls to reduce any risks so identified. The one-by-one

nature of parameter variation in a HAZOP study and failure consideration in an FMEA can neglect the effects of multiple concurrent failures or variations, which may have both significant likelihood and high criticality.

2.1.1.2 Event Tree Analysis

Event tree analysis (ETA) is an inductive risk assessment technique that represents an undesired occurrence as a sequence of events [Cramond 1985]. Event trees are similar in form to decision trees and are used to represent the spectrum of possible outcomes given a particular initial condition. The method is inductive in that it begins with a particular set of initial conditions and uses inductive logic rather than deductive logic to infer its results.

The events within an event tree may include the status of physical systems, operator actions, the activities of automated control systems, and random (stochastic) events both internal and external to the system. The events may or may not be independent of one another, but if they are not independent, then the interdependencies are explicitly included within the logical structure of the tree. A path through the event tree is constructed by selecting a unique outcome for each event within the event tree model. Thus, the path physically represents a unique sequence of events so that outcome O_{1P} occurs for event 1, *and* outcome O_{2P} occurs for event 2, *and* outcome O_{3P} occurs for event 3, and so forth.

If the event tree model is properly constructed, the set of all paths through the model represents the complete set of possible outcomes that can occur as a result of the given initial condition (but typically only the outcomes relevant to the analyst's needs are modeled). The results of an event tree analysis are initially qualitative descriptions of individual scenarios. If, however, one assigns conditional probabilities to the individual event outcomes, then one can also obtain quantitative results consisting of the scenario (path) definition and its probability of occurrence.

In contrast, the principal limitation of the event tree method is that an event tree is by definition an acyclic graph. Because cycles are prohibited, it can be difficult to represent the behavior of systems that embody feedback loops in an event tree model.

An event-tree-based analysis tool developed the American Society of Mechanical Engineers is the Risk Analysis and Management for Critical Asset Protection (RAMCAP) [ASME 2004] developed to provide a homeland security risk analysis and risk management decision-making tool for government and private industry. The Critical Asset Risk Analysis (CARA) methodology includes a screening analysis of assets that precedes the detailed analysis modeled after probabilistic risk assessment (PRA) methods. The challenge to using PRA methods in counter-terrorism analysis is that the available data are sparse for probability of attack, probability of failure given an attack, and probability of consequence occurrence given a failure. The probabilistic mathematics can provide quantitative estimates for these variables, but these estimates often over-represent the degree of knowledge and confidence in the results than is warranted.

2.1.2 Deductive Assessment Methods

While inductive risk assessment methods start with the definition of potential risk scenarios, deductive risk assessment methods start with the identification of consequences that are possible for a specific system. Deductive reasoning is then used to identify whether there are mechanisms by which those consequences can be achieved, and if so, to identify the scenarios or combinations of events that can cause those consequences to be realized. Deductive methods are by nature systematic "top-down" methods, which, if exercised in a disciplined manner, will coach the analyst into deductively identifying the complete list of possible causes for the analyzed consequence.

2.1.2.1 Fault Tree Analysis

Fault tree analysis (FTA) [Roberts et al. 1981] is the most common deductive logic-based risk assessment technique. FTA uses deductive reasoning, as expressed by logic diagrams, to determine how a particular undesired event can occur. The purpose of the logic diagram is to illustrate the individual steps in the deductive reasoning process so that others can understand not only the results (*how* and *why* things fail), but also the method by which those results were obtained (why *these* elements contribute to system failure). The logic diagram is constructed using the method of *immediate cause* in which one finds the immediate, necessary, and sufficient conditions for each deductive logical step to be satisfied. This method, also known as the "rule of small steps," helps ensure the logical completeness of the fault tree model by ensuring the completeness of the logic at each small step. The premise is that by being logically complete at each small logical step, and allowing the overall logic of the fault tree model to dictate the assembly of these individual logical statements, one has some confidence in the completeness of the overall logical model.

Once the fault tree logic diagram is constructed, it is generally solved to find the *minimal cut sets*. Each minimal cut set represents one set of necessary and sufficient conditions for the occurrence of the undesired event that the fault tree was constructed to investigate (system failure, for example). It is, in essence, a definition of one scenario that results in system failure. The overall group of minimal cut sets then represents the universe of possible scenarios that will lead to this undesired event (subject to the limited scope of the analysis). These *qualitative* results can then be used to provide *quantitative* insights because, when a probability of occurrence is associated with each basic event in each cut set, one can determine an overall probability for each cut set scenario and rank them accordingly. Furthermore, one can dissect the cut set results using simple mathematical manipulations to determine the importance of individual basic events to the overall risk performance of the system.

While FTA is a very structured and systematic way to assess a single system; it can also accurately represent the interactions among multiple systems. It is not unusual to model complex interactions among systems using event trees and let the causes for each event tree event be defined using FTA. In addition, FTA is one of the few techniques that adequately treat common mode and common cause failures. FTA also allows the analyst to consider the effects of human operators and automatic control systems on these individual failure scenarios through the application of "recovery events" to cut sets on a case-by-case basis.

The principal drawback to FTA is that it often represents time-dependent scenarios poorly. In addition, because a fault tree is an acyclic graph, it can be difficult to represent the behavior of systems that contain feedback loops or other circular dependencies in a fault tree model.

One variant of the FTA is Logic Evolved Decision (LED) analysis that is based on linked logic models, where each logic model is a directed graph called a process tree [Eisenhawer et al. 2003]. Two process trees are used for the decision analysis: a possibility tree that represents alternative attack scenarios, and an inference tree that defines how risk will be used to rank order the possibility tree scenarios. The risk evaluation uses approximate reasoning (fuzzy sets) for comparison of qualitative factors.

2.1.3 Other Types of Logical Models

2.1.3.1 Influence Diagrams

An influence diagram is an acyclic probabilistic network that consists of nodes and arcs [Jae and Apostoklakis 1992]. The nodes can represent system states, decisions, or chance or deterministic occurrences, while the arcs represent the conditional dependencies among these occurrences. The nodes ultimately influence a "value node" that quantifies the consequences for each possible combination of occurrences and system states. Conditional probabilities can be applied within the model nodes to represent the probability that a particular event happens *given* specific conditions in the other nodes to which it is connected (i.e., the states, decisions, or events that *influence* this node). Thus, an influence diagram consists of four distinct parts: the nodes, the influences upon the nodes (the dependencies among the nodes, as represented by the arcs), the conditional dependencies within each node upon other nodes in the model, and the conditional probabilities themselves.

The influence diagram method is conceptually similar to the event tree, decision tree, and fault tree methods described earlier. It can be applied as both an inductive and a deductive modeling tool in that one can begin either with the value node (the objective, as is done with fault tree analysis) or with a suitable initial condition (as is done with event tree and decision tree analysis). One could even begin with some of each and work both inductively and deductively as necessary until the model is complete. In addition, the method is not limited to simple binary events as is FTA. This flexibility makes the influence diagram an important tool to the risk analyst. Recent methods for solving influence diagrams [Jansma et al. 1996], which emphasize the development of "paths" (similar to ETA), have enabled influence diagrams to produce highly valuable risk assessment results.

2.1.3.2 Markov Models

Another type of logical model, the Markov model [McCormick 1981) is a directed graph that captures the concepts of system states and probabilistic transitions between states. To build a Markov model, an analyst examines every relevant configuration of a system – both functional and nonfunctional configurations – and defines them to be *states* of the system. The analyst then defines the probability of transition from each state to every other state (as a function of time and

other factors) to complete the model. State transitions that are precluded for physical reasons are assigned a transition probability of zero.

Markov models provide a natural, direct representation, through the use of cycles, of systems that embody feedback loops, systems whose components are repairable, and systems in which component failures interact. Recall that fault trees and event trees are acyclic graphs and, hence, do not readily accommodate these system characteristics. The two basic forms of Markov models are chains and processes. A Markov chain uses matrix multiplication in discrete time to obtain state transition probabilities. A Markov process uses a set of differential equations over continuous time. Relative to the other techniques discussed, Markov processes require a more sophisticated understanding of mathematics for their solution. In fact, most Markov models of real systems suffer from "state explosion" and hence are difficult to solve, requiring simulation. Complete path or scenario information is not a natural output of a Markov model.

An interesting application of Markov modeling is found in the continuous event tree methodology [Devooght and Smitds 1992a, Smitds and Devooght 1994]. In this method, the branching operations within an event tree model are viewed as state transitions within the framework of a Markov model. This allows the analyst to determine the population of each state (and, hence, of each branch within the event tree model) as a function of time. The method has been extended to a semi-Markov process to allow for the state and branch transition probabilities to vary as a function of the length of time the system has spent in that state [Devooght and Smitds 1992b].

Another approach uses Markov Latent Effects (MLE) to quantify imprecise subjective metrics through possibilistic or fuzzy mathematics, which are then aggregated using weighted sums to rank the credibility of various threat scenarios (Tidwell et al., 2004). The latent effects represent the influence that one decision element has on another. This approach explicitly evaluates the threat potential, recognizing that full probabilistic assessment is not possible due to a lack of experiences to provide probabilistic data sets.

2.1.3.3 Object-Based Methods

Another class of logic-based risk assessment methodologies owes its origins to object-oriented modeling methods developed for computer science. In an object-based risk assessment model [Wyss et al. 1999, Wyss et al. 2001, Wyss et al. 2004], one builds an object model to represent the behavior of the system to be analyzed (including normal and abnormal modes of operation, deterministic, and probabilistic behavior), and then queries this object model to extract risk models that have similar characteristics to those generated using the methodologies described above. If the object model is built using appropriate techniques, these models can be used to automatically extract many inductive risk assessment results from a single object model through various probabilistic simulation techniques. This feature is a great strength of the object modeling technique, because it does not require a human analyst to develop and verify each individual risk assessment model. While the extraction of a deductive risk assessment model from an object model is possible, it is generally more cumbersome to build and check the object models than to build the needed deductive logic models from scratch, so it is rarely done.

2.1.3.4 Expert Judgment Methods

Some aspects of risk assessment process are amenable to assessment by expert judgment. The security analyst may enlist the services of a subject matter expert to answer specific security questions related to the likelihood of attack, the effectiveness of the security system, or the consequences that might occur because of a successful attack. The subject matter expert responds with either a qualitative or a quantitative estimate of the relevant parameters.

Many factors influence the quality of the information that can be elicited from subject matter experts. Examples include the qualifications of the experts consulted and the time and resources available to the expert to gather and explore background information. However, two key aspects of the questions asked have a profound affect on the quality of the expert elicitation results. [Meyer and Booker 1991] First, experts provide far more accurate results when asked to compare among two or more options or situations than when asked to evaluate a single situation or question in a vacuum. Second, experts provide more accurate results to simple questions than to complex questions. In fact, subject matter experts will often decompose complex questions and/or construct their own comparison cases during the analysis process. Documented methods are available to assist in this process, some of which are described in this section.

One method for obtaining information from experts is to ask the experts to place the importance or severity of the various options or scenarios on an arbitrary scale, which may be qualitative (i.e., verbal descriptors) or numerical. For example, consequences can be ranked by placing the possible outcomes and/or consequences on an arbitrary scale that represents a consensus description of how "bad" one outcome is in relation to another. One might describe consequences using words like "minimal," "acceptable," or "catastrophic," or one might use positive numbers that range from zero (nothing bad has happened) to some maximum value which represents the worst thing that can possibly happen in the context of the analysis. Some applications have set this maximum consequence to 1.0 as a matter of convenience (for many years, the DOE nuclear consequence scale ranged from 0.0 to 1.0), (DOE Design Basis Threat, prior to 2001) while others have set it to more arbitrary values (the telecommunications outage index ranges from 0.0 to 333.33). [ATIS 1997a, ATIS 1997b] The actual range selected will depend more upon the planned uses of the resulting values than on the specifics of the risk assessment.

Rankings can be assigned on an arbitrary scale using a number of different methods. If there is consensus regarding the relative severity or importance of various outcomes or issues, this consensus can be represented numerically on an appropriate scale. Frequently the resulting numerical values are rounded in order to avoid giving a false impression regarding the level of precision involved in the consequence specification. Such a scale can also be imposed on analysts as a matter of policy by decision-makers. Yet, even here, the actual values are often arrived at by consensus among the community of decision-makers.

A second method for eliciting expert judgment involves performing pairwise comparisons among the various outcomes or issues of interest. Pairwise comparison methods (e.g., the Vital Issues Process, or VIP) [Engi and Glicken-Turnley 1995, Engi 1997] require those in the stakeholder and/or decision maker community to compare the various outcomes or issues in a pairwise manner to establish a consensus ranking among them. This process can be helpful

when there is not an a priori consensus regarding their relative severity or importance because it provides a logical framework for the discussion and resolution of differences and inconsistencies within the community. The various outcomes or issues can be assigned numerical values either by consensus from the resultant ranking, or, when full consensus does not exist, by obtaining consensus values for a few of them and then grouping or interpolating the remaining outcomes or issues with respect to those pegged values.

A third method for conducting expert judgment evaluations is somewhat more mathematically rigorous. The Analytic Hierarchy Process [Saaty 1988, Saaty 1990] is a multi-objective multi-criterion approach to the decision-making process that uses hierarchical decomposition of a complex consequence relationship into simpler parts and recomposition based on structured expert judgment. [Meyer and Booker 1991] The complex relationship is broken down into a series of independent (to the degree possible) criteria that contribute to the high-level consequence. The relative weights of these criteria are established by pairwise comparison in which numerical values are assigned to verbal descriptors of the relative importance of each factor with respect to every other factor. Examples of the verbal descriptors include "equal importance of both elements", "strong importance of one element over the other", and "extreme importance of one element over the other". Practitioners of the method indicate that pairwise comparison among more than nine elements is very difficult, so more complex relationships should be broken down into simpler relationships and using hierarchical decomposition. The numerical values related to the verbal descriptors are ultimately placed into matrix form, and it is asserted that the principal eigenvector for that matrix represents the relative importance of each element to the overall result, while the eigenvalue represents a measure of the consistency among the verbal descriptors derived from the pairwise comparison. Using this method, one develops what is essentially a hierarchical linear utility function that can be evaluated to obtain consequence values for the various consequence outcomes. All resulting values are between zero and one. The method has seen many applications, but can produce questionable results either when consequences are highly nonlinear or when the contributing criteria in the hierarchical decomposition are not independent.

Regardless of the method selected for eliciting comparisons from experts, the decision-makers and stakeholders need to agree that the resultant values are a fair representation of their beliefs regarding the relative severity or importance of the various outcomes. This is especially important in relation to arbitrary numerical consequence scales because there are no objective consequence values or calculations that can be used to benchmark this consequence scale.

Two expert-judgment-based methods were developed to support analysis of water and wastewater utilities, as follows:

- ***Vulnerability Self-Assessment Tool (VSAT)*** [Rees, D.C. and K. I. Rubin 2003] – The VSAT is a structured risk-based methodology and security planning software that provides qualitative risk estimates for users without formal background in risk assessment. The software uses the risk equation (Risk = Consequence * Vulnerability * Occurrence) and provides qualitative comparators (low, moderate, high, very high) for the user to select in an expert-judgment assignment of failure for a specific threat/asset combination. Consequence is also assigned in comparison tables with up

to five different attributes. Risks are then established in a 2-dimensional matrix of Consequence and Vulnerability using the same qualitative comparators for use in identifying potential for risk reduction measures.

- ***Risk Assessment Methodology*** – Water (RAM-W) [RAM-W, 2001] – The RAM-W is a qualitative risk assessment methodology that uses a comprehensive approach for use by a trained risk analyst using a set of tools that include pairwise comparisons, fault trees, consequence analysis and adversary path analysis. The approach begins with a determination of critical/mission functions, undesirable consequences, and the specific assets that need to be protected. The RAM-W included a cyber security assessment based on relative ranking from a best practices analysis; however, the methodology lacked an explicit link between physical security and cyber security.

2.2 Classification Scheme for Risk Assessment Methods

The risk assessment methods described in section 2.1 were developed principally to either explore the risk of functional engineering failures for complex systems or elicit perceived risk from expert opinion using comparison/contrast. In the context of cyber security, most efforts could be categorized into principally Red Team exercises or best practice checklists. As the team was seeking a method for analysis of combined physical and cyber security risk, a common structure to identify relationships and pattern end usage became necessary. Campbell and Stamp [2004] created the classification scheme shown in Table 1 to meet this need.

Table 1. Classification Scheme for Risk Assessment Methods

Level		Approach		
		Temporal	**Functional**	**Comparative**
Abstract	**Expert**	Engagement	Sequence	Principles
Mid-Level	**Collaborative**	Exercise	Assistant	Best Practice
Concrete	**Owner**	Compliance Testing	Matrix	Audit

The project team recognized that numerous Comparative approaches existed for cyber that could benefit from the addition of physical security. This led to the development of the Screening Tool described in Section 3. The Functional methods such as RAM-W [RAM-W 2001] and the Vulnerability Self Assessment Tool (VSAT) [VSAT; Rees and Rubin 2002] provided a comparative context, but lacked the quantitative basis desired to embed improved rigor in the risk estimates. An alternate approach founded on deductive logic was developed with a focus on system level risk (instead of detailed component level phenomena) to directly embed cyber security within an existing physical security structure. This concept became the basis for the Cyber-Physical Security Assessment Methodology (CPSAM) described in Section 4. The Temporal approaches for physical security (e.g., Joint Conflict and Tactical Simulation, JCATS) and cyber security (e.g., IDART[1]) were considered to be too complex for an initial effort to create an analysis method for blended physical and cyber security.

[1] Information Design Assurance Red Team (IDART™) at Sandia National Laboratories. http://www.sandia.gov/idart.

Figure 2 depicts how a facility's existing security level helps define the type of security assessment tool that is best used to evaluate the security risks at that facility. Existing security levels that are low are best evaluated with Screening and Best Practice Methods because the security systems are not sufficiently sophisticated to warrant Functional or Engagement/Red Team Assessments. However, evaluation of more sophisticated security systems requires greater cost, time, or input detail due to the complexity of these systems.

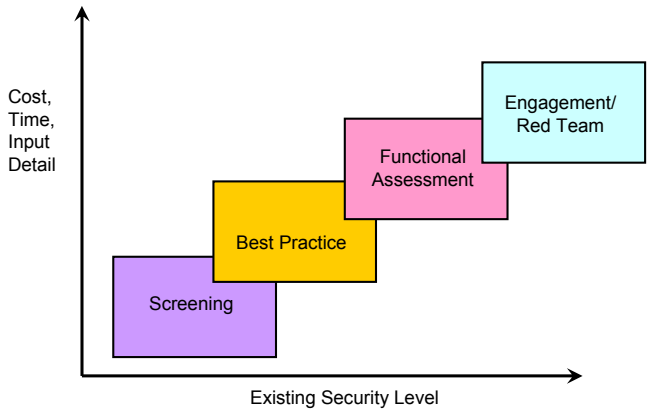

Figure 2. Security Assessment Tool Class Categories

3. Screening Tools

3.1 Sandia Quickstart Baseline Tool

The Sandia Quickstart Baseline Tool (SQBT) is a generic, automated tool based on the COBIT *Quickstart*™ [COBIT Quickstart 2003] from ISACA that was initially developed by this LDRD effort. The *Quickstart*™ tool was improved from a paper-based system to an HTML implementation by the addition of two information channels, scoring capabilities, and an evaluation function [Campbell and Smith 2006]. Figure 3 is a screen shot of the first two control objectives as displayed by the SQBT. The eight scoring columns are shown as radio buttons. The radio buttons enable the user to provide a raw score that indicates the implementation status of the user's organization for that particular control objective.

The "Quickstart Baseline" is an assessment tool based on "control objectives" (described below) within the structure of a maturity model. Like COBIT, the Baseline has a three-tiered, hierarchical structure. The Baseline consists of a set of 62 control objectives, organized into 30 "Processes" which are, in turn, organized into four "Domains". Each control objective is intended to describe some aspect of information assurance.

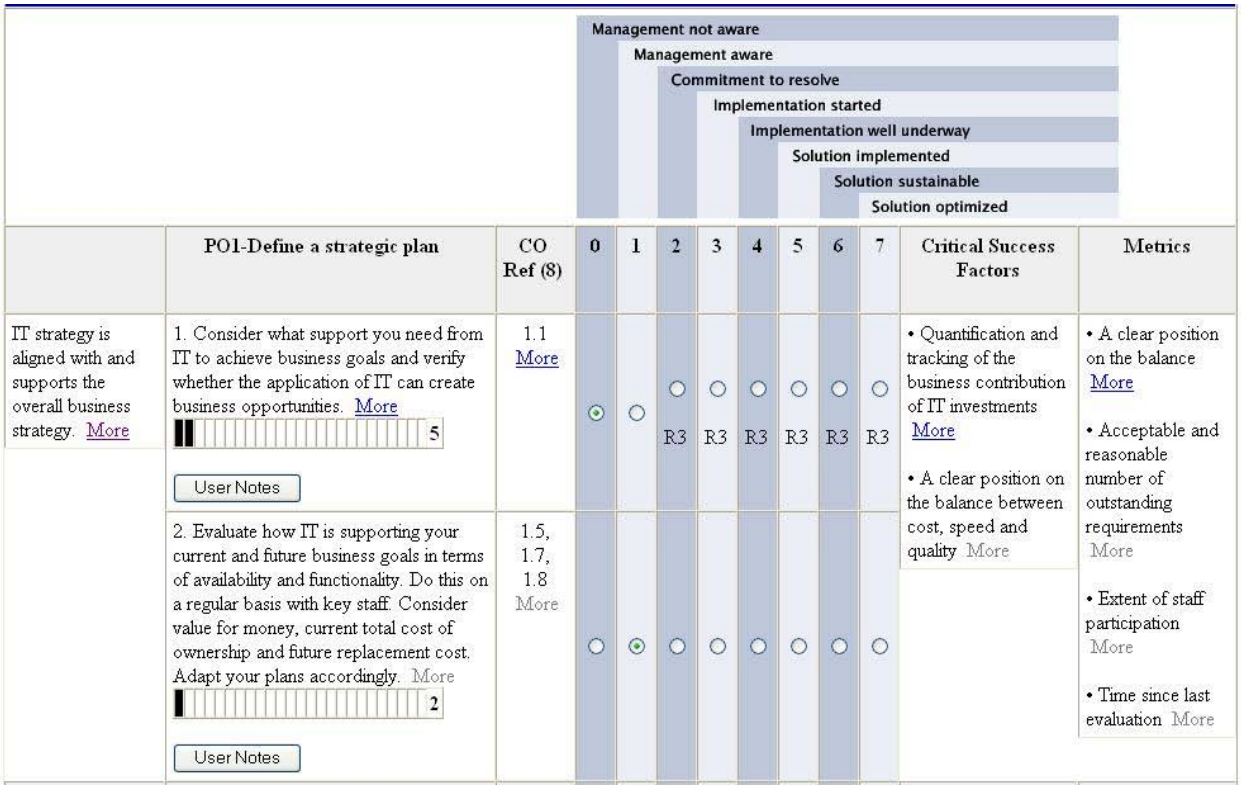

Figure 3. Sandia Quickstart Baseline Tool (two control objective examples)

The SQBT provides a scoring capability that includes weights, ranks, and thresholds. A "Submit" button follows the presentation of the control objectives in the HTML file. When the user clicks on this button, the SQBT multiplies the raw score for each control objective by the weight for that control objective to arrive at a weighted score. The sum of the weighted scores is the cumulative score. If ranks are defined, the SQBT then determines the rank using the cumulative score, as explained below. If thresholds are defined, the SQBT uses thresholds in determination of rank. For each Domain, the SQBT shows the following:

1. the lowest-numbered control objective in the Domain with the lowest raw score;
2. the lowest-numbered control objective in the Domain with the highest raw score;
3. the average raw score for all the control objectives in the Domain; and
4. the standard deviation of the raw scores for all the control objectives in the Domain.

Figure 4 shows part of a sample scoring window. Not shown in Figure 4 are two text boxes, one for the rank and a second for a list of control objectives along with the raw score each would need to be in order for the cumulative score to move to the next higher rank. If ranks are not defined, the headers for both boxes are shown but the boxes are empty.

Figure 4. Sample Scoring Window

3.2 Critical Infrastructure Control System Test, Analysis and Reporting Tool

The Critical Infrastructure Control System Test, Analysis and Reporting Tool (CICSTART) Survey and Questionnaire Tool adaptively poses questions to control systems personnel to assist the self-assessor or inexpert assessor of control system cyber-security. The questionnaire tool adaptively poses survey questions based on tasks that each respondent indicates he or she performs. For example, respondents who indicate they perform the task *install computer systems or platforms* will answer questions that assume familiarity with computer security fundamentals. Other respondents who do not share the responsibility to install computer systems will not answer these same questions. In this way, questions are asked only of those respondents who are likely to be able to answer them. The survey itself consists of questions drawn from a variety of sources, including the following:

- Sandia's past experience in assessing control systems,
- COBIT® Control Practices [IT Governance Institute 2004], and
- SCADA Security Policy Framework™ [Stamp and Kilman 2005].

The survey broadly addresses control systems security from governance and policy to security implementation and physical security of cyber assets. This breadth of coverage helps the self-assessor or inexpert assessor of control systems to more completely identify security problems, whether they are technical vulnerabilities in particular computer platforms or failings in security awareness and training programs. The survey and questionnaire tool even help the experienced assessor to achieve more consistent results and to better direct their discretionary efforts.

3.2.1 The CICSTART Survey

The CICSTART Survey finds its roots in Sandia's past experience assessing control systems in a number of critical infrastructure sectors, especially the water sector. Over time, Sandia assessors developed and maintained a large set of interview questions used to question site personnel. Answers to these questions often revealed security problems and directed assessors to dig deeper in particular problem areas.

The CICSTART Survey builds upon the strengths of these expert questions, extending coverage to include major COBIT® Control Practices and organizing the questions by the SCADA Security Policy Framework™. Question categories include *data security*, *platform security*, *communication security*, *personnel security*, *configuration management*, *audit*, *applications*, *physical security*, *manual operations*, and *security program*.

The survey approach presently allows assessors to gather simple statistics such as the number and percentage of respondents who perform a given task or respond to a question in a given way. Such statistics help assessors better allocate their time and budget. For example, if nearly all respondents agree that a particular security control is in place, there may be little benefit in verifying the fact relative to investigating another control that the responsible personnel indicate fails to perform its intended function. The survey approach also allows for future use in evaluating the effectiveness of security awareness and training programs.

Survey questions are linked to additional metadata including references to relevant COBIT® Control Practices, task prerequisites respondents must select before they will answer a particular question, and triggers or simple Boolean rules that indicate the assessor should look further based on accumulated responses to a question. Survey administration tools may use this metadata to implement advanced features both for survey respondents and for assessors. For example, the CICSTART Questionnaire Tool uses the metadata to adapt the survey to each respondent, limiting the number of questions each respondent must endure and raising the overall quality of survey results, and to automatically generate a report template of findings and a checklist for further investigation for the assessor.

Survey questions are formatted as plain text using a simple XML schema, providing tool and platform independence. Using standard XML tools, the survey may be distributed in a variety of ways, including by paper and by computer or Internet applications.

3.2.2 The CICSTART Questionnaire Tool

The CICSTART Questionnaire Tool is one means of administering the CICSTART Survey. The questionnaire tool addresses several important requirements including

- ease and flexibility of deployment,
- controlled access to survey results,
- no installation or configuration required of survey respondents,
- automatic report and checklist generation, and
- access to summary statistics.

To address the first three requirements, the CICSTART Questionnaire Tool is a web application on a VMWare virtual machine. The free VMWare Player application is available for Windows and GNU/Linux operating systems, allowing the questionnaire tool to be deployed in a variety of ways: in particular, hosted by the assessors and accessible via the Internet; hosted on-site on a single-purpose, local-area network and accessible in person; and hosted on-site on a corporate Intranet and accessible from corporate computers. Figure 5 shows the start page for the CICSTART Questionnaire Tool.

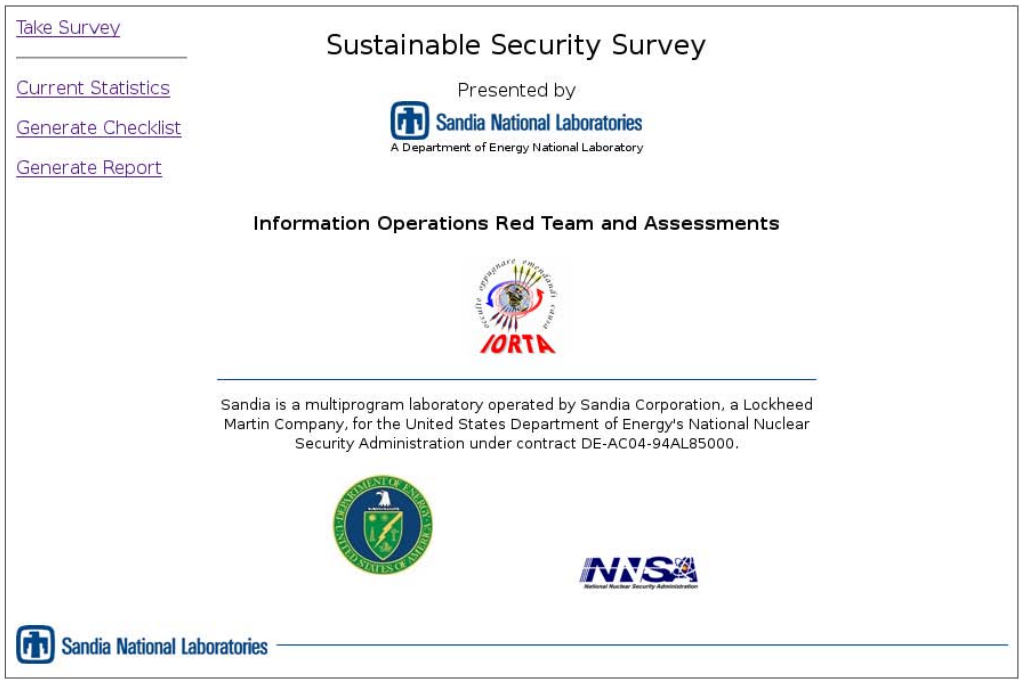

Figure 5. CICSTART Questionnaire Tool Start Page

When hosted on site (either on a local-area network or corporate Intranet), the assessed site maintains controlled access to the survey results. This option will often be the most suitable deployment option, especially for use in a self-assessment. Figure 6 shows an example of the survey questionnaire format.

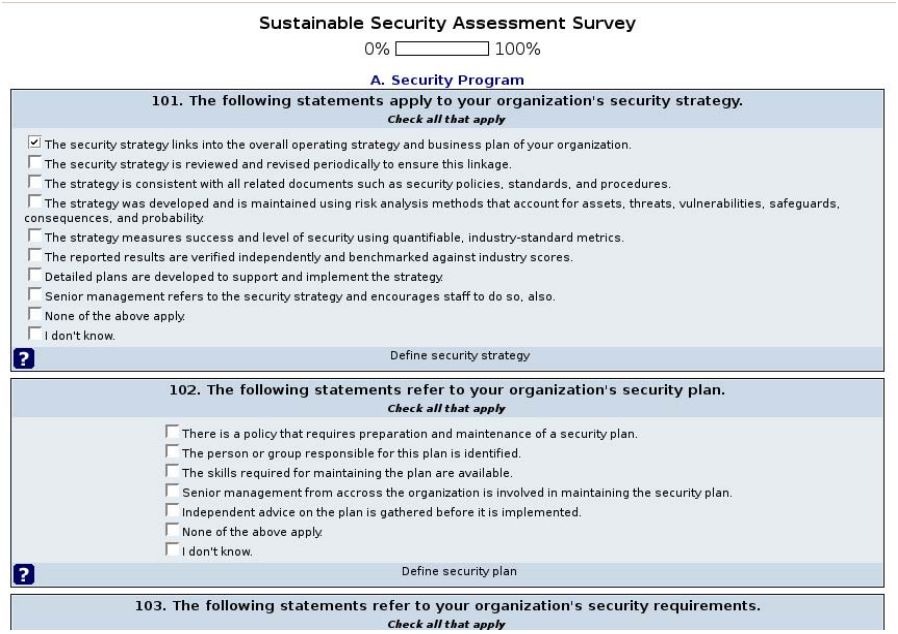

Figure 6. Example Survey Questions

Because the questionnaire tool is a web application, respondents only need a standards-compliant web browser and network access to the web-application server. The questionnaire tool is built on the open-source PHPSurveyor project [PHPSurveyor], customized to support the additional report and checklist generation capabilities. Another benefit of this implementation is that the survey may be taken concurrently by any number of site personnel.

When all required respondents have completed the survey, the assessor may generate a checklist of issues (Figure 7) that require further investigation and verification. Survey metadata known as triggers allow the questionnaire tool to generate items for the checklist, usually when no consensus answer to a question exists or when responsible personnel cannot answer a question.

Group: A. Security Program

Question Number: 104
Question: Is there a policy that specifies how security measures are communicated to authorized external users?
-Please verify whether or not there is a policy that specifies how security measures are communicated to authorized external users.

Question Number: 107
Question: Is there regular third-party evaluation of security policies?
-Please verify whether or not there is regular third-party evaluation of security policies.

Question Number: 108
Question: Are the actual security practices reviewed independently to assure that they properly reflect the policy?
-Please verify whether or not security practices are reviewed independently to assure they properly reflect policy.

Question Number: 114
Question: Are security requirements considered throughout the life of ICS projects (i.e. throughout concept, requirements, development, implementation, deployment, operation, and maintenance stages)?
-Please verify whether or not security requirements are considered throughout the life of ICS projects.

Group: B. Data Security

Question Number: 203
Question: Do any ICS applications share sensitive data with other applications?
-Please verify whether or not there are any ICS applications that share sensitive data with other applications.

Figure 7. Example Checklist

Related to the checklist, the questionnaire tool may also generate a report template (Figure 8). Items included in the report template are frequently related to industry standard security practices that by consensus of respondents are not implemented. Other items included in the report template are questions inconsistently answered, for example, when half the respondents answer in one way and the other half in a second way. Such discrepancies often indicate inadequate security awareness and training.

Group: A. Security Program

Question Number: 101
Question: The following statements apply to your organization's security strategy.
-There is confusion about your organization's security strategy. You may need ensure that your staff reviews and is familiar with the security strategy.
-There may be issues with your security strategy, such as how it is reviewed and is maintained and whether or not it is consistent with policies and procedures. Please review your security strategy.

Question Number: 102
Question: The following statements refer to your organization's security plan.
-There is some confusion about your organization's security plan. You may need to ensure that your staff is familiar with the security plan and its maintenance.
-There may be issues with your security plan, including how the plan is maintained and reviewed. Please review your security plan.

Question Number: 103
Question: The following statements refer to your organization's security requirements.
-There is some confusion about your organization's security requirements. You may need to keep your staff informed about how security requirements are documented and whether they are consistent with other requirements.
-There may be issues with your security requirements. They may not be well-documented, or they may not be consistent with other requirements. Please review your security requirements.

Question Number: 104
Question: Is there a policy that specifies how security measures are communicated to authorized external users?
-There is some confusion about whether or not there is a policy that specifies how security measures are communicated to authorized external users. It is recommended that you inform your staff of policies for communicating security requirements and ensure that they use them appropriately.

Figure 8. Report Template

Finally, the questionnaire tool provides simple access to summary statistics (see Figure 9).

Field Summary for 101:		
The following statements apply to your organization's security strategy.		
Answer	**Count**	**Percentage**
The security strategy links into the overall operating strategy and business plan of your organization. (a)	0	0.00%
The security strategy is reviewed and revised periodically to ensure this linkage. (b)	1	50.00%
The strategy is consistent with all related documents such as security policies, standards, and procedures. (c)	1	50.00%
The strategy was developed and is maintained using risk analysis methods that account for assets, threats, vulnerabilities, safeguards, consequences, and probability. (d)	1	50.00%
The strategy measures success and level of security using quantifiable, industry-standard metrics. (e)	0	0.00%
The reported results are verified independently and benchmarked against industry scores. (f)	0	0.00%
Detailed plans are developed to support and implement the strategy. (g)	0	0.00%
Senior management refers to the security strategy and encourages staff to do so, also. (h)	0	0.00%
None of the above apply. (NONE)	0	0.00%
I don't know. (IDK)	1	50.00%

Field Summary for 102:		
The following statements refer to your organization's security plan.		
Answer	**Count**	**Percentage**
There is a policy that requires preparation and maintenance of a security plan. (a)	1	50.00%
The person or group responsible for this plan is identified. (b)	0	0.00%
The skills required for maintaining the plan are available. (c)	0	0.00%
Senior management from accross the organization is involved in maintaining the security plan. (d)	1	50.00%

Figure 9. Example Survey Statistics

3.2.3 Operational Test

An alpha test for the CICSTART Survey and Questionnaire Tool was performed during the HLD-eCAM Energy Annex Training project of various state National Guard units to conduct security assessments of control systems. The test used the questionnaire tool deployed in two manners, first on a stand-alone, local-area network set up in a conference room on site and then on a corporate Intranet accessible to all relevant site personnel.

During the test, six site personnel completed the full survey. The survey results revealed several insights that were included in the National Guard out-brief to the site ownership. Difficulties arose in merging the survey results from the two questionnaire tool deployments, demonstrating the need to implement the capability into the questionnaire tool.

The alpha test demonstrated the further need to help inexpert cyber-security assessors explain when, how, and why vulnerabilities in control systems become security problems that need to be addressed. This need is addressed in part by report and checklist generation. But, the alpha test did not include report or checklist generation. These features were later tested in the beta test.

A beta test used a later iteration of the questionnaire tool with the added report and checklist generation features. Responsibility to use the questionnaire tool was turned over to the National Guard, with assistance from a Sandia assessor. For the beta test, the questionnaire tool development team was unable to add the ability to adapt the survey to the knowledge, skills, and responsibilities of each survey respondent. Those who completed the survey during the beta test reported that it was simply too long and that they were unable to fully answer some of the questions, confirming the need to adaptively pose only those survey questions relevant to each respondent. But, unexpectedly, the need to adapt the survey to each respondent extends to adaptively offering answer choices – a question may be important to ask of two different sets of personnel, but not all answer choices should be offered to each set. The next iteration of the questionnaire tool provides this level of adaptability. Finally, beta testers found the wording of some survey questions to be awkward. These questions will be rewritten in the next iteration of the CICSTART Survey.

4. Risk Estimation and Uncertainty

When evaluating risk from a random event, such as an earthquake, the analyses implicitly assume that our uncertainty is aleatory (stochastic or random). The probability measure of uncertainty is then well suited for uncertainty that is aleatory in nature. However, an adversarial act is not a random event; it is an intentional act by a thinking, malevolent person. Much of our uncertainty of the risk of a cyber attack is epistemic (state of knowledge) even if we assume that the probability of attack is 1. Neither the physical attack nor the cyber attack are a random events — they are carefully selected, planned, and executed by the adversary—but we have significant uncertainty as to what the adversaries will do and what their capabilities (e.g., “zero day” exploits) are.

To estimate the effectiveness of physical and cyber security system elements, uncertainty should be considered, but there is insufficient information to justify the use of probability as the measure of uncertainty for cyber security. We have applied the belief/plausibility measure of uncertainty from the Dempster/ Shafer Theory of Evidence to consistently capture the considerable epistemic (state of knowledge) uncertainty associated with estimating the effectiveness of cyber security elements. Appendix A summarizes the belief/plausibility measure for uncertainty. However, there is much more knowledge in physical security vulnerability analysis such that probabilistic measures of uncertainty can be used. The challenge for this project was to find a mathematical basis that can accommodate varying states of knowledge and estimate vulnerability, without over-representing the degree of knowledge.

This section provides a technique for using the belief/plausibility measure to estimate the effectiveness of a cyber security system. Since probability is a special case of belief/plausibility, the technique can also be used for evaluating combined cyber/physical security elements where the effectiveness of the physical security system elements can be modeled probabilistically. The believe/plausibility mathematics also can include the extra-special case of absolute certainty, where there is no uncertainty in the estimating parameter.

A Java computer code, *BeliefConvolution*, was written to convolute belief/plausibility distributions for algebraic operations on independent variables, including probabilistic OR (union), and to calculate belief/plausibility for fuzzy sets. The code allows aggregation of degrees of evidence into bins (linear or log spaced) to reduce the number of degrees of evidence during the successive convolution of a large number of variables.

Parts of the *BeliefConvolution* code were incorporated into the overall Java tool described in Section 5 of this report.[2]

4.1 Evaluation Technique

The goal is to evaluate Conditional Risk, *CR*, for a facility

$$CR = P_{S|A} \times C \qquad \text{(Eqn. 1)}$$

where $P_{S|A}$ is the conditional probability of adversary success (S) given an attack (A) and C is the consequence given adversary success. The risk is conditional because it assumes the adversary initiates the attack; overall risk would also include a consideration of the likelihood that the adversary initiates the attack against the facility and target of interest.

CR is dependent on the threat scenario, which includes the adversary capabilities and the target of interest to the adversary in the facility. Thus, a given facility has a set of conditional risk (*CR*) values, one for each threat scenario. Let k denote a threat scenario.

$$CR_k = P_{S|A,\,k} \times C_k \qquad \text{(Eqn. 2)}$$

[2] Specifically, calculations involving expected value intervals developed for *BeliefConvolution* were incorporated in the overall tool.

where CR_k is the conditional risk for threat scenario k.

$P_{S|A,\ k}$ is comprised of many factors based on the specific details of the threat scenario. C_k is comprised of many different types of consequences. So the evaluation of each of these terms consists of an evaluation of each constituent factor.

The project team has chosen a path analysis technique to model the cyber-related actions of the threat scenario. Since $P_{S|A,\ k}$ is a probability, it will be composed of numerous lower-level probabilities associated with the threat scenario along the paths of concern.

Let j denote a path associated with threat scenario k. For each path there is a risk

$$CR_{k,\ j} = P_{S|A,\ k,\ j} \times C_k \qquad \text{(Eqn. 3)}$$

We are dealing with intentional adversary acts, not random failures, so the adversary can choose a specific path. Therefore the measure for CR_k is *not* the sum of the $CR_{k,\ j}$,, but is the $CR_{k,\ j}$ with the highest value.[3] For paths with the same $CR_{k\ ,j}$, the one with the highest C_k is of most concern.[4] The measure for CR_k is

$$\text{Measure for } CR_k = max\ \{CR_{k,\ j} \mid all\ j\} \qquad \text{(Eqn. 4)}$$

Similarly, the measure for CR is the highest CR_k, where for equal CR_k the highest C_k is of more concern

$$\text{Measure for } CR = \max\ \{CR_k \mid \text{all } k\} \qquad \text{(Eqn. 5)}$$

Let C denote overall consequence for which C_k is the value for threat scenario k.

C is the sum of all the constituent types of consequences. Let m denote a specific type of consequence.

$$C_k = \sum C_{k,\ m} \qquad \text{(Eqn. 6)}$$

where the C_m have been scaled as appropriate for summation. For example, a death may be equivalent to $\$10^6$, where dollars, or "$", is a measure of the willingness to pay to prevent a specific consequence.

$P_{S|A,\ k,\ j}$ for a path j is evaluated by combining the probabilities for each edge in the path, and the probability for each edge is evaluated by combining the probabilities for each security primitive on the edge for the specific threat scenario k (see Figure 14).

---

3 For random acts, the measure of CR_k would be the sum of the $CR_{k,\ j}$.

4 "Highest" based on plausibility.

The probabilities are combined using the algebra of probability. Assuming independence, given two probabilities P_X and P_Y for two events X and Y, the probability of both X and Y (AND or intersection) is $Px \cdot P_Y$, and the probability of either X or Y (OR or union) is $Px + P_Y - Px \cdot P_Y$.

Each variable has uncertainty.[5] That is, a given constituent of $P_{S|A,\ k,\ j}$ is a probability with a range [0, 1] and there is uncertainty as to the exact value of the variable in that range.

Similarly, each constituent of $C_{k,\ m}$ has a range [minimum consequence, maximum consequence] and there is uncertainty as to the exact value of the variable in that range. Uncertainty for a variable is captured by assigning a measure to the values over the range for the variable.[6]

One widely used measure for uncertainty is probability. But if the information available is nonspecific, the probability measure does not include all of the uncertainty.[7]

Due to the lack of fidelity of the information available for evaluating intentional acts, a measure of uncertainty more general than probability will be used, that measure being belief/plausibility. Probability is a special case of belief/plausibility, so if some variables can be modeled with probability the use of the belief/plausibility measure is still valid.[8]

It is important to note that each possible *value of a variable* is calculated by performing the mathematical operations of concern on the values of the constituent variables. The *uncertainty in each value* is calculated by convoluting uncertainty distributions for the constituent variables. For example, for three variables *A, B*, and *C* the possible values for $D = (A + B) C$ are $\{d \mid d = (a + b) c$ where a is over A, b is over B, and c is over $C\}$. The uncertainty for each d is evaluated by convoluting the uncertainty distributions for *A, B*, and *C* using the mathematics appropriate for the measure being used for uncertainty (i.e., belief/plausibility for our evaluation).

4.1.1 Simple Example of Belief/Plausibility

This section provides a simple example of the use of the belief/plausibility measure for uncertainty.

Consider two independent consequences X and Y to be added to form an overall consequence $Z = X + Y$. For any given specific values of X and Y, x and y, Z has a specific value $z = x + y$.

If X and Y have no uncertainty, then both X and Y have one value, x_1 and y_1, and Z has the one value $z_1 = x_1 + y_1$, and there is no uncertainty for Z.

---

5 Here variable means a random variable, say X, which is a mapping from the Sample Space (S) of interest to the Reals (R). $X: S \rightarrow R$. S is the domain, R is the codomain, and $X(S)$ is the range. To be precise, "values of a random variable" means the range of the random variable.

6 Only one value will occur, but there is uncertainty as to which value will occur.

7 Nonspecificity is a type of uncertainty. See Appendix A.

8 For a variable such as $P_{S|A,\ k}$ quantified as a probability, the term probability is used in the objective, or frequency sense as the value of the variable. The uncertainty in the variable is measured with belief/plausibility and under certain conditions belief and plausibility both reduce to probability where the term probability is used in the subjective or state of knowledge sense as a measure of uncertainty. Thus, probability is used to mean two different concepts: the value (objective) and the uncertainty in the value (subjective).

If *X* and *Y* have uncertainty, and the information about *X* and *Y* is specific enough to use probability as the measure of uncertainty, then the uncertainty in *Z* resulting from addition of all the possible values of *X* and *Y* is a probability calculated by convoluting the probability distributions for *X* and *Y* under addition to produce a probability distribution for *Z*.

If the information about *X* and *Y* is too nonspecific to use probability as the measure of uncertainty, then a more general measure of uncertainty called belief/plausibility can be used. For many of the variables associated with the effectiveness of cyber security elements in countering intentional acts by adversaries, the information is too nonspecific to use probability and the use of a belief/plausibility measure is needed.

Instead of assigning a measure of uncertainty to values of *X* and *Y* (as done with probability), belief/plausibility allows a measure of uncertainty to be assigned to intervals over *X* and *Y*. For example, let *X* range from 0 to 100 and let the degrees of evidence be as follows:[9]

0.20 for the interval [0, 10]
0.35 for the interval [2, 25]
0.45 for the interval [82, 100]

The belief/plausibility for exceeding a given consequence can be easily evaluated. Let *x* be the consequence to be exceeded. The plausibility for exceeding *x* is the sum of all degrees of evidence for which the associated interval has *any* overlap with the interval (*x*, 100]. The belief for exceeding *x* is the sum of all degrees of evidence for which the associated interval lies (completely) *within* (*x*, 100]. For example, let *x* be 15; the interval of interest is (15, 100]. Figure 10 shows the situation graphically.

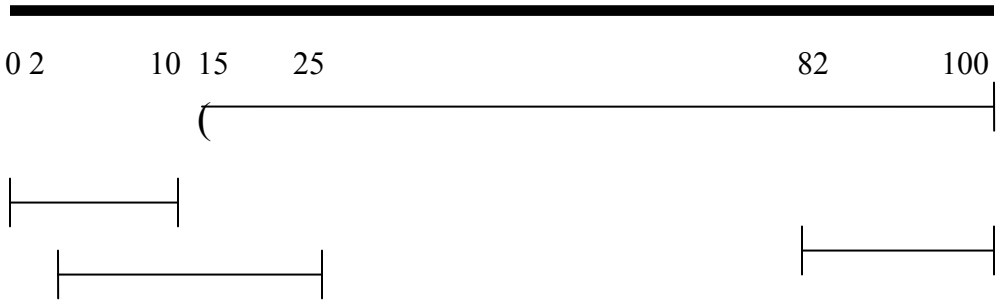

Figure 10. Intervals for Simple Example

Both [2, 25] and [82, 100] overlap (15, 100] so the plausibility for (15, 100] is 0.35 + 0.45 = 0.8. [82, 100] is the only interval that lies within (15, 100] so the belief for (15, 100] is 0.45.

[9] Using standard interval symbols, [means include and (means exclude. For example, [*a, b*] denotes all real numbers between *a* and *b* including *a* and *b*. (*a, b*] denotes all real numbers between *a* and *b* excluding a and including *b*.

The interval calculation is more complex for convolution, but is straightforward for convolution involving simple functions such as addition or multiplication of two constituent variables. The degrees of evidence for the result are calculated by forming the relation consisting of all 2-tuples with the first element of the tuple an interval with a non-zero degree of evidence from the first variable and the second element of the tuple an interval with a non-zero degree of evidence from the second variable. The degree of evidence for each tuple is the product of the two degrees of evidence for each element in the tuple assuming independence. An interval for the convoluted result is obtained by applying the appropriate function (e.g., addition or multiplication) to the intervals in the tuple.

For example, let *X* range from 1 to 20 with the following degrees of evidence and intervals:

0.8 for [2, 15]
0.2 for [1, 10]

Let *Y* range from 0 to 30 with the following degrees of evidence and intervals:

0.7 for [5, 25]
0.3 for [0, 4]

Let *Z* be the convolution resulting from the addition of *X* and *Y*. *Z* ranges from 1 to 50. *Z* has the following degrees of evidence for the indicated tuples:

0.8 * 0.7 for <[2, 15], [5, 25]>
0.8 * 0.3 for <[2, 15], [0, 4]>
0.2 * 0.7 for <[1, 10], [5, 25]>
0.2 * 0.3 for <[1, 10], [0, 4]>

Since the function is addition, *Z* has the following degrees of evidence for the indicated intervals:

0.56 for [7, 40]
0.24 for [2, 19]
0.14 for [6, 35]
0.06 for [1, 14]

The likelihood of exceedance for *Z* is shown in Figure 11. Also, Figure 11 shows the expected value interval for *Z*.

(NOTE: Had *Z* been modeled using probability, belief and plausibility both become probability and the expected value interval is a point value, the mean. For this case, Figure 11 would have one curve, probability, and a point estimate expected value, the mean of the probability distribution.)

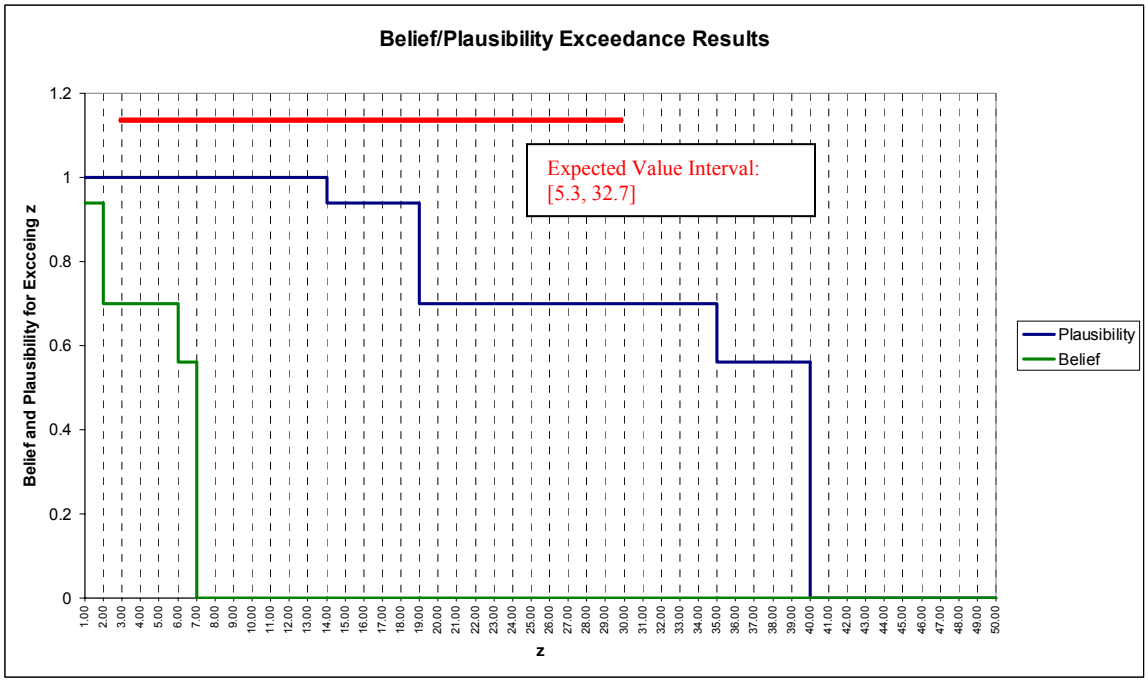

Figure 11. Belief/Plausibility for Example for Variable *Z*

4.1.2 Vagueness

To summarize results, a variable may be partitioned into various linguistic events. This aspect was explored, but not incorporated into the Functional Assessment tool described in Section 5. For example, "Consequence" can be partitioned into "Minor", "Moderate", "Major", and "Catastrophic". These linguistic terms are fuzzy sets where fuzziness reflects a type of uncertainty called vagueness. Vagueness is the uncertainty in categorizing a *known* value of a variable. For example, for a given day that is partially cloudy there is uncertainty as to the degree to which that day is "Sunny." A fuzzy set captures vagueness by allowing partial membership of a value in the set.

Figure 12 shows a partitioning of Z into linguistic events represented by fuzzy sets. The linguistic methods are available in *BeliefConvolution*; however, these have not been enabled in the methodology described in Section 5.

As an example of vagueness, consider the value $z = 21$. It is uncertain as to whether "21" is "Minor" or "Moderate" as reflected by the partial degree of membership of "21" in each of these fuzzy sets.

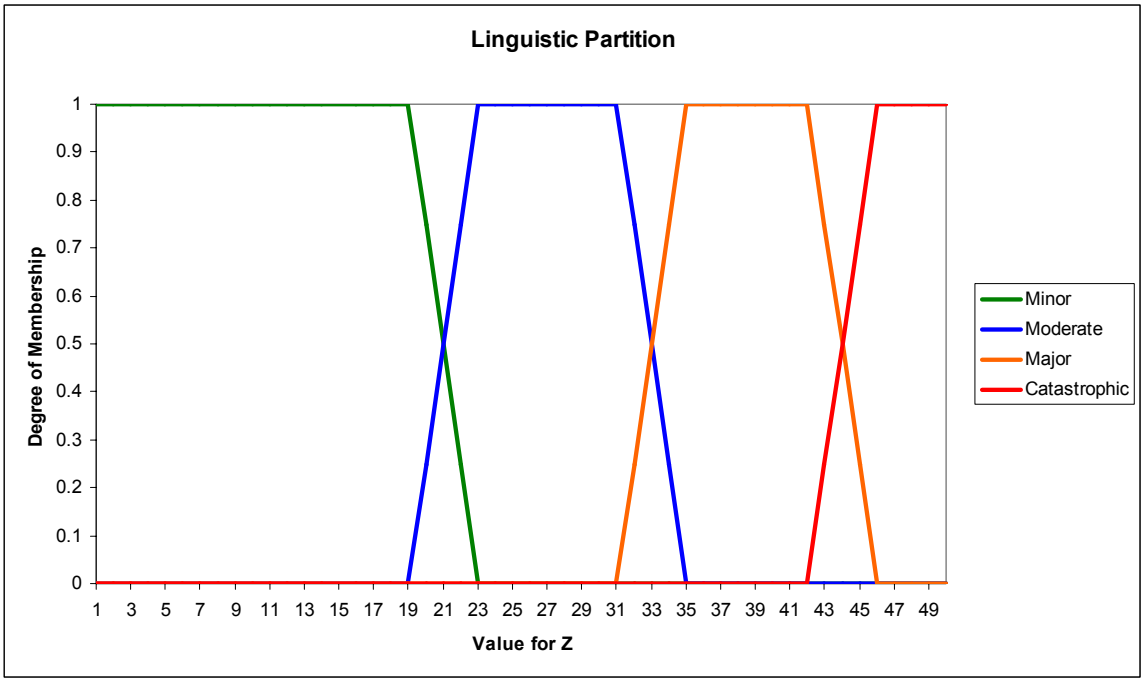

Figure 12. Linguistic Partition of *Z*

A belief/plausibility interval can be calculated for each fuzzy set. For *Z*, the results are shown in Figure 13. These results were calculated using the *BeliefConvolution* code.

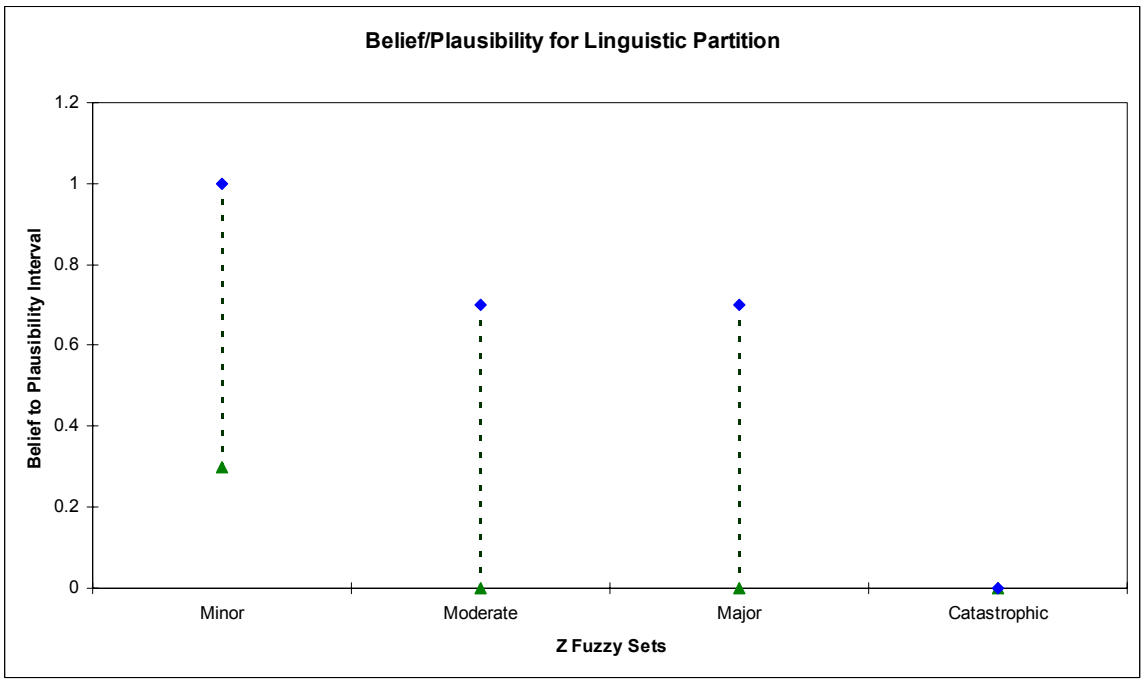

Figure 13. Uncertainty for Linguistic Partition of *Z*

For the simple example $Z = X + Y$, Figure 11 is the "numerical" result and Figure 13 is the "linguistic" result.

4.1.3 Evaluation of Conditional Risk (*CR*)

CR as defined in Equation 1 can be evaluated using belief/plausibility as the measure for uncertainty and that the results of the evaluation can be provided in both numeric and linguistic form.

CR will be built up from its constituent factors as discussed earlier. Each factor will have a numerical result (and if a linguistic partition is defined for that variable) a linguistic result, so the results can be provided at any level of the model.

That is, CR, CR_k, $P_{S|A,\ k,\ j}$, C_k, and so on each have a numerical result such as that in Figure 11, and- given definition of a linguistic partition- a linguistic result such as that in Figure 13.

Evaluation of $P_{S|A,\ k}$

For threat scenario k, $P_{S|A,\ k,}$ is built up from the $P_{S|A,\ k,\ j}$ as follows, using the simple example for two paths shown in Figure 14.

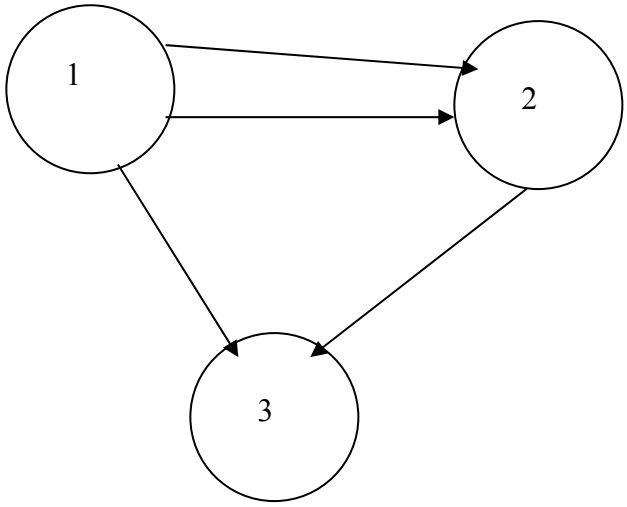

Figure 14. Two-Path Example.

Path 1-2-3 consists of two edges (1-2 and 2-3), and the edge from 1-2 is modeled as two parallel edges called $[1\text{-}2]_1$ and $[1\text{-}2]_2$. Path 1-3 has one edge 1-3. For the threat scenario of interest, two types of data are applied to the graph: (a) security primitives for cyber security and (b) adversary characteristics. There may be numerous security primitives, such as Authentication, Network Access Control (NAC), and User Access Control (UAC). Each security primitive has a level of "hardness"; example levels for Authentication are:

- no password,
- weak password with periodic change,
- strong password with no periodic change,

- strong passwords with periodic change, and
- strong passwords with periodic change and limits on failed password attempts.

The adversary characteristics consist of factors such as:

- funding,
- goal intensity
- stealth
- physical access
- cyber skills
- implementation time
- cyber organization size

Let *t* denote a security primitive of a certain type, for example, Authentication, and let *h(t)* denote the hardness level of that primitive (e.g., strong passwords with periodic change.) Let *e(j)* denote an edge on a path *j*. Each security primitive for each edge for each path is evaluated as the probability that the adversary with characteristics specified by threat scenario *k* can defeat this security primitive.[10] Let $P_{h(t),\, e(j),\, k}$ denote this probability. The uncertainty for each $P_{h(t),\, e(j),\, k}$ is captured by assigning degrees of evidence over the [0, 1] range**.** Using expertise available, a library of degrees of evidence can be pre-defined for each $P_{h(t),\, e(j),\, k}$.

For a given edge *e(j)* the measure for that edge is the probabilistic combination of all the $P_{h(t),\, e(j),\, k}$ associated with that edge. Let $P_{e(j),\, k}$ denote the probability that the adversary successfully compromises edge *e(j)*. A higher probability implies more vulnerability.

$$P_{e(j),\, k} = OPERATION\ P_{h(t),\, e(j),\, k} \text{ over all } t \text{ elements on the edge} \qquad \text{(Eqn. 7)}$$

where *OPERATION* denotes the collection of probabilistic addition and/or probabilistic multiplication as appropriate for the logic of the security elements on the edge.

For example, let *e(j)* have two security primitives *SP1* and *SP2* with hardness levels *h1(SP1)* and *h2(SP2)*. If both *SP1* and *SP2* must be defeated, then for threat scenario *k*:

$P_{e(j),\, k} = P_{h1(SP1),\, e(j),\, k} * P_{h2(SP1),\, e(j),\, k}$. If defeat of either *SP1* or *SP2* must is sufficient, then
$P_{e(j),\, k} = P_{h1(SP1),\, e(j),\, k} + P_{h2(SP1),\, e(j),\, k} - P_{h1(SP1),\, e(j),\, k} * P_{h2(SP1),\, e(j),\, k}$.

Parallel edges can be used to model "OR" choices where an adversary can move from goal state 1 to goal state 2 via either path between goal state 1 and goal state 2. With this convention on any given edge, the logic then is purely "AND" since an adversary must defeat all of the security primitives on that particular link. Using this convention

$$P_{e(j),\, k} = \Pi\, P_{h(t),\, e(j),\, k} \text{ over all the elements on the edge} \qquad \text{(Eqn. 8)}$$

and for *n* parallel edges $e_n(j)$, the measure for the combined edges is

[10] Probability in the objective sense (frequency).

$$P_{e(j),k} = \sum_{probabilistic\ sum\ all\ n} P_{e_n(j),k} \quad \text{(Eqn. 9)}$$

The probability for the path $P_{S|A,\ k,\ j}$ as is evaluated as the product of the probabilities of each edge in the path:

$$P_{S|A,k\ j} = \prod_{all\ e(j)} P_{e(j),k} \quad \text{(Eqn. 10)}$$

The probability for the threat scenario k is $P_{S|A,\ k}$ and is evaluated as the worst path:

$$P_{S|A,\ k} = max\ \{P_{S|A,\ k,j} \mid all\ j\ paths\} \quad \text{(Eqn. 11)}$$

since for an intentional act the adversary has the choice of paths. (NOTE: This assumes that the adversary knows which path is easiest, which is a conservative assumption. In practice, a defender might use deception and operational security techniques to attempt to limit the adversary's knowledge.)

Evaluation of CR_k

The consequence C_k is evaluated as the sum of the constituent consequences m for threat scenario k as indicated in Equation 6. Degrees of evidence must be assigned to each $C_{k,\ m}$ based on the information available for the potential consequences for the specific target and specific adversary capabilities specified by threat scenario k. Contrary to the case for the $P_{h(t),\ e(j),\ k}$, a library of degrees of evidence cannot be created for the $C_{k,\ m}$ since the evidence is target-specific.

After both $P_{S|A,\ k}$ and C_k have been evaluated, CR_k and then CR are evaluated using Equations 2 and 5, respectively.

4.1.4 Example Calculation

This section applies the *BeliefConvolution* code to an example calculation of CR. The example has a great deal of uncertainty in the data to emphasize the ability of the approach to model highly uncertain information.

For this example consider a case with two paths of concern as shown earlier in Figure 12, using the convention that "OR" choices are modeled as parallel edges. One of the parallel edges for 1-2 has two security primitives SP_1 and SP_2 with hardness $h_2(SP_1)$ and $h_4(SP_2)$.[11] The other edge for 1-2 has one security primitive SP_3 with hardness $h_2(SP_3)$. Edge 1-3 has security primitives/hardness: $h_4(SP_1)$ and $h_6(SP_2)$, and edge 2-3 has security primitive/hardness $h_3(SP_5)$. Consider a specific threat scenario k.

[11] Here, $h_n(SP_m)$ denotes hardness level n for security primitive type m.

Assume that from the data library, the degrees of evidence for the security primitives of the specified hardness for the specified threat are as given in Table 2.

Table 2. Data for Degrees of Evidence for Security Primitives for Threat Scenario k

Security Primitive and Hardness	Interval	Degree of Evidence
$h_2(SP_1)$	[0.3, 0.6]	0.7
	[0.4, 0.5]	0.2
	[0.5, 0.75]	0.1
$h_4(SP_2)$	[0.5, 0.7]	0.2
	[0.2, 0.8]	0.8
$h_2(SP_3)$	[0.4, 0.8]	1.0
$h_4(SP_1)$	[0.2, 0.4]	0.9
	[0.2, 0.6]	0.1
$h_6(SP_2)$	[0.1, 0.4]	0.4
	[0, 0.5]	0.6
$h_3(SP_5)$	[0.5, 0.9]	0.5
	[0.4, 0.8]	0.2
	[0, 1.0]	0.3

The probability for path 1-3 is:

$P[h_4(SP_1)] * P[h_6(SP_2)]$

The probability for path 1-2-3 is:

$\{P[h_2(SP_1)] * P[h_4(SP_2)] + P[h_2(SP_3)] - P[h_4(SP_2)] * P[h_2(SP_3)] * P[h_2(SP_3)] \} * P[h_3(SP_5)]$.

Using the results of the calculation and using the upper expected value interval for ranking paths, Path 1-2-3 is worse (higher probability of adversary success) than path 1-2, so using Equation 9, $P_{S|A,\ k}$ is $P_{S|A,\ k,\ 1\text{-}2\text{-}3}$.

Figure 15 shows the numerical result for path 1-2-3.

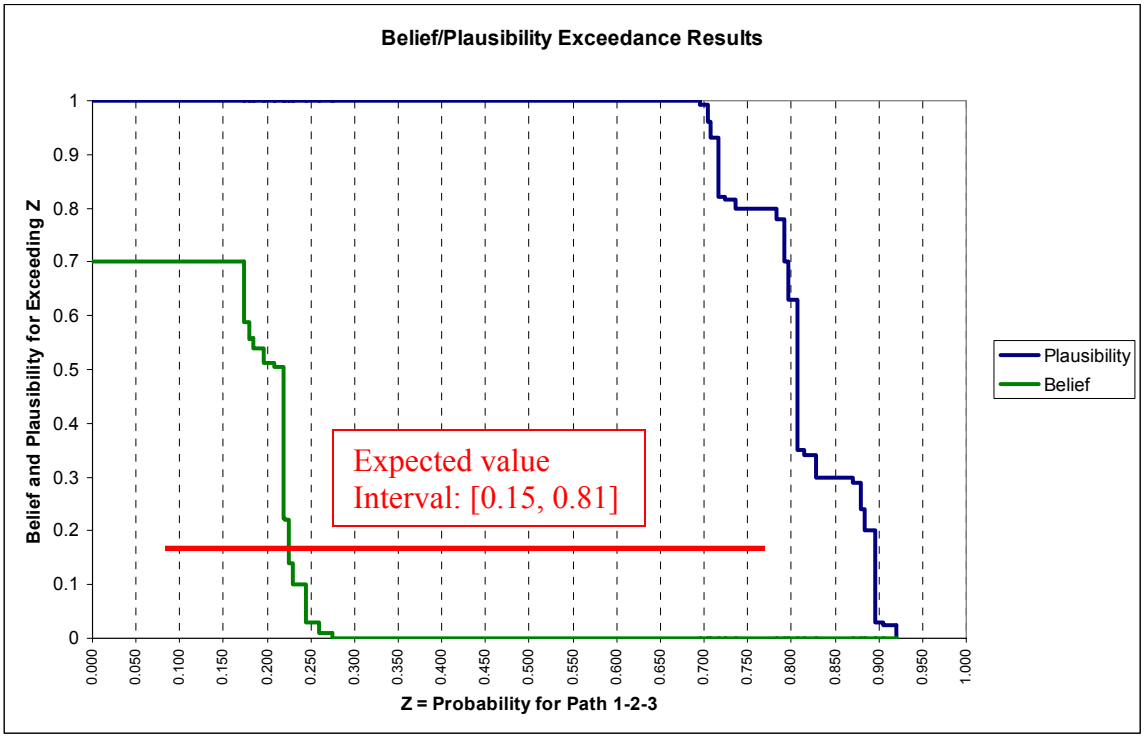

Figure 15. Numerical Result for Probability of Adversary Success (Path 1-2-3)

Let C be comprised of two types of consequences with evidence for threat scenario k as indicated in Table 3.

Table 3. Consequences for Example

Consequence	Interval	Degrees of Evidence
$C_{k,1}$	$[10^4, 10^6]$	0.55
	$[10^3, 10^7]$	0.45
$C_{k,2}$	$[10^2, 10^6]$	0.60
	$[10^4, 10^7]$	0.33
	$[10^1, 10^5]$	0.07

Figure 16 shows the results for C_k.

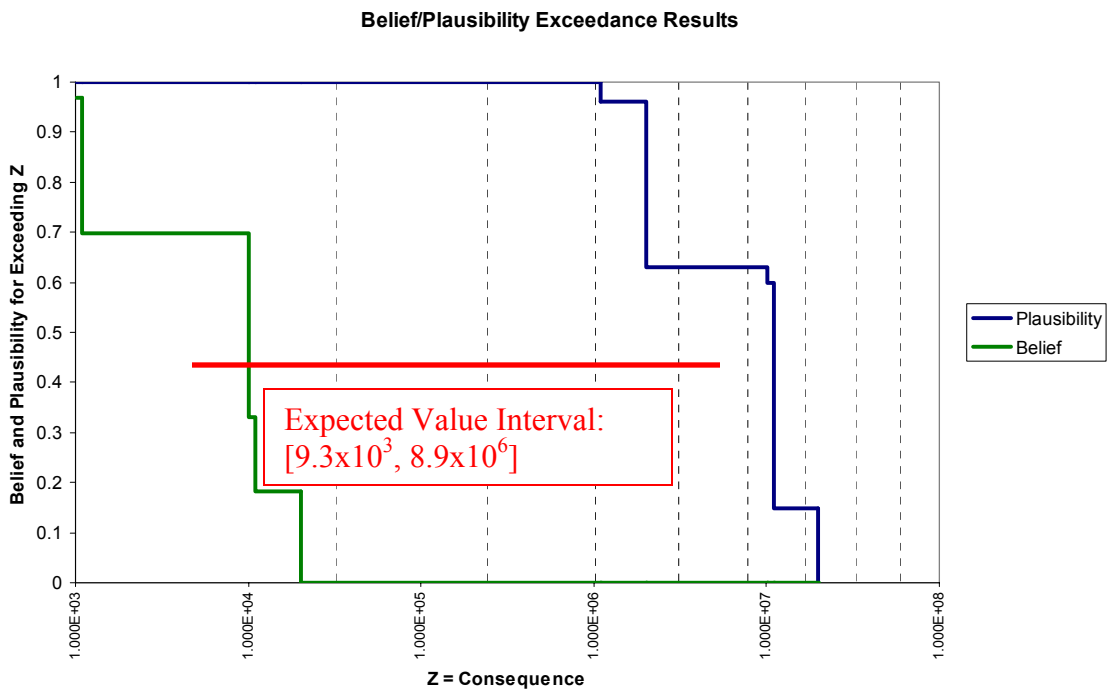

Figure 16. Numerical Result for Consequence

Figure 17 shows the results for CR_k.

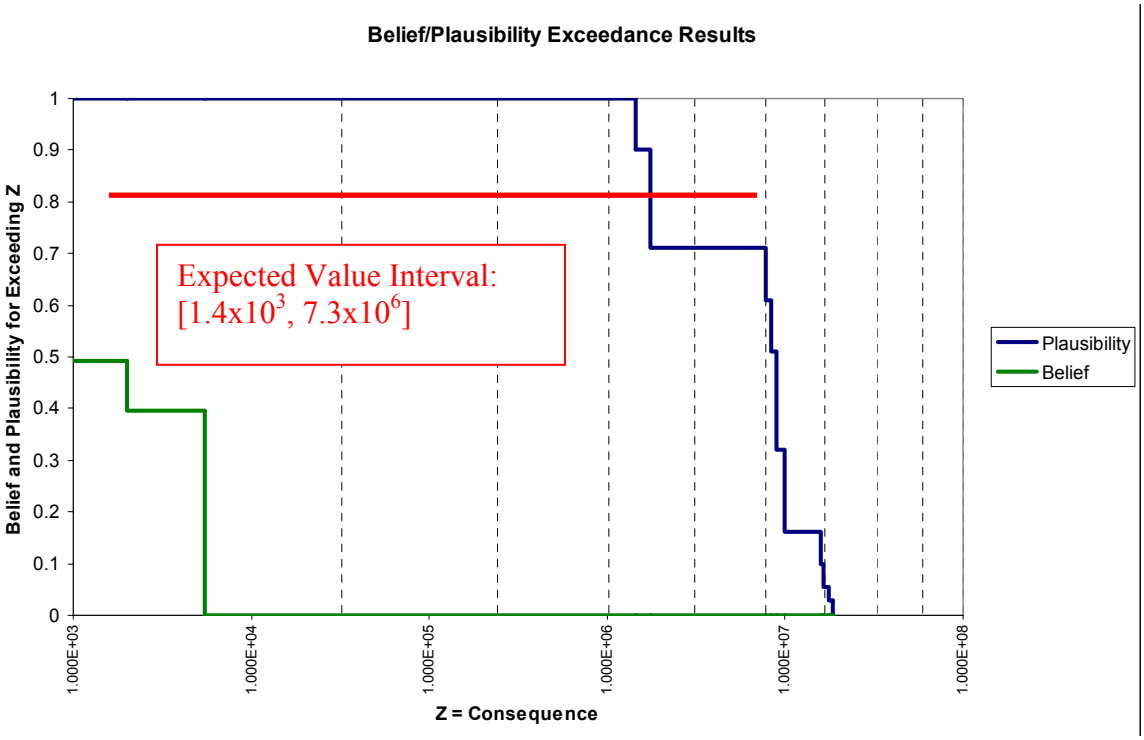

Figure 17. Numerical Result for Conditional Risk

Assume that Conditional Risk is defined linguistically as in Figure 18.

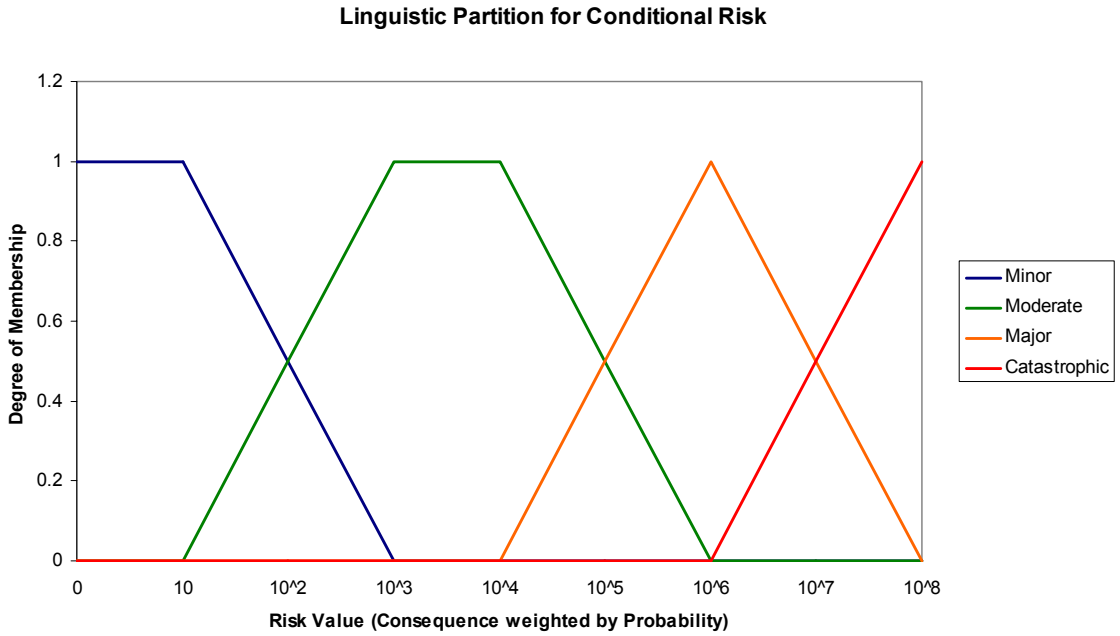

Figure 18. Linguistic Partition for Conditional Risk

Using the linguistic partition of Figure 18, the linguistic result for the example can be calculated using *BeliefConvolution*. These results are provided in Figure 19 and show, for this example, the state of knowledge is highly uncertain for the Moderate and Major descriptors for risk. These results are provided in Figure 19.

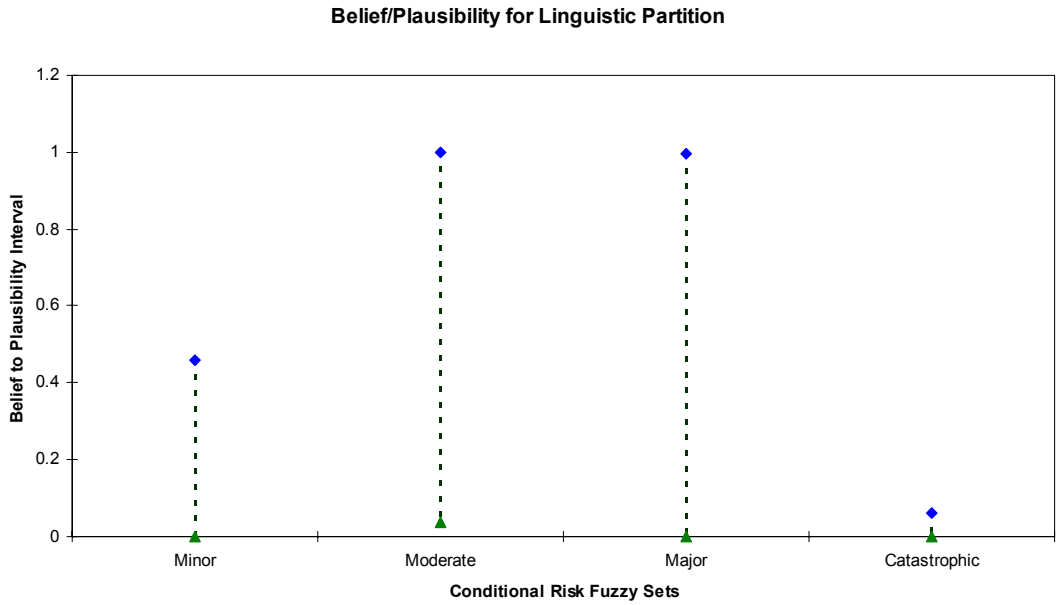

Figure 19. Linguistic Result for Conditional Risk

4.2 Presentation of Results for a Collection of Scenarios

The detailed numerical results for each scenario are presented in Figure 20, Figure 21, and Figure 22. In application, a number of scenarios will be evaluated and the expected value interval can be used to summarize the results for a large number of scenarios.

The following variables will be evaluated for a number of specific threat scenarios:[12]

- P_{Epo}, the probability of PPS success for a physical-only attack
- P_{Ec}, the probability of cyber system success for a cyber-only attack
- P_{Epz}, the probability of PPS success given defeat of cyber controlled elements
- C_n, the consequence of type n (e.g., deaths, outage weeks, etc.)
- C, the sum of all C_n using the common measure of \$, willingness to pay.

A threat scenario includes: adversary resources (attributes and knowledge), target, and attack plan. Let "k" denote a specific threat scenario. Each of the variables is dependent on the threat scenario, so the evaluation will consist of the set of values $\{P_{Epo\,k}, P_{Ec\,k}, P_{Epz\,k}, C_{n\,k}, C_k\}$ for each threat scenario.

For $P_{Epo\,k}$ and $P_{Epz\,k}$, a point value will be calculated. It is assumed that the point value is the mean of a probability distribution if these variables were evaluated by convoluting probability distributions for all the constituent terms.

For $P_{Ec\,k}$, $C_{n\,k}$, and C_k a belief/plausibility distribution will be calculated, and an expected value interval of the distribution will be calculated.

$C_k = \sum C_{n\,k}$ where convolution is used to propagate uncertainty.[13] Similarly, other measures can be calculated using convolution. For example, $P_{Ec\,k} \cdot C$ is the conditional risk for a cyber only attack. $P_{Ecp\,k} \equiv 1 - (1 - P_{Ec\,k}) \cdot (1 - P_{Epz\,k})$ is another measure.

Let $[E_*(X)$, and $E^*(X)]$ denote the expected value (interval) for variable X based on a belief/plausibility measure; for the special case of probability $E_*(X) = E^*(X) = \text{Mean}(X)$, the mean of the probability distribution. All of the variables of interest (probabilities and consequences) have non-negative values, and assuming the variables are noninteractive (independent for probability), for any two variables X and Y:

- $E_*(X+Y) = E_*(X) + E_*(Y)$ and $E^*(X+Y) = E^*(X) + E^*(Y)$
- $E_*(X \cdot Y) = E_*(X) \cdot E_*(Y)$ and $E^*(X \cdot Y) = E^*(X) \cdot E^*(Y)$

[12] A system effectiveness probability is related to the adversary success probability by $P_{system} = 1 - P_{adversary}$.

[13] The BeliefConvolution Java code performs the convolution using the belief measure. Probability is a special case of belief.

For example, as indicated in Figures 15, 16, and 17, from the detailed convolution calculation, the upper expected value for conditional risk, 7.3×10^6, is equal to the product of the upper expected values for adversary success and consequence, 0.81 and 8.9×10^6, respectively. The lower expected value for conditional risk, 1.4×10^3, is equal to the product of the lower expected values for adversary success and consequence, 0.15 and 9.3×10^3, respectively.

So, the expected values of functions involving addition/multiplication of variables can be calculated without performing the convolution. (Of course, without performing the convolution the uncertainty distribution for the function is not known.) For example, the expected value for $P_{Ec\,k} \cdot C$ can be directly calculated from the expected values of $P_{Ec\,k}$ and C without performing the convolution and then calculating the expected value of $P_{Ec\,k} \cdot C$.

The top-level numerical results can be presented as expected values for the variables of interest, where the expected value is in general an interval that is a point (the mean) for some of the variables.

The top-level results are formed from a matrix, each row consisting of a vector. Specifically, a row is:

$\{\text{Mean}(P_{Epo\,k}), [E_*(P_{Ec\,k}), E^*(P_{Ec\,k})], \text{Mean}(P_{Epz\,k}), [E_*(C_{n\,k}), E^*(C_{n\,k})], [E_*(C_k), E^*(C_k)]\}$ and k ranges over all threat scenarios to form all the rows of the matrix.

For any function of interest, the expected value interval can be easily calculated, without convolution, from the expected values of its arguments as previously discussed. For example, for the function $P_{Ecp\,k} \equiv 1 - (1 - P_{Ec\,k}) \cdot (1 - P_{Epz\,k})$ the expected value is:

$[1 - (1 - E^*(P_{Ec\,k})) \cdot (1 - \text{Mean}(P_{Ecz\,k})), 1 - (1 - E_*(P_{Ec\,k})) \cdot (1 - \text{Mean}(P_{Ecz\,k}))]$.

For a specific threat scenario k, let $P_{x\,k}$ denote a probability of type x ($P_{Epo\,k}$, $P_{Ec\,k}$, $P_{Epz\,k}$, $P_{Ec\,p}$, etc.) and let $C_{x\,k}$ denote a consequence of type x ($C_{n\,k}$, or C_k).[14] For each $P_{x\,k}$ there is an expected value interval $[E_*(P_{x\,k}), E^*(P_{x\,k})]$. For each $C_{x\,k}$ there is an expected value interval $[E_*(C_{x\,k}), E^*(C_{x\,k})]$. There is also an expected value interval for the function $P_{x\,k} \cdot C_{x\,k}$: $[E_*(P_{x\,k}) \cdot E_*(C_{x\,k}), E^*(P_{x\,k}) \cdot E^*(C_{x\,k})]$.

The top level numerical results for a specific threat scenario are these expected value intervals for $P_{x\,k}$, $C_{x\,k}$, and $P_{x\,k} \cdot C_{x\,k}$. Figure 20 is an example of the top-level results for one threat scenario. Note *that* $P_{x\,k} \cdot C_{x\,k}$ is $C_{x\,k}$ scaled down by $P_{x\,k}$; that is, $P_{x\,k} \cdot C_{x\,k}$ is the probability of the consequence where $C_{x\,k}$ is the consequence given adversary success and $P_{x\,k}$ is the probability of adversary success given attack.

$P_{x\,k}$ will be called the "Likelihood", $C_{x\,k}$ is the "Consequence", and $P_{x\,k} \cdot C_{x\,k}$ will be called the "Expected Consequence."

[14] $P_{x\,k}$ can also represent one minus any of the probabilities of interest, such as $1 - P_{Ec\,k}$.

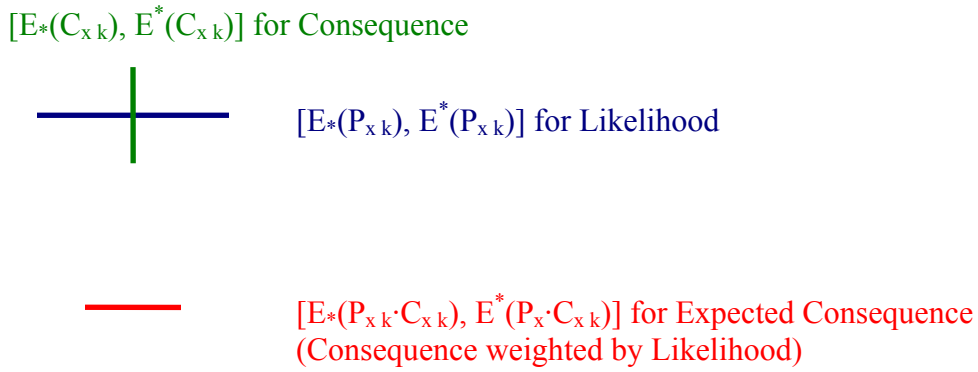

Figure 20. Example Results for One Threat Scenario

Using the top-level numerical results for one threat scenario, aggregate results can be generated; e.g., for a fixed target, the results for all threat resources can be presented as in Figure 21.

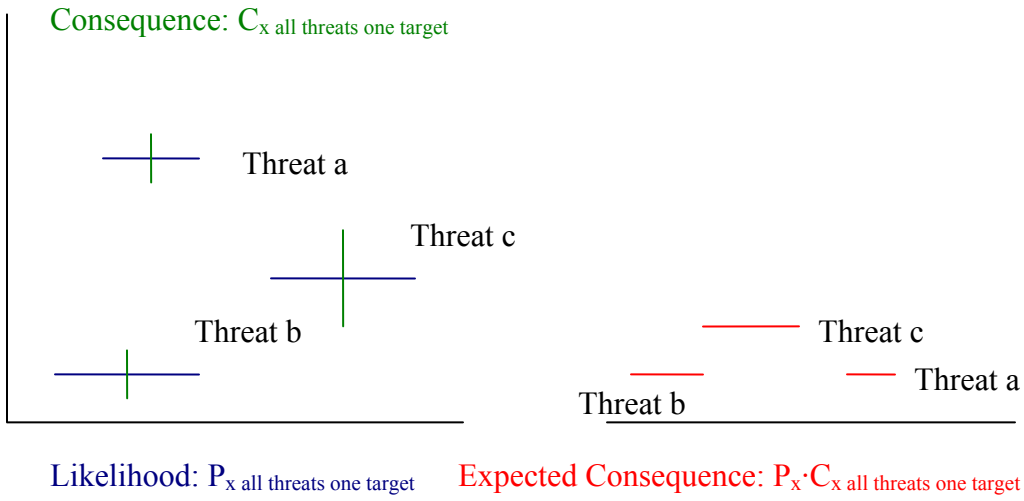

Figure 21. Example Results for All (Three) Threat Resources for One Target

Similarly, for a fixed threat resource over all targets, the results can be summarized (Figure 22).

Other roll-ups can be created as necessary.

In summary, the top-level numerical results are based on aggregating expected values for Consequence, Likelihood, and Expected Consequence for each threat scenario. The scenarios can be ranked by decreasing concern using either of two methods: (1) rank by the upper value of the expected value interval, E^*, and subrank by the lower value of the expected value interval, E_*, or (2) rank by the point estimate $(E^* + E_*)2$. The linguistic results for a collection of scenarios can be presented as in Figure 23; here scenarios are rolled up over all targets for a specific threat resource.

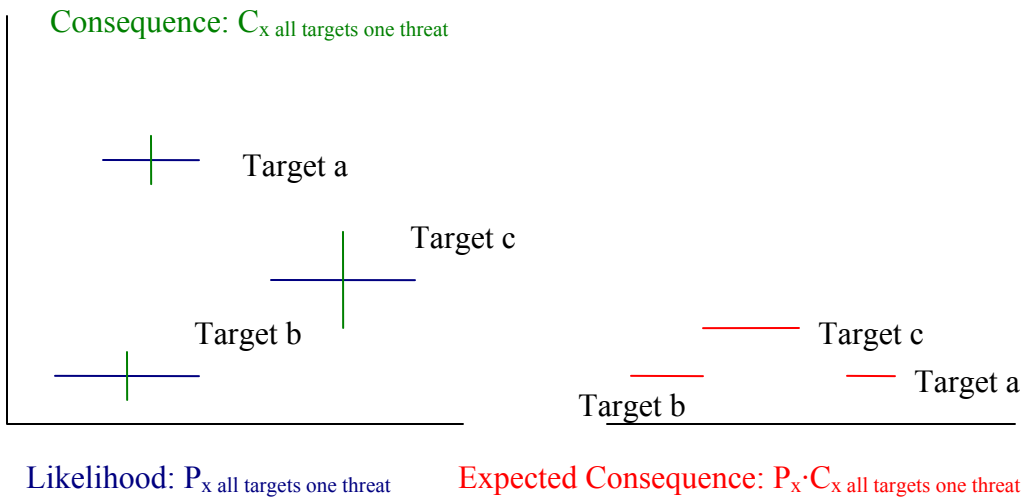

Figure 22. Example Results for All (Three) Targets for One Threat Resource

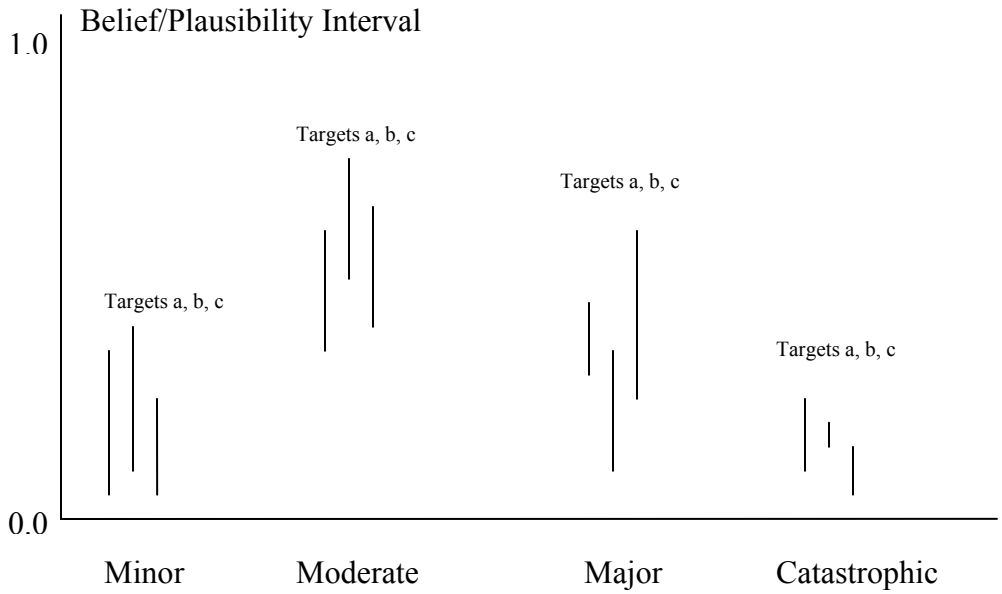

Figure 23. Linguistic Results for Consequence for One Threat Resource over All (Three) Targets

Scenarios can be ranked linguistically by decreasing concern using plausibility for the "worst" fuzzy set, "Catastrophic" in the example, subranked by plausibility for the next worst fuzzy set, "Major" in the example, and so on.

5. Functional Tool

5.1 Cyber-Physical Security Analysis Methodology (CPSAM)

The risk assessment methodology shown in Figure 24 is a "functional assessment" that uses six steps to estimate the risk associated with various consequences of concern, protective measures, asset locations, and threat levels. The goal of the methodology is to provide the facility operator with a relative ranking of the assets that might require improved protection. The facility operator can then perform trade studies to determine where to spend a limited capital improvement budget – as well as where to best position operational security assets and personnel.

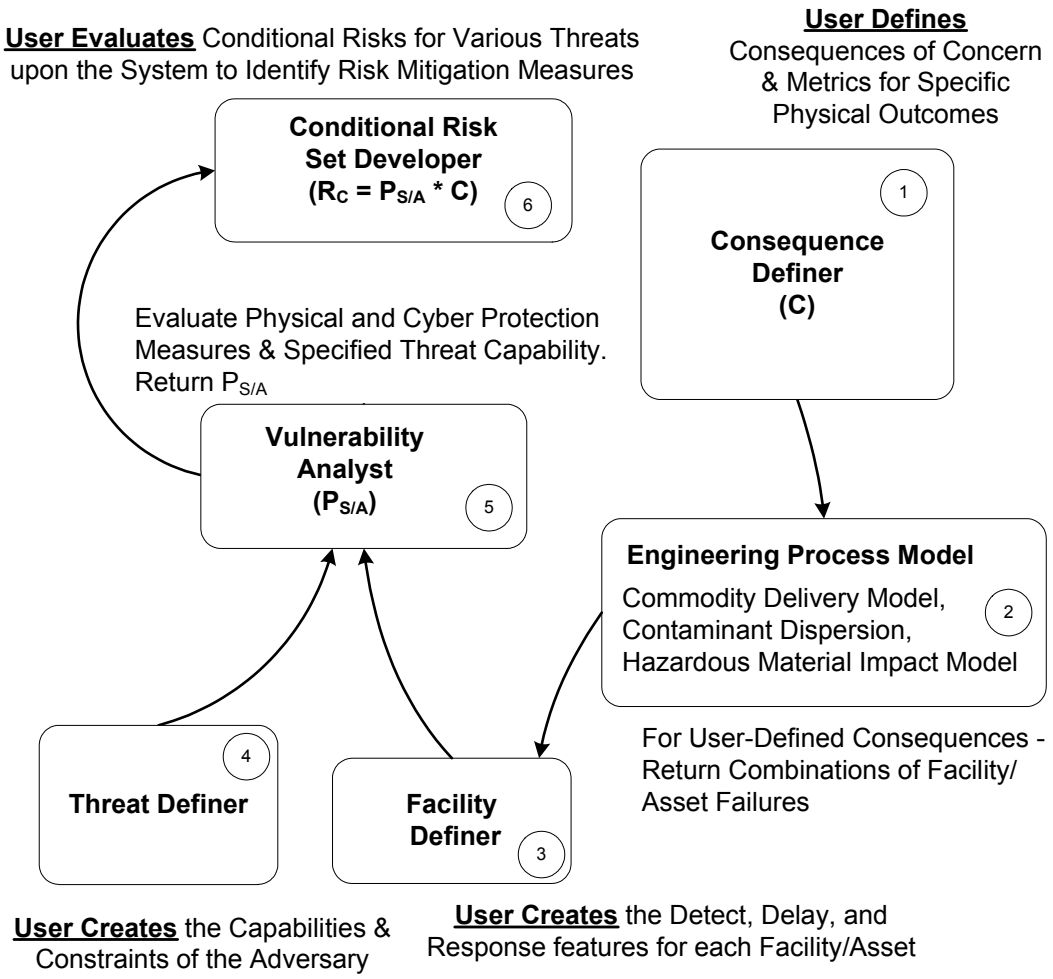

Figure 24. CPSAM Use Case Diagram

The high-level use case for this methodology is shown in Figure 24. Step one (1) is that the risk analyst must first determine the Consequence of Concern (CoC) and with associated "unacceptable" system states. An Engineering Process Model (EPM) in step two (2) or expert opinion can then be used to determine the asset-specific "cut sets" that can cause a given CoC. The Vulnerability Assessment (VA) process comprises the traditional steps of a) defining the protective measures (both physical and cyber) for each asset; b) defining the appropriate Threat

Model; and c) comparing the protective measures, asset susceptibility and threat characteristics to generate the "Probability of Success given an Attack" ($P_{S|A}$) for each cut set that can cause a CoC. The Conditional Risk Set Developer then estimates the risk associated with various consequences of concern, protective measures, asset locations and threat levels. The risk is "conditional" because the model assumes that a specified attack occurs. Figure 25 shows a flow diagram showing the detailed steps for data inputs required by the user.

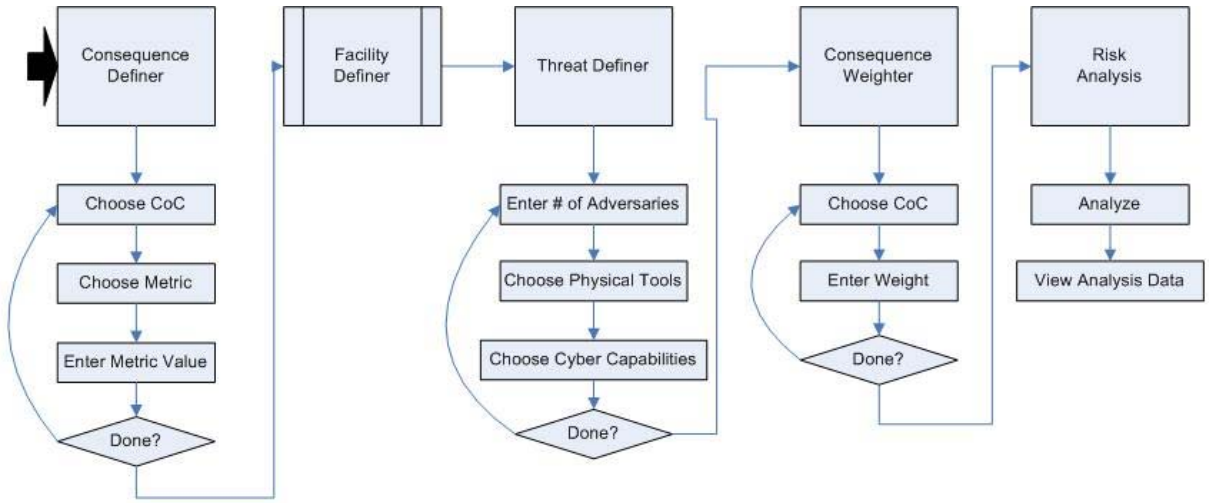

Figure 25. User Input Flow Diagram

5.2 Consequence Definer

5.2.1 Consequences of Concern

For water systems, CoCs might include: a) inadequate flow; b) unacceptable quality; and c) hazardous material releases. For example, the "inadequate flow" CoC might be further broken down as:

- loss of service to "general/residential" customers,
- loss of service to critical customers (hospitals or large employers), or
- loss of fire-fighting service to those same classes of customers.

Unacceptable quality might include both unpalatable water (due to taste, odor, and turbidity) and nonpotable water that would cause illness in the general population. For these CoCs, the physical outcomes are typically duration, extent, and recovery costs. The associated metrics are time, numbers of customers affected, and dollars. The physical outcomes for hazardous material releases from water system facilities might include Cl_2, H_2O_2 and NH_3 releases. The associated metrics are health impacts such as deaths, illness levels, and hospital admissions. Figure 26 shows that the use can define a consequence with a title description, metric name and value/range with the Belief/Plausibility method to describe uncertainty in the native metric value.

Section 5.2.2 describes how a risk analyst can differentiate between differing CoCs with disparate physical outcomes and metrics using a Willingness to Pay consequence conversion process.

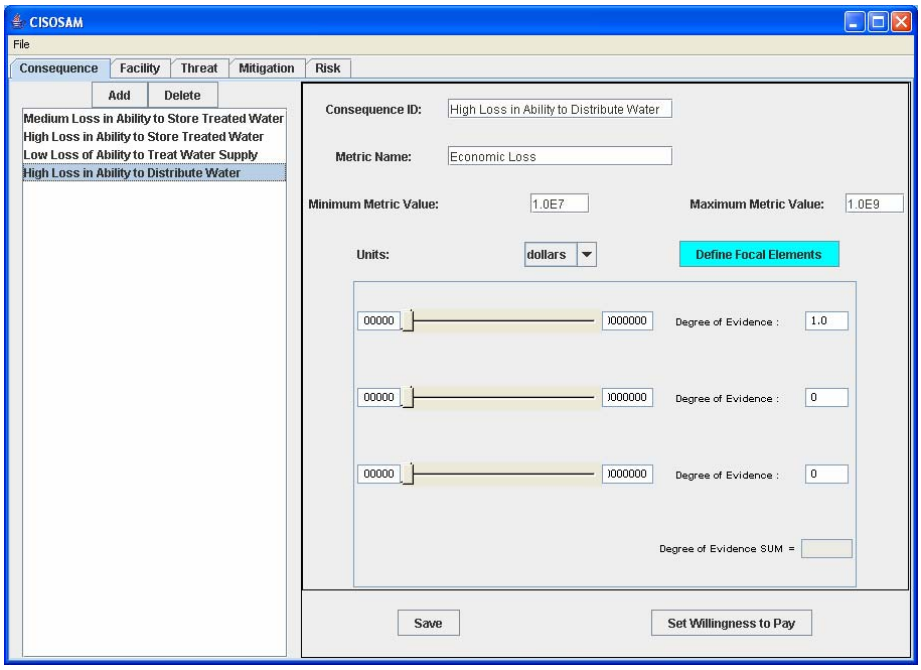

Figure 26. Native Consequence Definer

5.2.2 Consequence Conversion – Willingness to Pay

A major challenge to developing a single score for differing consequences is that there are potentially differing physical outcomes with differing metrics that define the transition from an acceptable state to an unacceptable state. The Telecommunications Outage Index [ANSI 1997] is one example of a scoring function that uses a multi-criteria utility function to scale the importance of each failed service, the duration of each outage, and the magnitude of each outage (based on the number of customers and the time of day when the outage occurs).

This research developed a general consequence index method that assesses the user-perceived importance of selected CoCs by using a willingness to pay function. For each CoC, several physical outcomes may be evaluated. For example, a previous section of this report described three general classes of consequences for water systems: inadequate flow, unacceptable quality, and hazardous material releases. The physical outcomes for flow and water quality might include duration, extent, and recovery cost for the abnormal event. Each of these physical outcomes may have a different importance value that can be estimated by a willingness to pay to prevent this outcome. For each physical outcome, metrics defining the severity of that outcome can be used as a scaling factor to distinguish between small, medium, or large impacts, and commensurate willingness to pay to prevent the occurrence. For example, the willingness to pay to prevent one physical outcome might have a high value, but if a large number of similar

physical outcomes were to occur, the willingness to pay is not a simple product of the willingness to pay value and the number of occurrences (see Figure 27).

Since not all consequences, physical outcomes, and metrics have equal perceived impacts to the infrastructure management, the willingness to pay approach provides a method to discriminate these details. The consequence index can then be used in conjunction with the vulnerability values to identify the most important risks that need to be managed.

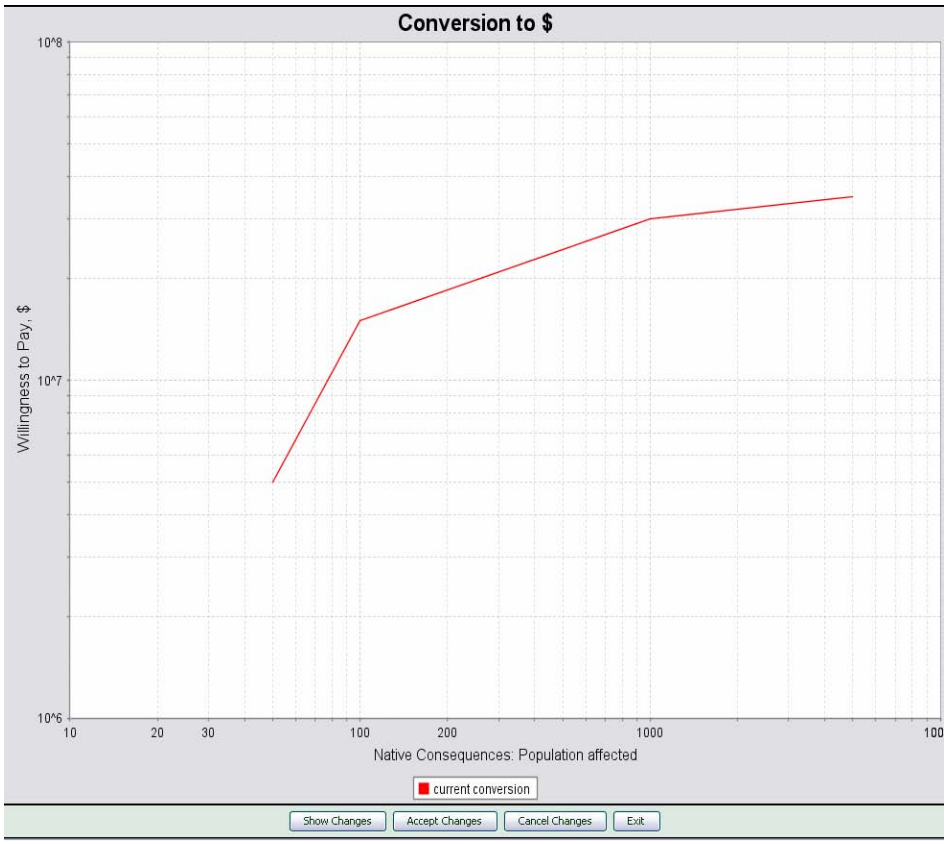

Figure 27. Example of How Willingness to Pay is Affected by Consequence Magnitude

5.3 Engineering Process Model

Engineering process models (EPMs) exist for most every large commodity infrastructure. Each one has potential features to exploit to connect an asset, or combinations of assets, that can cause the CoC. These can be complicated engineering models that the project decided not to explicitly embed within CPSAM.

For water systems, the EPANET software [EPANET] has the following seven asset-types: reservoirs; pumps, pipes, junctions, valves, emitters, and tanks. EPANET has a Graphical User Interface (GUI) and file import capability that allows an analyst to produce a flow-based model of a water system. Add-on software can be written to generate the appropriate cut sets for various flow-based CoCs. These cut sets can then be used as inputs for the Facilities Definer that describe the protective measures associated with the critical asset and Vulnerability Analyst

(VA) functional blocks. (Note: the prototype CPSAM software did not link to EPANET at the completion of the LDRD effort.)

5.4 Facility Definer

The Facility Definer is a multi-step process (Figure 28) where the user inputs the asset linked to the CoC in the Engineering Process Model Step (Figure 29). Each asset then requires an assignment to a specific physical location and a cyber location, which requires input details on the physical security and cyber security posture.

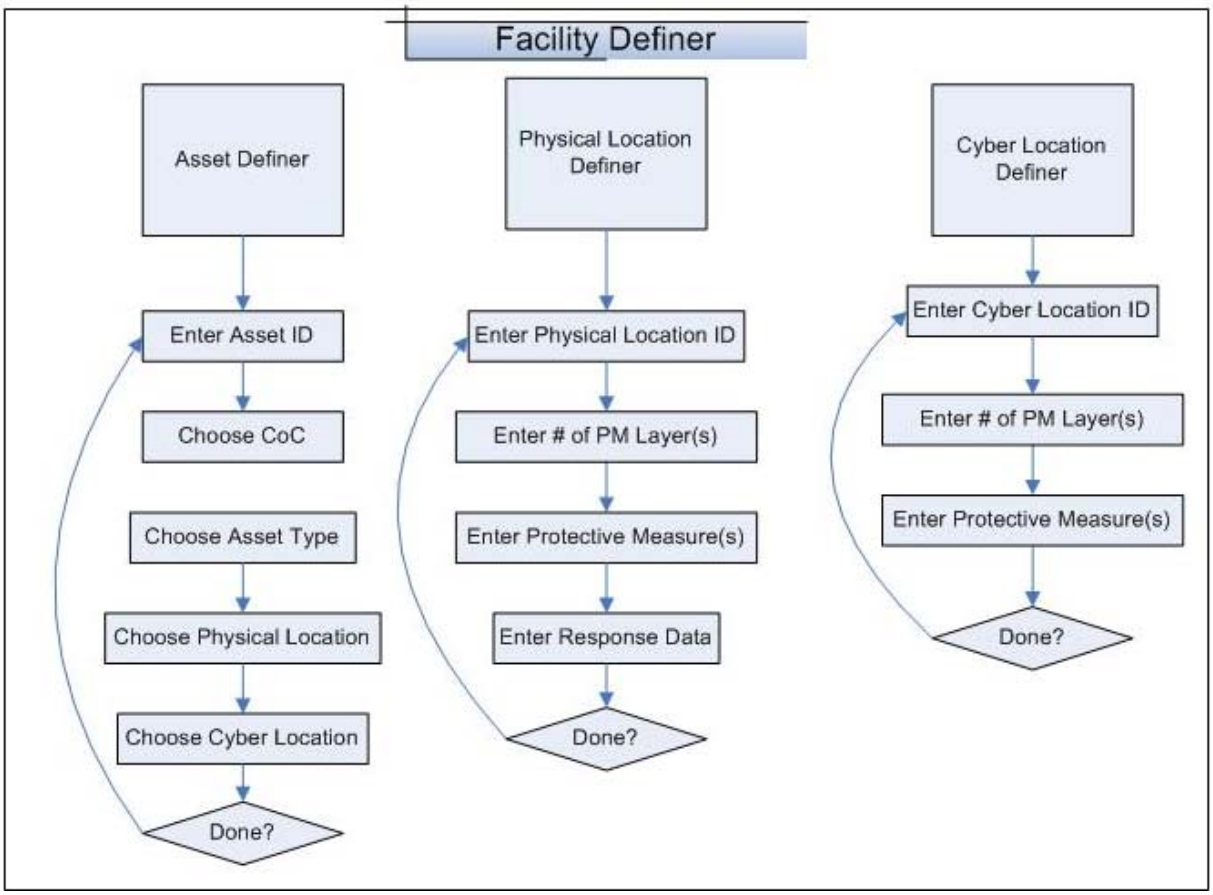

Figure 28. Facility Definer User Input Flow Chart

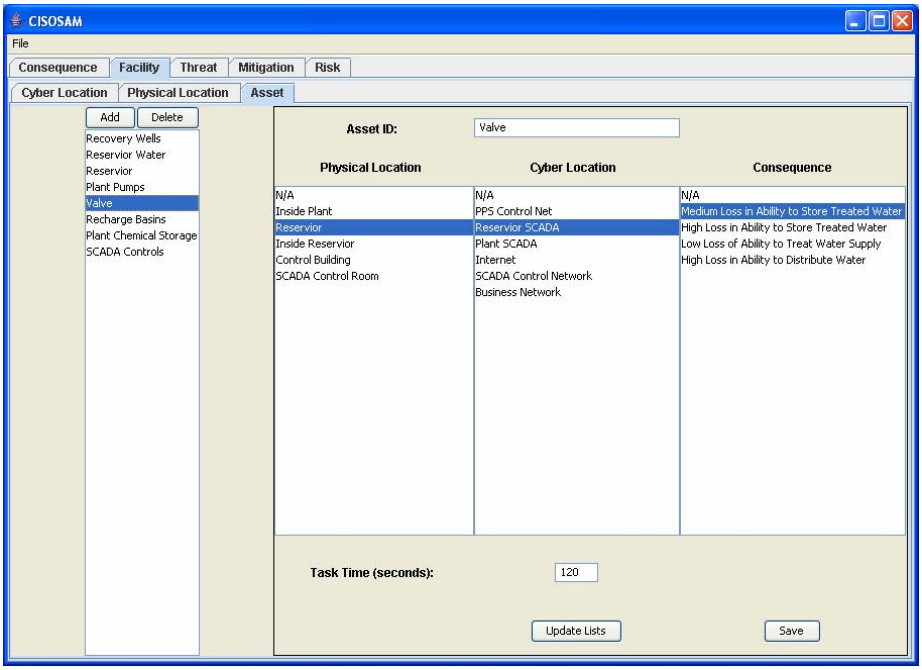

Figure 29. Asset Definer

The Physical Security posture is evaluated by the Estimate of Adversary Sequence Interruption (EASI) mathematics, which are based on a probabilistic logic analysis method (Garcia 2001). For each physical location, the number of protective layers are identified and defined by type, description, and whether these layers are cyber-controlled. Section 5.6.1.1 describes the EASI model.

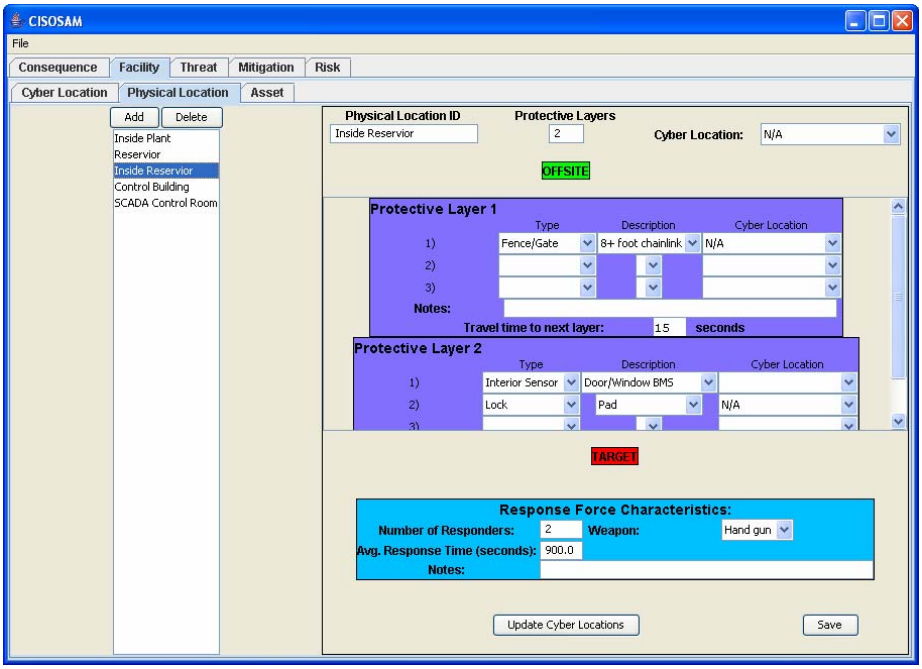

Figure 30. Physical Location and Physical Security Posture User Input

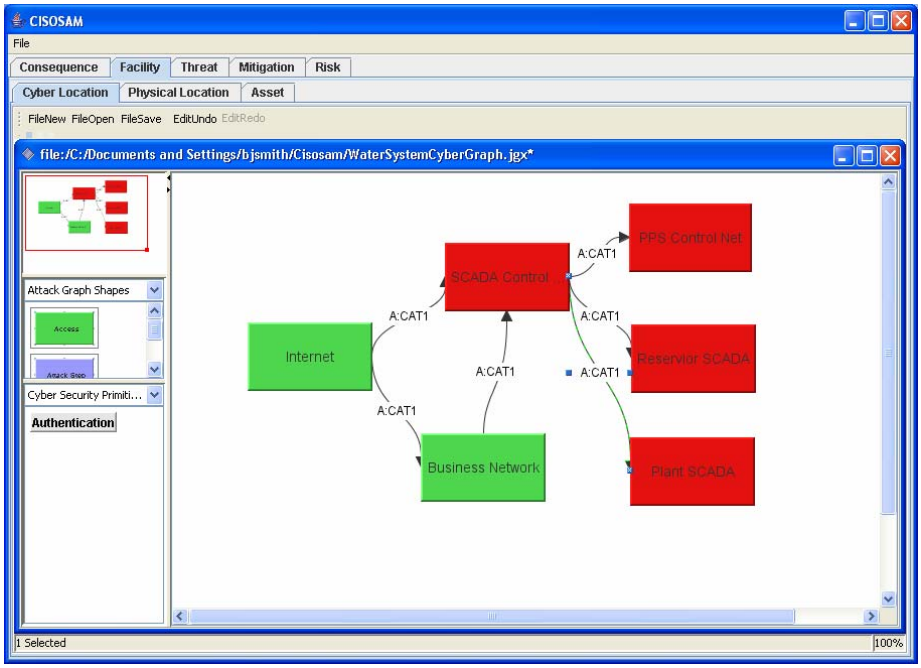

Figure 31. Cyber Location and Cyber Security Posture User Input

5.5 Threat Definer

The Threat Definer (TD) establishes the specific capabilities of the adversary. In this research, the physical-attack capabilities were sorted into four broad categories: hand-tools, power-tools, explosives, and vehicles. The susceptibility of various physical protection systems (e.g., doors, locks, and fences) to these four physical attack categories has been well quantified in experimental programs. Figure 32 shows the selections for an open field Threat ID that contains the integer number of adversaries and weapons that are used to compare against the Physical Security posture using the EASI calculation engine. Considerably less work has been done to quantify cyber adversaries and their impact on cyber protective systems.

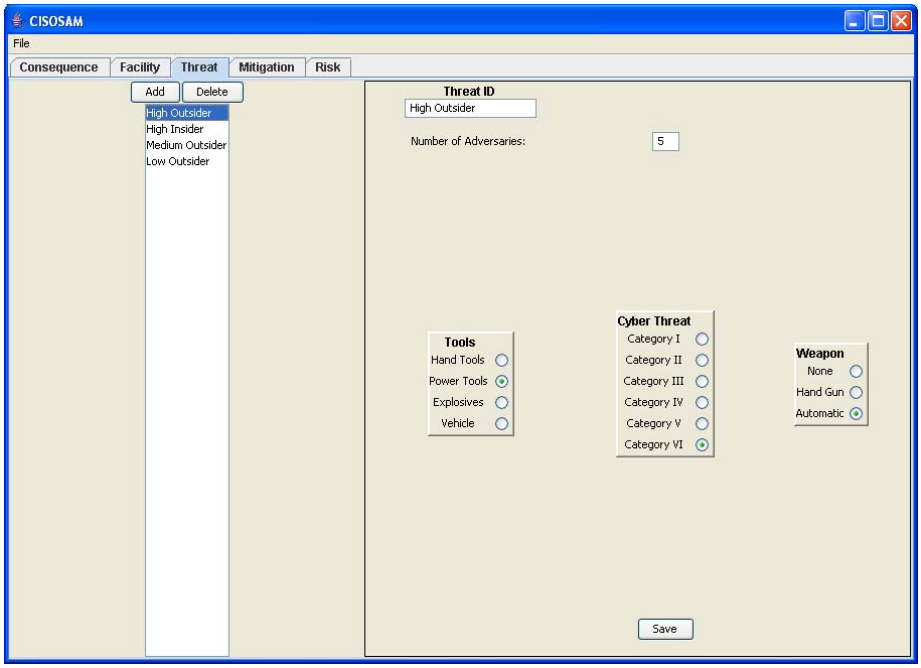

Figure 32. Threat Definer

5.5.1 Adversary Physical Threat Characteristics and Capabilities

In the physical-security literature, the physical-attack capabilities are often sorted into four broad categories: hand tools, power tools, explosives, and vehicles. The susceptibility of various PPSs (e.g., doors, locks, and fences) to these four physical attack categories has been well quantified in experimental programs (Garcia 2001). For example, power tools and explosives can damage the state of a physical asset or facilitate entry through locked doors.

5.5.2 Adversary Cyber Threat Characteristics, Capabilities, and Categories

Most past work on cyber adversaries have focused on qualitative descriptions of characteristics and capabilities. This work required quantified cyber adversary characteristics and capabilities for determine the impact on elements of cyber protection systems (CPSs). The following cyber-attack capabilities [Duggan 2005] were used to derive category-based cyber threats:

- Funding
- Goal Commitment Intensity
- Stealth
- Physical Access
- Cyber Skills
- Implementation Time
- Cyber Organization Size

5.5.2.1 Funding

The funding characteristic is often an enabler for other characteristics. Funding is a multiplier factor that may enhance any other attribute. However, increased funding may reduce the adversary's stealth since they may now be using resources outside their own organization. Representative High (H), Medium (M) and Low (L) funding levels for this characteristic might currently be:

- H – Hundreds of thousands to millions of dollars.
- M – Thousands to hundreds of thousands of dollars.
- L – Zero to thousands of dollars.

5.5.2.2 Goal Commitment Intensity

This characteristic is based on the adversary's determination in achieving their goals or objectives. Representative levels for this characteristic are:

- H – A group member is willing to die to achieve the organization's goals.
- M – A group member is willing to be caught or captured, and possibly go to prison.
- L – Group members are not willing to be caught or captured.

5.5.2.3 Stealth

This characteristic is defined as the required level of stealth necessary to achieve the adversary's goal. When the required level is not maintained, the goal will not be achieved. Representative levels are:

- H – Loss of stealth prior to attack execution cannot be tolerated.
- M – Either loss of absolute stealth can be tolerated or total stealth cannot be achieved due to other restrictions.
- L – Either stealth prior to execution is not a requirement or stealth is not considered important to the threat group.

5.5.2.4 Physical Access

This characteristic is based on whether the threat group is able to gain physical access to some cyber resource for some portion of an attack. In addition, access to certain design-level information is sometimes only available to someone with physical access to the actual system. The group's funding level and time frame often enhance this characteristic. Representative levels are:

- H – Able to gain long-term physical access to the cyber resource by placing someone in the proper employment, turning an insider, or other means.

- M – Able to identify where physical access is needed and then gain the required physical access through some short-term method such as blackmail, coercion, or breaking and entering.

- L – Cannot physically access the cyber resource.

5.5.2.5 Cyber Skills

This characteristic considers the threat's cyber skills, training programs, and research programs but does not include skills that are found outside the organization, or those that may be purchased. Representative levels are:

- H – Detailed knowledge of current exploits, an internal training program, and an active research and development (R&D) program in new exploits.

- M – Good knowledge of current exploits, some capacity for internal education, but not much for R&D on new exploits.

- L – Some knowledge of current exploits and some skill with information technology, no capacity for a training or R&D program.

5.5.2.6 Implementation Time

This characteristic is the total amount of time that an organization is willing to use in planning, developing, and deploying a cyber attack. It includes the time necessary for all steps up to the actual execution of an attack. Representative levels are decades/years, years, months, and weeks.

5.5.2.7 Cyber Organization Size

This characteristic accounts for the size and social networking ability of the cyber-literate members of the threat group. The levels imply a structure for the group as well. Representative levels are:

- Hundreds – Hundreds of individuals are working together and communicating.

- Tens of Tens – Many small groups communicate loosely between the groups. Limited information is moved between groups.

- Tens – Small workgroups that work independently.

- Ones – Individuals that work independently.

Table 4 shows a consolidation of the cyber threat characteristic parsed with the scaled capabilities to derive six general categories. These six general cyber threat categories are compared with the cyber protection system capabilities to determine the success or failure of the cyber threat against the cyber protection system on an element by element basis (see Section 5.6.2 System Effectiveness against Cyber Attack)

Table 4. Threat Characteristics

Threat Category	Funding	Goal Intensity	Stealth	Physical Access	Cyber Skills	Implementation Time	Cyber Org Size
I	H	H	H	H	H	Decades/ Years	Hundreds
II	H	H	H	M	M	Years	Tens of Tens
III	M	H	M	M	M	Months	Tens
IV	L	M	H	L	H	Months	Tens
V	L	M	M	L	M	Months	Ones
VI	L	L	L	L	L	Weeks	One

5.6 Vulnerability Analyst

The method produces a quantitative measure for "asset invulnerability," which is the ability of the facility to withstand an attack against a commodity-delivery asset without suffering the CoCs associated with that asset's failure.

Figure 33 illustrates the top-level logic for assessing the asset invulnerability. The consequences associated with an asset's failure can be avoided by either: 1) preventing the failure of the asset; or 2) mitigating the effects of the asset's failure.

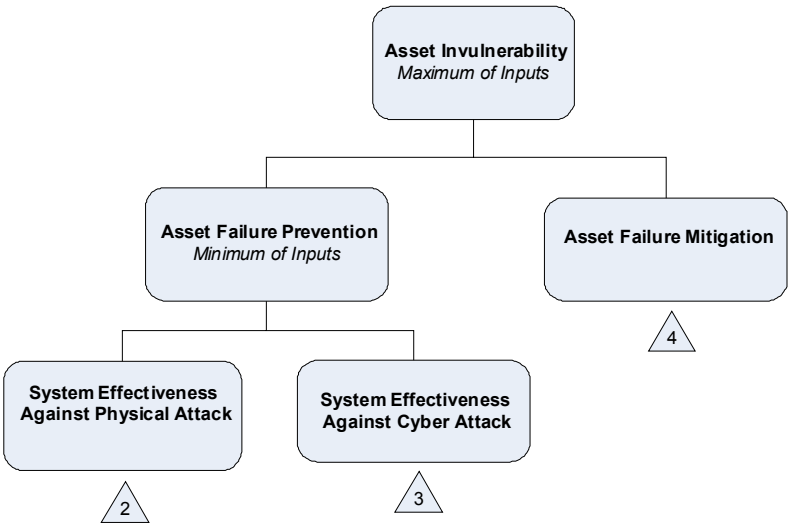

Figure 33. Top-Level Logic for Asset Invulnerability.

Preventing asset failure by providing physical and cyber protection measures to defeat attacks is generally desirable, but may not be practical in all circumstances. For example, some facilities may have assets located over a large geographic area or in exposed locations that are difficult to protect. Consequently, protecting all of the key assets in some facilities may be very costly or practically impossible. In such circumstances, mitigating the effects of asset failures may be a more cost-effective way of avoiding the CoC than providing a highly effective physical and CPS.

Because either prevention or mitigation of asset failure can ensure that the CoC does not occur, this analysis develops effectiveness measures for both asset failure prevention and asset failure mitigation. The CPSAM methodology then selects the larger of those two measures as the measure of asset invulnerability.

5.6.1 System Effectiveness against Physical Attack

There are a number of established methods that can be used to estimate PPS effectiveness [Garcia 2001, Biringer et al. 1999]. These methods generally involve an assessment of: 1) the likelihood that a response force can interrupt an attack before the adversary completes his attack sequence; and 2) the likelihood that the response force can stop the adversary from proceeding on with the attack sequence. These two factors are termed the probability of interruption (P_I) and the probability of neutralization (P_N).

In the CPSAM approach, established methods are used to estimate the PPS effectiveness against physical attacks through the use of the EASI (Estimate of Adversary Sequence Interruption) model [Bennett 1977] to estimate P_I. The probability of neutralization (P_N) is then estimated from the ratio of the number of adversaries to the number of response force personnel using results derived from many runs of the BATLE code (Brief Adversary Threat Loss Estimator)[Engi and Harlan 1981]. The effectiveness of the PPS (P_E) is then taken to be:

$$P_E = P_I * P_N$$

The selected evaluation tools are first used to estimate the PPS effectiveness against a physical attack with all elements (for example, barriers and alarm systems) assigned their normal performance values.

In order to account for the possibility that the attacker may be able to eliminate or degrade some of the PPS elements through cyber means, an analysis is also run assuming that any cyber-controlled PPS elements are defeated (that is, have zero performance values for detection and delay).. For example, if a lock is cyber-controlled then this proposed methodology assumes that the penetration time for the physical barrier it secures is reduced to zero if a cyber attack is successful. The results of the analysis of PPS effectiveness with cyber-controlled element performance set to zero are coupled with estimates of the difficulty of defeating the CPS elements with a cyber attack to produce the measure of effectiveness of the system against a cyber-enabled physical attack.

The remainder of this report uses the following definitions. The likelihood that the fully functioning PPS will be effective against a physical-only attack is denoted P_{Epo}. The likelihood

that the PPS is effective with its cyber-controlled elements defeated is denoted P_{Epz}. The likelihood that the CPS will be effective against a cyber attack is denoted P_{Ec}.

5.6.1.1 The EASI Model

EASI (Estimate of Adversary Sequence Interruption) is a simple method for evaluating the performance of a physical protection system along a specific path under specific conditions of a threat and system operation. The model computes a probability of adversary interruption from an analysis of the interactions of detection, delay, response and communication. An adversary is considered "interrupted" when a response force arrives to encounter the adversaries and takes actions that force the adversaries to abandon their pursuit of their ultimate objective (at least temporarily) in order to counter the actions of the response force. If the response force successfully defeats the adversary force, the adversary is said to be neutralized and cannot complete their ultimate objective. If the adversary force defeats the response force, the adversary can resume pursuit of their ultimate objective. In this case, the interruption of the adversary was temporary.

To compute the probability of interruption using the EASI model, the analyst first precisely identifies the adversary path or scenario to be evaluated. Each opportunity for detecting the adversary is recorded, along with its corresponding probability of adversary detection P_{Di}. Each detection probability is a point estimate value that represents the aggregate performance of the detector sensing abnormal or unauthorized activities, the transmission of the detector's signal to an alarm assessment point, and the realization by the human in assessing the alarm that an adversary attack is in progress which requires activation of response mechanisms. In addition, the analyst estimates the time required for the adversary to accomplish each step along the path or scenario T_i, and especially the delay times between the identified detection opportunities. The mean time for each step is recorded, along with its standard deviation. The EASI model interprets these values as parameters for a Normal distribution. The analyst characterizes the security response force in terms of the probability of successful communication of the alarm to the response force P_C as well as the time required for the response force to react and interrupt the adversary T_R (again, a mean and standard deviation are required).

The method by which the EASI model computes the probability of interruption is as follows. Each possible detection point is examined separately as though adversary detection occurs exactly at that point. The adversary is detected at exactly that point only if they are not detected at all previous detection opportunities and are successfully detected at that point. To obtain the likelihood of exactly this scenario occurring, one multiplies together the nondetection probabilities for all previous detection opportunities times the detection probability at this point. Now, given that the adversary is detected at this point, interruption requires that the alarm be successfully communicated to the response force and that the remaining adversary task time is greater than the response force reaction time. The analyst provides the alarm communication probability directly. The method convolves the normal distributions for all remaining adversary step times with the response force reaction time to determine the probability that the response force arrives in time to interrupt the adversary. Thus, the probability of interruption given detection at exactly the i^{th} detection opportunity P_{Ii} can be written as:

$$P_{Ii} = \left[\prod_{j=1}^{i-1}(1-P_{Dk})\right] \cdot P_{Di} \cdot P_C \cdot P\left\{\left(T_R - \sum_{k=i}^{n} T_k\right) < 0\right\}$$

Since these computed interruption probabilities satisfy probabilistic conditions of being mutually exclusive, the overall probability of interruption for this path P_I is simply the sum of the interrupttion probabilities at each detection opportunity, or:

$$P_I = \sum_{i=1}^{n}\left(\left[\prod_{j=1}^{i-1}(1-P_{Dk})\right] \cdot P_{Di} \cdot P_C \cdot P\left\{\left(T_R - \sum_{k=i}^{n} T_k\right) < 0\right\}\right)$$

The EASI method can be viewed graphically in terms of a small event tree. Each detection opportunity represents an event tree question that must be decided probabilistically based on the probability of detection or nondetection. At the end of the event tree, one adds two questions to represent the performance of the response force: one to represent the probability that the alarm is successfully communicated to the response force, and one to represent the probability that the response force arrives in time to interrupt the adversary. This probability will vary within the event tree to represent the differences in remaining adversary step times as described above. A representative event tree for a system with three possible detection opportunities is shown in Figure 34.

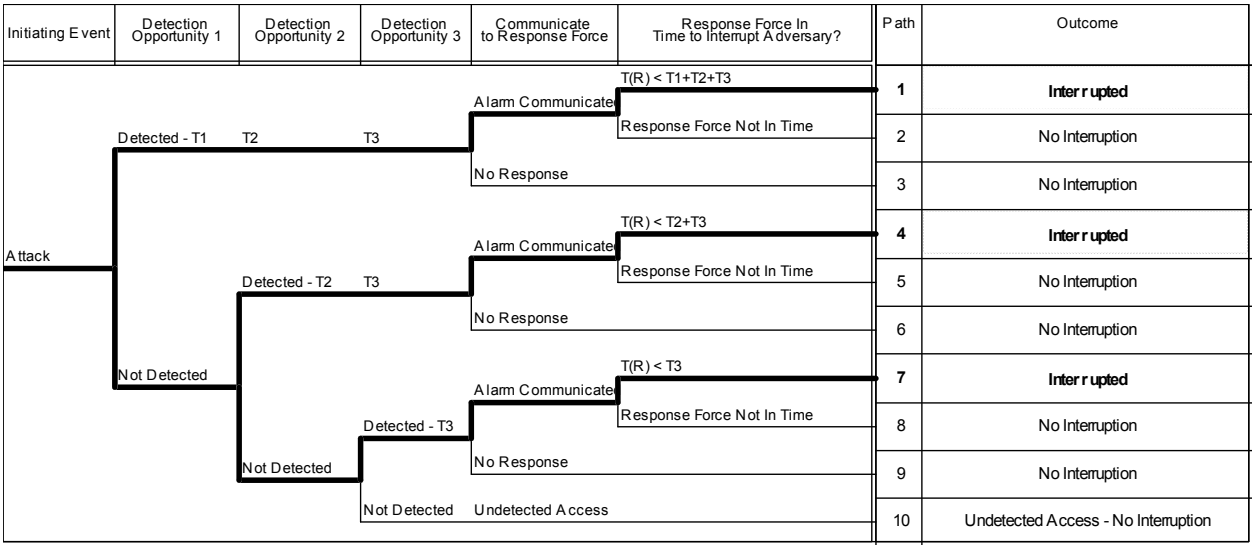

Figure 34. EASI Event Tree

5.6.2 System Effectiveness against Cyber Attack

Cyber security evaluations have, in the past, focused on compliance with industry best practices or red team evaluations. In contrast, this analysis approach has developed a tool for scoring the expected performance of a facility's CPS based on the adversary characteristics and the elements of cyber protection present in the facility. The CPS evaluation model uses information on the attacker's cyber capabilities (e.g., physical access and cyber skills) and the security primitives (such as authentication, Network Access Control (NAC), and User Access Control (UAC)) provided by the cyber protection elements to produce a system effectiveness value for P_{Ec} for each cyber-controlled asset. The quantity ($1-P_{Ec}$) is then a measure of the likelihood that the

attacker could successfully penetrate the CPS and manipulate the target asset from a given CAP in order to cause a CoC. The details of the CPS evaluation model will be given in a future report.

P_{Ec} is computed for attacks that originate offsite (e.g., through the internet) and at any on-site locations from which a commodity delivery asset can be manipulated by cyber means. For on-site locations, these "Cyber Attack Points" (CAPs), might include the central control station, router or switch locations, remote terminal unit locations, communications centers, and points at which access can be gained to the administrative local area network. (NOTE: these "on-site" locations can be located remotely from the commodity delivery asset that can cause the CoC.) The likelihood that the defined threat can cause failure of the commodity delivery asset under analysis from a CAP is denoted $(1 - P_{Ec}(CAP))$. The system effectiveness against physical attacks is estimated for each on-site CAP using the approach described in the previous subsection and is denoted $P_{Ep}(CAP)$. $P_{Ep}(CAP)$ is a measure of the likelihood that the PPS will prevent the adversary from gaining physical access to the CAP. A physical-enabled cyber attack is one in which the adversary gains physical access to a CAP and launches a cyber attack from that location. The likelihood of success of such an attack is the product of the likelihood of gaining physical access to the CAP, which is $(1-P_{Ep}(CAP))$, and the likelihood of successfully mounting a cyber attack from that point, which is $(1-P_{Ec}(CAP))$. A system effectiveness measure, $P_E(CAP_j)$, is then computed for the j^{th} CAP as:

$$P_E(CAP_j) = 1 - (1- P_{Ec}(CAP_j))*(1- P_{Ep}(CAP_j))$$

Figure 35 illustrates the process used to estimate P_{Ec}.

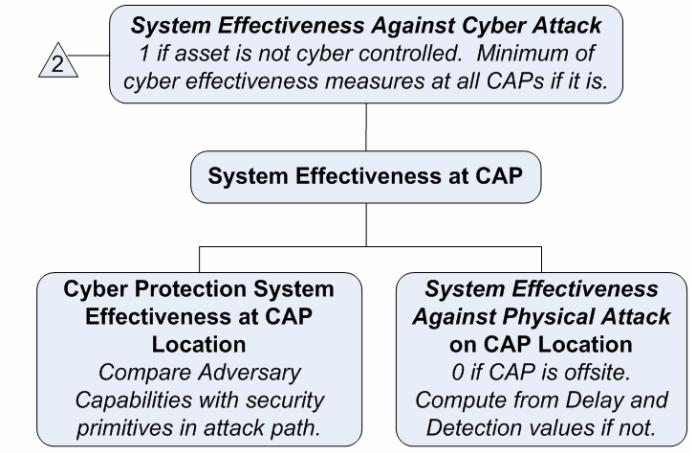

Figure 35. System Effectiveness Against Cyber Attack

The overall system effectiveness measure against cyber attack is the minimum value of $P_E(CAP_j)$ for all CAPs considered. The CAP with the minimum system effectiveness measure is the one for which a combined physical and cyber attack is most likely to produce adversary success. If a facility has a very robust PPS, an offsite location may be the most effective place from which to launch a cyber attack. If it is relatively easy to get to an administrative local area network (LAN) terminal (e.g., one that bypasses the external firewall and other boundary access controls)

but difficult to penetrate further into the facility by physical attack then the "best" attack route may be to carry out a physical attack to get to that LAN terminal, and then launch a cyber attack from that location. If the CPS is highly effective for all CAPs then a physical-only attack may be the most successful strategy. The overall system effectiveness measure against cyber attack (cyber-only attacks and physical-enabled cyber attacks) is the value for the CAP from which the combined physical and cyber attack is in some sense optimal for the adversary.

5.6.2.1 Security Primitives

This section describes various categories for the security primitives proposed by Stamp, et al., in [Stamp and Kilman]. This section provides the foundation for the end goal, which is to use Belief Theory to compare the categories for the security primitives with the categories for the adversary characteristics that were proposed by Duggan [Duggan].

The security technology primitives (which include authentication, NAC, UAC, cryptography, integrity checking, data-aging protection, logging/monitoring/ auditing, and system management [Stamp and Berg] can be used to model the cyber security barriers between cyber locations. This section presents an overview of the security guidelines and best practices for these primitives from the Network Reliability and Interoperability Council (NRIC) [NRIC] and the National Security Agency (NSA) [SNAC].

5.6.2.1.1 Authentication

Authentication assures that a person is who they say they are. Authentication can be implemented through the use of passwords. Authentication has the following three basic elements in a typical commodity delivery infrastructure facility:

- ***Length/type/character set of password:***
  - Use strong passwords [NRIC], which are 12 or more characters in length on Windows™ systems and at least eight characters on a UNIX™ system.
  - Include upper and lowercase letters, numbers, and special characters.
  - Do not use dictionary words.
  - Eliminate passwords found in a list of easily guessed or "cracked" passwords. (NOTE: Administrators should obtain password-guessing programs and run them frequently to identify users who have easily guessed passwords.)
- ***Change frequency***—Change passwords regularly (i.e., every 30 to 90 days).
- ***Limitation of login attempts***—Limit the number of failed password attempts [SNAC].

5.6.2.1.2 Network Access Control

Access control, in the context of network security, is the ability to limit and control access to systems via communication links. To implement access control, it is recommended that a segmented/partitioned network architecture be implemented. More specifically, where practical, it is suggested that user traffic networks, network-management infrastructure networks, customer-transaction system networks, and enterprise communication/business operations networks be separated and partitioned [NRIC].

NAC has the following basic elements in a typical commodity-delivery infrastructure facility:

- ***Traffic filtering device*** (e.g., a firewall). Those services that are not explicitly permitted by the site's security policy are prohibited, and hence blocked at the firewall [SNAC].

- ***Physical separation and physical protection of networks***.

- ***Virtual private networks***. (VPN). Note: This element may overlap with the security primitives for cryptography and authentication.

- ***Authentication***. This element's characteristics may be either identical to or be a subset of those listed for the authentication primitive in Section 5.6.2.1.1. For commodity delivery facilities, one difference may be the presence or absence of two-factor authentication, such as a SecurID Card or other token. Two other issues are user authentication to the network and network administrator authentication to the network devices themselves.

5.6.2.1.3 User Access Control

In terms of UAC, it is good practice to implement "least privilege" rights [NRIC]. Applying "least privilege" limits users' access to only the data and services required to perform their jobs [SNAC].

"Privilege escalation" is an important attack path. It is often easier for an adversary to compromise a user account and then obtain root/administrator privileges than it is for that adversary to directly compromise an administrator account. (NOTE: This path applies to both user accounts and general-purpose accounts such as network printers.)

UAC contains the following elements:

- ***Physical protection of the system***

- ***Access control policy***—How are rights are administered?

- ***Type of software*** used to implement access control.

5.6.2.1.4 Cryptography

Cryptography protects the confidentiality of data by encoding the data so that only authorized parties can read the data.

Cryptography contains the following elements:

- ***Algorithms***—Use industry-accepted algorithms (e.g., 3DES or AES) [NRIC]. Many non-standard cryptography algorithms and key lengths do not protect the confidentiality of data. Another issue is the correctness of the software implementation. Software from reputable security vendors or well-vetted open sources is often preferable to home-grown software because it has been better tested and proven. Weakness in software components, such as the random number generators, can drastically reduce the "work factor" associated with a given software implementation of a putatively strong algorithm.

- ***Key lengths***—Use only industry-accepted key lengths. The minimum recommended key length for a given algorithm may change every year based on the advances (driven by Moore's Law) in easily obtainable yet powerful computing hardware.

- ***Key management procedures***—An adversary typically attacks the key distribution and key storage processes rather than the underlying cryptography algorithm. For example, an adversary with high physical access might steal a user's password. Similarly, a well-funded adversary might subvert an insider in order to obtain copies of the keying material.

5.6.2.1.5 Integrity Checking

Integrity checking provides protection against data modification. Integrity checking is performed for:

- ***System files***—Check the integrity of system files that are susceptible to malevolent modification [NRIC]. Integrity checking will not prevent data modification, but will detect it.

- ***Incoming messages***—Just because a message says it is from a trusted source does not mean it was written/sent by that trusted source.

- ***System hardware***—Perform Power-on Self Tests (POSTs) and Built-In Tests (BITs).

- ***Executing commands***—Check for "faithful execution." Of these four elements, "faithful execution" is likely not implemented in commodity delivery infrastructures because it is currently implemented only in high-assurance products used in government security applications.

5.6.2.1.6 Data-Aging Protection

Even if the integrity of messages is checked, adversaries can still replay previous messages. Data aging contains the following elements:

- ***Timestamps***—A timestamp or an incrementing counter should be used to mitigate replay attacks. The accuracy of this timestamp needs to be ensured because an inaccurate timestamp provides no mitigation [NRIC].

- ***Network-wide time synchronization***—In process control systems, the absolute time of a given event's occurrence may be a crucial element in a realtime control algorithm. In contrast, incrementing counters may suffice for data-aging purposes in non-realtime systems.)

5.6.2.1.7 Logging/Monitoring/Auditing

Logging, monitoring, and auditing are needed to observe events and respond to the activities of the network. Logging/monitoring/auditing contains the following three elements:

- ***Event recording***—All security-related events should be automatically documented in logs [NRIC] by the system. However, automated logging or documenting security-related events provide no real benefit if no one views the documentation.

- ***Event assessment***—A trained individual should view the documentation on a regular basis, searching for anomalies.

- ***Response***—If an alarm or an anomaly is found in the documentation, a "response" must occur in a timely manner. If there is no meaningful and timely response, then event recording and event assessment may be pointless exercises. The lack of response is an important vulnerability in cyber systems. (NOTE: An example of a response would be to change the filtering rules at the firewall.)

5.6.2.1.8 System Management

System management deals with the maintenance of computer systems on the network. System management contains the following elements and associated best practices:

- ***Backups***—Perform current or implement new system backup procedures [SNAC]. This backup should be stored on another system. Also, when feasible, the system on which the backup is stored should not be co-located with the other system.

- ***Malware protection***—Deploy malware protection tools in a manner in which signatures are kept current [NRIC]. A malware protection tool with old signatures will provide little or no protection against new malware.

- ***Configuration management***—Keep operating systems and applications current through updates or patches. Also, disable all unnecessary services. Potentially vulnerable

services (e.g., Network Time Protocol (NTP), Remote Procedure Calls (RPC), Finger, RSH-type commands, etc.) should be disabled, if unneeded, or provided with additional protection [NRIC].

- ***Security testing***—Test the security of all of the devices (clients, servers, switches, routers, firewalls and intrusion detection systems) on the network periodically and after any major configuration changes on the network [SNAC].

5.6.2.2 Security Primitive Effectiveness

Best practices and guidelines provide a standard for measuring each primitive. These were the basis for a CPEI calculation using "elicitation tables" to sort a particular installation's CPS into one of several "Cyber Security Posture" (CSP) categories for each security primitive. Tables 5 through 11 show examples of the elicitation tables. Instead of a purely subjective categorization of CSPs from I (weak) to V (strong), the tables use policies and features to quantify CSPs.

Belief Theory is then used to compare these categories for the CSPs with the adversary characteristics proposed by Duggan. The end goal of this process is to quantify the "belief" that a given adversary category can defeat a given CSP category for a given security primitive.

Table 5. Authentication

Posture Category	Cyber Security Posture
I	No passwords.
II	Weak passwords. No periodic changes.
III	Strong passwords. No periodic changes.
IV	Strong passwords. Periodic changes.
V	Strong passwords. Periodic changes. Limits on failed password attempts. Passwords are "cracked" every month to find users who choose easily guessed passwords.

Table 6. Network Access Control

Posture Category	Cyber Security Posture
I	Remote login via password-protected dial-up connections. No firewall.
II	Remote logins allowed from Internet. IP address filtering and port blocking.
III	Remote logins allowed via VPN connection.
IV	No remote logins. SCADA controls are accessible only from LAN-connected terminals.
V	No remote logins. SCADA LAN is physically separate from other LANs.

The NAC elicitation table assumes that the VPN uses an SSL connection but does not require token-based authentication. Similarly, the Authentication elicitation table assumes that token-based access is not used for user-to-host or administrator-to-network device authentication.

Table 7. User Access Control

Posture Category	Cyber Security Posture
I	Physical access unmonitored. Rights given to everyone.
II	Physical access monitored. Rights assigned to individual users.
III	Physical access monitored. Rights assigned to groups. All cyber equipment is physically secured.

In the UAC elicitation table, "physically secured" means that key system hardware is either in a locked enclosure or has certain input/output (I/O) devices disabled. Locking or disabling hardware helps prevent simple privilege escalation attacks such as booting from a floppy disk or compact disk (CD). The difference between rights assigned to "individual users" vs. "groups" is the notion of "roles-based access control (RBAC)." With RBAC, essential system roles are clearly defined. Each user is then assigned one, or more, system roles. Each role is given the "least privilege" necessary to accomplish its job function. This approach makes it easier to audit user privileges than assign system rights on a per-individual basis.

Table 8. Cryptography

Posture Category	Cyber Security Posture
I	Plaintext communications over wireless channel.
II	Plaintext communications over Public Internet or PSTN.
III	Encrypted communications over a VPN using non-industry-accepted algorithms and key lengths. Non-standard key management practices are used.
IV	Encrypted communications over a VPN using either non-industry-accepted algorithms or key lengths. Industry-standard key management practices are used.
V	Encrypted communications over a VPN using industry-accepted algorithms and key lengths. Industry-standard key management practices are used.

Table 9. Integrity Checking

Posture Category	Cyber Security Posture
I	No integrity checking.
II	Integrity checking for incoming messages. Hardware POST.
III	Integrity checking of incoming messages and all susceptible system files. Hardware BIT.

Table 10. Data Aging Protection

Posture Category	Cyber Security Posture
I	No data-aging protection.
II	Messages are time-stamped.
III	Messages are accurately time-stamped.

Table 11. Logging/Monitoring/Auditing

Posture Category	Cyber Security Posture
I	No logging or monitoring
II	Logging, but logs are not viewed
III	Some security-related events are logged. Untrained individual monitors logs.
IV	Some security-related events are logged. Trained individual monitors logs.
V	All security-related events are logged. Trained individual monitors logs. Able to respond.

(NOTE: Categories I-IV are functionally equivalent for this security primitive since detection and assessment without a response is typically useless.)

Table 12. System Management

Category	Cyber Security Posture
I	No system management.
II	Performing backups. Keeping OS/Applications current through patching.
III	Current malware protection tools. Performing backups. Keeping operating system (OS)/ Applications current through patching. Disabling unnecessary services. Performing periodic security testing.

5.6.2.3 Comparison of Adversary Capabilities with Security Primitives

The adversary's capabilities (against each security primitive) must be compared with the instantiation of that security primitive on a given virtual link. In each table in this section, each variable (for which degrees of evidence are assigned) can be physically interpreted as: "the probability (subjective, frequency) that the adversary with the stated capabilities can defeat the associated security primitive of the indicated level".

If the intervals to which degrees of evidence are assigned are points, then the belief measure is a probability measure. If only one point has a degree of evidence of 1.0 then there is no uncertainty in that belief.

Within each box in a given table, the notation is that a given interval [*a,b*) has a degree of evidence value of *X*; we can then calculate belief and plausibility from the degrees of evidence. The "[" and "(" symbols have the standard meaning in set theory.

5.6.2.3.1 Authentication

In Table 13, the level of physical access is a main discriminating factor. If the adversary has high or medium access and a long implementation time, then a subverted insider is likely to disclose their own password or find another user's password. Hence, Category I through III adversaries are hard to deter with password-based authentication systems. The cyber skills are a secondary discriminating factor between Category IV through VI adversaries. A high level of cyber skills is required to break strong passwords that are changed infrequently. Finally, all adversary levels can break weak passwords that seldom change.

Table 13. Comparison of Adversary Capabilities with Authentication Security Primitive

Authentication Category	Threat Category					
	I	II	III	IV	V	VI
I No passwords	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1
II Weak passwords. No periodic changes.	[1] 1	[1] 1	[1] 1	[1] 1	[0.9,1] 1	[0.8,1] 1
III Strong passwords. No periodic changes.	[1] 1	[0.7, 1) 0.1 [1] 0.9	[0.7, 1) 0.2 [1] 0.8	[0.7, 1) 0.2 [1] 0.8	[0.7, 1) 0.4 [1] 0.6	[0,0.3) 0.8 [0.3,0.7) 0.1 [0.7,1.0] 0.1
IV Strong passwords. Periodic changes.	[1] 1	[0.7, 1) 0.3 [1] 0.7	[0,0.3) 0.3 [0.3,0.7) 0.4 [0.7,1.0] 0.3	[0] 0.5 (0,0.3] 0.5	[0] 0.7 (0,0.3] 0.3	[0] 0.9 (0,0.3] 0.1
V Strong passwords. Periodic changes. Limits on failed password attempts.	[1] 1	[0.7,1.0) 0.5 [1] 0.5	[0,0.3) 0.6 [0.3,0.7] 0.4	[0] 0.9 (0, 0.3] 0.1	[0] 0.9 (0, 0.3] 0.1	[0] 1

This version of Table 13 was generated as follows:

1) Threat Category I (with high cyber skills and physical access) is believed to always win against a commodity infrastructure's cyber-security posture.

2) All threat categories are believed to win against the lowest two categories (no passwords and weak, unchanging passwords) for the Authentication security primitive. However, Threat Categories V and VI might occasionally fail because they have low physical access and/or low cyber skills. They also have a limited implementation time (e.g., months).

3) The Threat Categories IV, V, and VI typically fail against Authentication Category V (strong passwords that change frequently) because their limited implementation time (months) makes a brute-force, password-cracking attack unlikely even if they have good

cyber skills. In addition, their low level of physical access also makes password-stealing unlikely.

4) The other boxes (e.g., the ones in the middle of the table) can be viewed as follows:

   - The interval [0] can be viewed as a "really low" likelihood of success.
   - The interval [0,0.3) can be viewed as "low."
   - The interval [0.3,0.7) can be viewed as "medium."
   - The interval [0.7,1.0) can be viewed as "high."
   - The interval [1] can be viewed as a "really high" likelihood of success.
   - The evidence assigned to each interval is then a *subjective* comparison of the Threat Categories' physical access, cyber skills, and implementation time with each Authentication Category's attributes for password length and change frequency. Varying degrees of evidence were thereby assigned to each of the intervals described above.

5.6.2.3.2 Network Access Control

In Table 14, cyber skills are the primary method for defeating Authentication Categories II and III. If no remote logins are allowed, then physical access becomes the primary determinant. For Category I, the main problem is finding the dial-up modem's phone numbers, which can be solved via a mix of cyber skills and physical access.

This security primitive is not based on work-factor. As such, the adversary's chance of success is based on more qualitative factors such as:

a) Have exploits for a particular device (e.g., a firewall) existed in the past?

b) What is the likelihood that an adversary can either obtain an existing exploit or craft a new "zero day" exploit for a given device?

c) What is the likelihood that a given device will be misconfigured to allow a given known exploit?

d) What is the likelihood that an adversary can use their physical access to subvert an insider to introduce an exploitable vulnerability?

Table 14. Comparison of Adversary Capabilities with Network Access Control Security Primitive

NAC Category	Threat Category					
	I	**II**	**III**	**IV**	**V**	**VI**
I Password-protected dial-up. No firewall.	[1] 1	[1] 1	[1] 1	[1] 1	[0.7,1] 1	[0.3, 0.7) 0.5 [0.7,1] 0.5
II Remote login from Internet. Firewall.	[1] 1	[0.3, 0.7) 0.2 [0.7, 1.0] 0.8	[0.3, 0.7) 0.5 [0.7, 1.0] 0.5	[0.3, 0.7) 0.2 [0.7, 1.0] 0.8	[0.3, 0.7) 0.5 [0.7, 1.0] 0.5	[0, 0.3) 0.8 [0.3, 0.7] 0.2
III Remote logins via VPN.	[1] 1	[0, 0.3) 0.5 [0.3, 0.7] 0.5	[0, 0.3) 0.8 [0.3, 0.7] 0.2	[0.3, 0.7) 0.8 [0.7, 1.0] 0.2	[0, 0.3) 0.8 [0.3, 0.7] 0.2	[0] 1
IV No remote logins. SCADA net not physically isolated from other LANs.	[1] 1	[0.3, 0.7) 0.2 [0.7, 1.0] 0.8	[0.3, 0.7) 0.8 [0.7, 1.0] 0.2	[0] 0.6 (0, 0.3] 0.4	[0] 0.8 (0, 0.3] 0.2	[0] 1
V No remote logins. SCADA LAN physically isolated from other LANs.	[1] 1	[0, 0.3) 0.5 [0.3, 0.7] 0.5	[0, 0.3) 0.8 [0.3, 0.7] 0.2	[0] 0.8 (0, 0.3] 0.2	[0] 0.9 (0, 0.3] 0.1	[0] 1

This version of Table 14 was generated as follows:

1) The belief that an adversary can succeed against NAC Category I is predicated on the ability to obtain the phone number(s) for the dial-up modems by either cyber or physical means. (NOTE: The strength of the password protection would be evaluated via the Authentication security primitive.)
2) Threat Category VI is unlikely to win against the higher NAC categories but has some chance of finding an existing vulnerability in a firewall to defeat Authentication Category II. The other threat categories have evidence based on a combination of their cyber skills and their implementation time.
3) NAC Category III is likely safe against adversaries without high cyber skills since an attack would likely exploit an implementation flaw in the VPN software. (Again, the strength of the authentication would be evaluated by that security primitive.) Implementation time is again a secondary indicator.
4) NAC Category IV is likely safe against an adversary without medium or high physical access. Cyber skills and implementation time become secondary factors.
5) NAC Category V is likely safe against an adversary without high physical access. Cyber skills and implementation time are secondary factors in how close to "0" the evidence is that an adversary (without high physical access) could exploit this NAC category.

5.6.2.3.3 User Access Control

Table 15 assumes that anyone with medium or high physical access and medium or high cyber skills has some likelihood of defeating this security primitive, even if the equipment is physically secured.

Table 15. Comparison of Adversary Capabilities with User Access Control Security Primitive

UAC Category	Threat Category					
	I	**II**	**III**	**IV**	**V**	**VI**
I Physical access unmonitored. Rights given to everyone.	[1] 1	[0.7,1] 1	[0.3, 0.7) 0.2 [0.7, 1.0] 0.8	[0.3, 0.7) 0.5 [0.7, 1.0] 0.5	[0.3, 0.7) 0.8 [0.7, 1.0] 0.2	[0, 0.3) 0.8 [0.3, 0.7] 0.2
II Physical access monitored. Rights given to individuals.	[1] 1	[0.3, 0.7) 0.2 [0.7, 1.0] 0.8	[0.3, 0.7) 0.5 [0.7, 1.0] 0.5	[0, 0.3) 0.8 [0.3, 0.7] 0.2	[0, 0.3] 1	[0] 0.8 (0,0.3] 0.2
III Rights given to groups. All equipment is physically secured.	[1] 1	[0.3, 0.7) 0.5 [0.7, 1.0] 0.5	[0, 0.3) 0.8 [0.3, 0.7] 0.2	[0] 0.8 (0,0.3] 0.2	[0] 0.9 (0,0.3] 0.1	[0] 1

This version of Table 15 was generated as follows:

1) Threat Category VI is unlikely to win if physical or cyber access is controlled. However, adversaries have some chance against UAC Category I.

2) The performance of the other Threat Categories against UAC Category I is based on the combination of physical access with cyber access and implementation time being secondary factors.

3) The performance against UAC Categories II and III then more heavily weights the evidence towards adversaries who have medium/high physical access and medium/high cyber skills.

5.6.2.3.4 Cryptography

The cryptography security primitive is also based on work factor like the authentication security primitive. Physical access also plays some role since it can be used to attack the key distribution system. In general, work factor based attacks are more difficult against cryptosystems then they are against password-based authentication. However, the difficulty of stealing keying material may often be comparable to that of stealing passwords. (Table 16)

Table 16. Comparison of Adversary Capabilities with Cryptography Security Primitive

Cryptography Category	Threat Category					
	I	II	III	IV	V	VI
I Plaintext over wireless	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1	[0.7,1] 1
II Plaintext over Internet PSTN	[1] 1	[0.7,1] 1	[0.3,7) 0.5 [0.7,1] 0.5	[0.7,1] 1	[0.3,7) 0.8 [0.7,1] 0.2	[0,0.3] 1
III VPN with non-standard algorithms, key lengths and key management	[1] 1	[0.3,7) 0.5 [0.7,1] 0.5	[0, 0.3) 0.5 [0.3, 0.7] 0.5	[0.3,7) 0.5 [0.7,1] 0.5	[0, 0.3) 0.8 [0.3, 0.7] 0.2	[0] 1
IV VPN with either non-std algorithms or key lengths. Industry-standard key management	[1] 1	[0, 0.3) 0.8 [0.3, 0.7] 0.2	[0] 0.5 [0, 0.3] 0.5	[0, 0.3) 0.8 [0.3, 0.7] 0.2	[0] 0.8 [0,0.3] 0.2	[0] 1
V VPN with industry-accepted algorithms, key lengths and key management	[1] 1	[0] 0.5 [0, 0.3] 0.5	[0] 0.8 [0,0.3] 0.2	[0] 0.5 [0, 0.3] 0.5	[0] 1	[0] 1

This version of Table 16 was generated as follows:

1) Plaintext over wireless channels is typically accessible to everyone. So, Cryptography Category I is basically useless as a security measure. However, adversaries with low cyber skills might not be able to obtain the requisite radio hardware.

2) Installing a sniffer program in the public Internet or the PSTN takes a fairly high level of either cyber skills or physical access to the ISP or Telco equipment. Implementation time is a secondary factor. The scores for Threat Categories II through V reflect that.

3) Threat Category I always wins because (even against industry-standard practices) they can eventually steal keying material or subvert an insider.

4) The values for Threat Categories III through V against the various VPN levels need further study. However, many "non-standard" aspects of crypto have been successfully attacked. Examples include the original cryptography for NetScape, WEP for IEEE 802.11b networks, and DVDs.

5.6.2.3.5 Integrity Checking

This section assumes that best-practice cryptographic means are used to provide integrity checking of messages and files. So, adversary attacks against integrity checks for messages and files are similar to those against the cryptography security primitive.

The BIT and POST functions for the process-control hardware are also assumed to be industry-standard. So, they will help protect against adversarial attacks against in-situ hardware. They will not protect against life-cycle attacks that subvert the hardware before installation into the commodity delivery infrastructure's networks.

Table 17. Comparison of Adversary Capabilities with Integrity Checking Security Primitive

Integrity Checking Category	Threat Category					
	I	II	III	IV	V	VI
I No integrity checking.	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1
II Integrity checking for incoming messages. Hardware POST.	[1] 1	[0, 0.3) 0.5 [0.3, 0.7] 0.5	[0, 0.3) 0.8 [0.3, 0.7] 0.2	[0.3, 0.7) 0.8 [0.7, 1.0] 0.2	[0, 0.3) 0.8 [0.3, 0.7] 0.2	[0] 1
III Integrity checking of incoming messages & system files. Hardware BIT.	[1] 1	[0.3, 0.7) 0.2 [0.7, 1.0] 0.8	[0.3, 0.7) 0.8 [0.7, 1.0] 0.2	[0] 0.6 (0, 0.3] 0.4	[0] 0.8 (0, 0.3] 0.2	[0] 1

This version of Table 17 was generated as follows:

1) Every adversary class wins if there is no integrity checking.

2) The values for Integrity Checking Category II were set equal to those for NAC Category III. In both cases, the security primitive is likely safe against adversaries without high cyber skills since an attack would likely exploit an implementation flaw in the integrity checking software or algorithm. Implementation time is a secondary indicator.

3) The values for Integrity Checking Category III were set equal to those for NAC Category IV. In both cases, the security primitive is likely safe against an adversary without medium or high physical access. (For example, an adversary would corrupt system files via a trusted-user account such as an admin account. They might corrupt system hardware during their role as a trusted system installer or maintainer.) Cyber skills and implementation time become secondary factors.

5.6.2.3.6 Data Aging

This security primitive is not based on work-factor. As such, the adversary's chance of success is based on more qualitative factors such as:

- Knowledge of the existing time offsets between the process control system elements and the control center.
- The ability to change system clock settings at a given device in order to enable a particular attack.

Table 18. Comparison of Adversary Capabilities with Data Aging Security Primitive

Data Aging Category	Threat Category					
	I	II	III	IV	V	VI
I None	[0.7, 1.0] 1	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1
II Time Stamps	[1] 1	[0.7,1] 1	[0.3, 0.7) 0.2 [0.7, 1.0] 0.8	[0.3, 0.7) 0.5 [0.7, 1.0] 0.5	[0.3, 0.7) 0.8 [0.7, 1.0] 0.2	[0, 0.3) 0.8 [0.3, 0.7] 0.2
III Accurate Time Stamps	[1] 1	[0.7,1] 1	[0.3,7) 0.5 [0.7,1] 0.5	[0.7,1] 1	[0.3,7) 0.8 [0.7,1] 0.2	[0] 1

This version of Table 18 was generated as follows:

1) If there is no data aging protection then any level of adversary can replay messages at will if they can find the technical documentation for the message formats. Given that SCADA/PCS protocol formats are widely available in textbooks and on the Web, this is a minimal bar. However, adversaries with low cyber skills do have some chance of failure.

2) If time stamps are used then the adversary must also be able to eavesdrop on the system to determine the time offsets between a particular piece of process control system hardware and the control center. This requires medium to high cyber skills or medium to high physical access. So, the performance of the Threat Categories against Data Aging Category II is based on the combination of physical access with cyber skills and implementation time being a secondary factor. (NOTE: This row is set identically with UAC Category I.)

3) If accurate time stamps are used then the adversary must be able to access the system via cyber or physical means to change system settings. This often requires high cyber skills or high physical access. However, implementation time is again a secondary factor that allows Threat Category II to often succeed.

Another attack against data aging would be to alter the contents of in-transit messages to stage denial-of-service attacks. Protection against message alteration is handled by the Cryptography security primitive.

5.6.2.3.7 Logging/Monitoring/Auditing (LMA)

This security primitive is not based on work-factor. As such, the adversary's chance of success is based on more qualitative factors, which include:

- What is the probability that the adversaries' exploit will be logged?
- What is the probability that the logged data will be recognized as an exploit?
- Is there a meaningful and timely response to a logged and assessed event?

Table 19. Comparison of Adversary Capabilities with Logging/Monitoring/Auditing (LMA) Security Primitive

LMA Category	Threat Category					
	I	II	III	IV	V	VI
I None	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1
II Logging, but logs not viewed	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1
III Someone watches logs	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1
IV Trained person watches logs	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1
V Trained person responds in a timely manner	[1] 1	[0.3, 0.7) 0.5 [0.7, 1.0] 0.5	[0, 0.3) 0.8 [0.3, 0.7] 0.2	[0] 0.8 (0,0.3] 0.2	[0] 0.9 (0,0.3] 0.1	[0] 1

This version of Table 19 was generated as follows:

1) If there are no logs, or the logs are not viewed, then the adversary always wins. The adversary also always wins if there is no response.

2) The belief that an adversary can deceive a trained person is fairly subjective and based on their combination of cyber skills and physical access. If the adversaries have high cyber skills, then they will use an exploit that will not be observable in the logs. If they have high physical access, then they will bribe the monitor to simply not respond. Implementation time is a secondary factor. (NOTE: The current belief values for Category V are identical to those for Category III of the UAC security primitive in Table 15, since that row is also a subjective mix of physical access and cyber skills. Those values require further study.)

5.6.2.3.8 System Management

This security primitive is not based on work-factor. As such, the adversary's chance of success is based on more qualitative factors that are quite similar to those for the NAC security primitive. The qualitative factors include:

- Have exploits for a particular application or operating system (OS) existed in the past?
- What is the likelihood that an adversary can either obtain an existing exploit or craft a new "zero day" exploit for a given application or OS?

- What is the likelihood that a given application or OS will be misconfigured to allow a given known exploit?

- What is the likelihood that an adversary can use their physical access to subvert an insider to introduce an exploitable vulnerability?

Table 20. Comparison of Adversary Capabilities with System Management Security Primitive

Systems Management Category	Threat Category					
	I	II	III	IV	V	VI
I No system management	[1] 1	[1] 1	[1] 1	[1] 1	[1] 1	[0.7,1] 1
II Backups, current OS/Apps thru patching.	[1] 1	[0.3, 0.7) 0.2 [0.7, 1.0] 0.8	[0.3, 0.7) 0.5 [0.7, 1.0] 0.5	[0.3, 0.7) 0.2 [0.7, 1.0] 0.8	[0.3, 0.7) 0.5 [0.7, 1.0] 0.5	[0, 0.3) 0.8 [0.3, 0.7] 0.2
III Current malware protection tools & OS/ applications, backups, unnecessary services disabled, security testing.	[1] 1	[0, 0.3) 0.5 [0.3, 0.7] 0.5	[0, 0.3) 0.8 [0.3, 0.7] 0.2	[0.3, 0.7) 0.8 [0.7, 1.0] 0.2	[0, 0.3) 0.8 [0.3, 0.7] 0.2	[0] 1

This version of Table 20 was generated as follows:

1) Every threat category is likely to win against Category I (no systems management). However, threat Category IV may be unable to find an exploit in a timely fashion. (NOTE: This is identical to the belief that an adversary can defeat "plaintext over wireless" in Table 16.)

2) For Category II (current OS through patching), the belief is predicated on the adversary having sufficient cyber skills to use a new zero-day exploit for an OS before it is patched by the system administrator. However, they need to find an existing exploit for one or more applications since the site is not current on malware protection. So, this row was set equal to NAC Category II ("Remote login from Internet are allowed, but a firewall is in place.")

3) For Category III (current OS through patching and malware protection with current signatures), the belief is predicated on the adversary having sufficient cyber skills to use a new zero-day exploit for an OS or an application before it is patched by the system administrator. Alternately, the adversary might have sufficient cyber skills and implementation time to craft new zero-day exploits. Finally, they might use their physical access and implementation time to inject vulnerabilities into the system. One example is subverting the system administrators. So, this row was set equal to NAC Category III ("remote logins via VPN").

5.6.2.4 Example Network for Typical Commercial Installations

Figure 36 shows a simple example network where not all links implement all three security primitives which may be a typical case for commercial installations. For example, the links 1-2 and 1-3 might be links from an external CAP (Node 1) to an internal network (Node 2) and a business partner (Node 3) since those two links implement the Authentication and NAC security primitives. The link from the business partner to the internal network (3-2) might have NAC and UAC, but weaker Authentication requirements than the direct link from the external CAP. Node 4 might then be a host on the internal network that can cause a CoC. From a trusted network (e.g., Node 2) only Authentication and UAC are implemented; NAC is added for connections that come directly from an un-trusted network (e.g., Node 3). So, this example is a simple example of how weaker security at a business partner can weaken the CSP for a commodity delivery infrastructure.

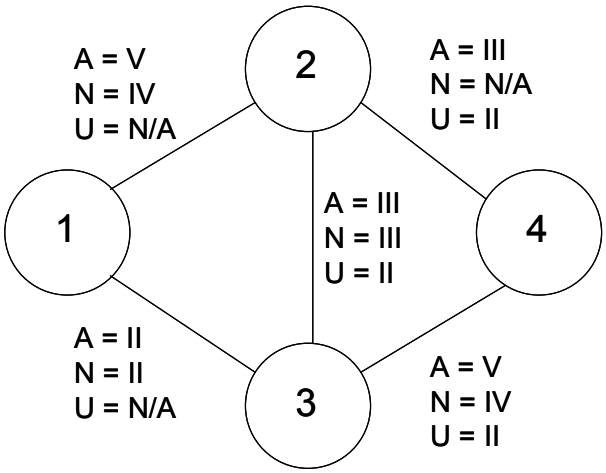

Figure 36. Example Network

Table 21 shows the evidence that this network's cyber security posture will be effective against the defined threat categories. This simple example is qualitatively correct since it shows that the network would likely not withstand a highly skilled adversary (e.g., Threat Category I). However, it would be reasonably effective against lower-level adversaries (e.g., Threat Categories IV-VI) who have poor physical access. For the medium-skilled adversaries, the proposed analysis technique allows an analyst to play "what if" games that complement "best practices reviews" and red-teaming exercises. The values in Table 21 can be calculated by convolution of terms or by combining the expected values for terms if the variables are noninteracting (independent) variables. An example of the latter technique is as follows. The expected value interval for adversary success for each security primitive can be calculated using Equation 3 in Appendix A with the data from Table 6, Table 7, and Table 8. The expected value interval for adversary success for each path segment is the product of the expected value intervals of all the security primitives on that segment. The expected value interval for adversary success for a path is then the product of the expected value intervals for each path segment. System effectiveness for a path (the probability that the adversary is detected) is one minus the expected value interval of the overall probability of adversary success. The path with the lowest midpoint value of its expected value interval for system effectiveness is then deemed the "weakest path".

Using the data from Table 5 in Equation 3, the lower expected value for A=V under Threat Category III is: 0.6*0 + 0.4*0.3 = 0.12. The upper expected value for A=V under Threat Category III is: 0.6*0.3 + 0.4*0.7 = 0.46. The expected value interval is thus [0.12, 0.46].

For path segment 1-2 under Threat Category III, the expected value interval for adversary success is (A=V)*(N=IV) = [0.12, 0.46]*[0.38, 0.76] = [0.046, 0.35]. Similarly, for path segment 2-4 the expected value interval for adversary success is (A=III)*(U=II) = [0.94, 1] * [0.5, 0.85] = [0.47, 0.85]. The expected value interval for adversary success for path 1-2-4 under threat category III is [0.046, 0.35]* [0.47, 0.85] = [0.021, 0.30]. The system effectiveness is 1 - [0.021, 0.30] = [0.7, 0.98] as indicated in Table 21.

Table 21. CPS Effectiveness

Cyber Threat Category	CPS Effectiveness Interval	Easiest Attack Path
I	[0]	(1,3,4)
II	[0.12, 0.68]	(1,2,4)
III	[0.7, 0.98]	(1,2,4)
IV	[0.9, 1.0]	(1,3,2,4)
V	[0.97, 1.0]	(1,3,2,4)
VI	[1]	No Possible Path

This simple example shows the tradeoffs between business necessity (e.g., the business partner may need access to the internal network for maintenance reasons) and security. The lower-level adversaries need to enter through the business partner's network (1-3-2-4). For the mid-level adversaries, they can defeat the protections for the service provider's business network (1-2-4). Finally, the highest-level adversary can penetrate the internal network directly after the subverting the business partner's network (1-3-4).

5.6.3 System Effectiveness Against Cyber-Enabled Physical Attack

Prior security evaluation methodologies have assessed the effectiveness of cyber and physical security systems separately. In order to quantify the effectiveness of a security system against blended cyber/physical attacks, it was necessary to develop a mathematical basis on which such quantification could be based. This project considered two different approaches [Gordon and Wyss 2005]. An "Expected Value Approach" involved applying weighting factors to the individual component delay times and detection probabilities based on the likelihood that a cyber attack could disable the function for that component. These weighted detection and delay values were then used in a traditional physical security analysis to estimate a weighted system effectiveness. While this method was simple, it was found to exhibit a number of undesirable characteristics [Gordon and Wyss 2005], and was abandoned in favor of the "Bounding Analysis Approach".

The Bounding Analysis Approach involved performing multiple separate physical security analyses in which individual components were presumed to be disabled based on postulated cyber attacks. The likelihood of adversary success in accomplishing the postulated cyber attacks was considered separately and incorporated in a final overall security system effectiveness for each postulated blended attack. While this method was more complex than the Expected Value Approach, it produced robust results that could be readily explained to both analysts and customers. The remainder of this section provides a brief description of this approach. Greater detail regarding the method and comparison can be found in Gordon and Wyss, 2005.

For the cyber-enabled physical attack, both the cyber attack on the cyber-controlled PPS elements and the physical attack on the degraded PPS system must succeed for the attack to succeed. The likelihood that the cyber attack succeeds is (1- P_{Ec}), and the likelihood that the physical attack succeeds against the degraded PPS is (1- P_{Epz}), so the likelihood that both attacks succeed is the product of these two values. The likelihood that the overall system is effective against the cyber-enabled physical attack (P_{Ecp}) is then 1 minus the likelihood that the attack is successful or:

$$P_{Ecp} = 1 - (1- P_{Ec}) * (1- P_{Epz})$$

The overall system effectiveness against physical attacks, P_{Ep}, (which includes cyber-enabled physical attacks if they are possible) is then taken to be the lower of P_{Epo} and P_{Ecp}. Figure 28 illustrates the approach used to estimate P_{Ecp}.

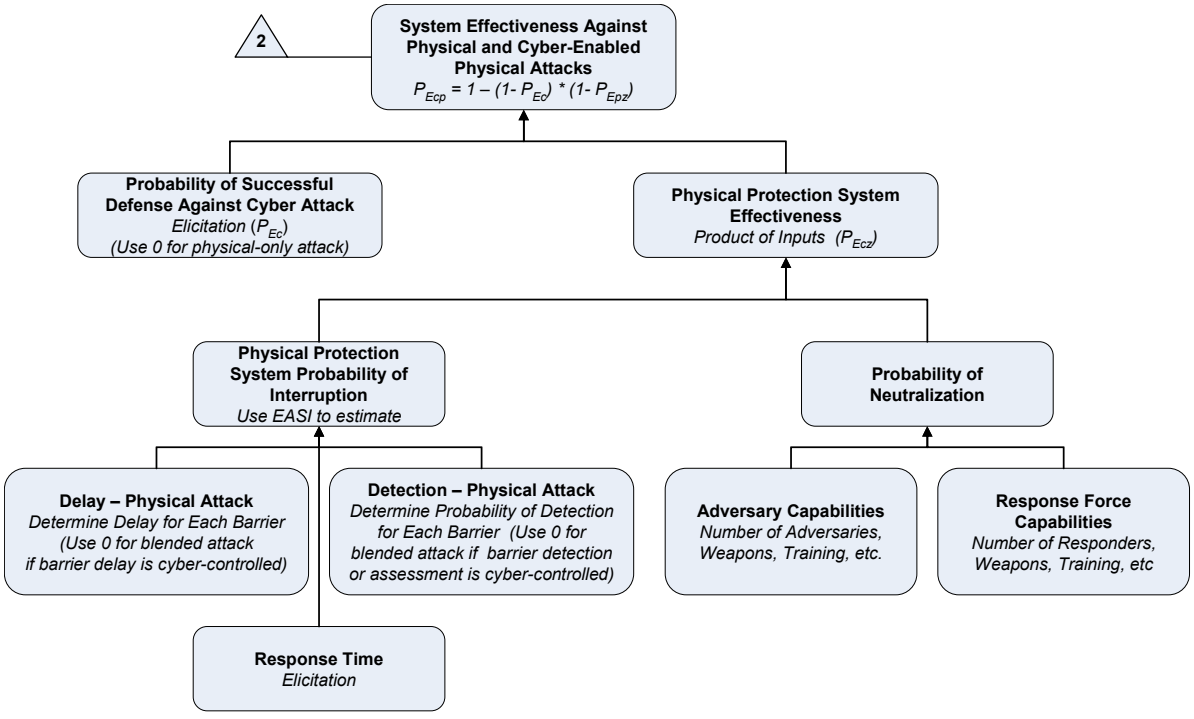

Figure 37. System Effectiveness against a Cyber-Enabled Physical Attack

5.6.4 System Effectiveness Against Physical-Enabled Cyber Attack

The system effectiveness against a physical-enabled cyber attack is a special case where the adversaries are must first succeed with a physical attack to reach the cyber access point within the facility. No special analysis mathematics are needed for this attack type as was the case for the cyber-enabled physical attack (Section 5.6.3). The CPSAM estimates the system effectiveness for all four attack types and displays the results in the data tree, but only the most successful attack is included in the risk calculation.

5.7 Asset Failure Mitigation

The adversary can choose to use any of the attack types discussed above. The measure of asset failure prevention performance is considered to be the minimum of the measures for system effectiveness against physical attacks (P_{Ep}) and the system effectiveness against cyber attacks (P_{Ec}).

Infrastructure facilities will likely score poorly on asset failure prevention when compared with hardened Government facilities. Consequently, their ability to mitigate asset failures is an important part of their protection posture. Figures 38 and 39 illustrate how asset failure mitigation effectiveness is addressed in this proposed methodology. An effective response to asset failures may prevent a given CoC from occurring. If there is sufficient time to either repair or replace the failed assets or implement an alternative mode of operation that bypasses the failed assets then the ultimate consequences of those asset failures may be avoided. In practice, mitigation can be effective if all of the following conditions apply:

- Written procedures are established for performing the mitigation actions.
- Operators and maintenance personnel are trained to carry out the procedures.
- Any spare parts or materials required for the mitigation actions are maintained in a secure location separate from the asset location.

The Asset Failure Mitigation effectiveness is a unit-less quantity, based on the time required to complete the mitigation actions (T_m) and the expected time available from detection of the failure until the CoC is inevitable (T_{me}). If the ratio of T_m to T_{me} is small, then it is likely that the asset failure can be corrected and the CoC avoided. If that ratio is large, then there will be insufficient time to correct the problem caused by the asset failure, and the consequences of concern are likely to occur.

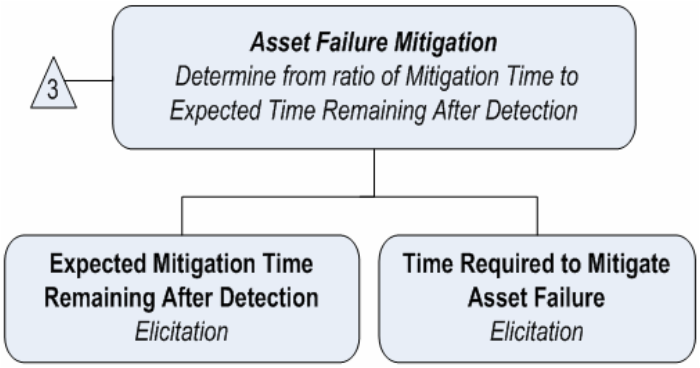

Figure 38. Asset Failure Mitigation

The sigmoid function shown in Figure 39 is one example of how to estimate the mitigation effectiveness measure from the ratio (T_m / T_{me}). The values for this parameter for a given utility could be set via Monte Carlo simulation, game theory, or subject matter expert opinion.

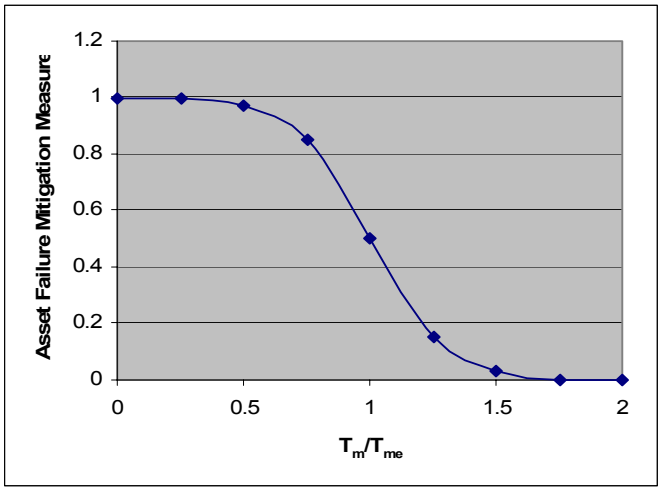

Figure 39. Example of Asset Failure Mitigation Measure

An asset's failure may not be immediately visible to system's operator. If no detectors monitor the asset's performance or if the attacker can interfere with the detection and assessment processes, then some time may pass before the asset failure becomes apparent to the operators in terms of changes in facility performance (e.g., decreased flow in major transmission elements, high temperature in critical components, or complaints from end users). The value selected for T_{me} should take into account the time required to detect the failure – including actions the attacker can take to delay detection. In the simplest formulation of this methodology, the facility operators would provide estimates of T_m and T_{me}.

5.7.1 Advanced Consequence Mitigation Analysis

In the current version of CPSAM, asset failure mitigation is modeled with a simple time function. However, the team recognizes that mitigation is a function of when the adversary actions occur in relation to the onset of consequences. The following describes an advanced

consequence mitigation analysis methodology that was developed, but not included in CPSAM because of the large number of input data requirements.

The EASI method was developed to model high-security facilities. One of its underlying assumptions is that a consequence is inevitable if the adversary force completes all of their tasks. This assumption is true if one is modeling the theft of nuclear material or an explosively induced dispersal of radioactive materials. It is even true in many lower-security facilities if the consequence can be achieved by an attack against a single target location. However, the assumption fails for many highly redundant systems, and especially for systems that are both redundant and physically distributed, such as networked distribution infrastructures (water, electricity, telecommunications, etc.). These infrastructures have been designed to withstand most single failures without causing large consequences in order to ensure that they are robust against natural phenomena such as earthquakes and hurricanes. To accomplish this, two major design features have been used: *reserve capacity* and *redundant distribution networks*. *Reserve capacity* manifests itself in water storage tanks and electrical spinning reserve. *Redundant distribution* is embodied in loop-type distribution architectures for electricity, water, and telecommunications, and redundant geographically diverse point-to-point links in other telecommunications and commodity-distribution applications. The result of these design features is twofold:

- first, the reserve capacity often gives repair crews time to respond to achieved damage states before actual customer consequences occur, and

- second, redundancy often means that an adversary must attack multiple geographically diverse assets almost simultaneously to cause consequences.

Drawing upon the event tree representation of the adversary attack to extend the EASI method can account for consequence achievement and consequence mitigation. For the sake of simplicity in the event tree graph, this discussion neglects the step of communicating with the response force to initiate response as this is generally quite reliable, especially in situations where lightning-fast response is not expected. Instead, we add events to the tree that consider:

- the time required for an adversary to accomplish a sabotage event,

- the likelihood that the adversary can successfully execute the sabotage if given sufficient time to carry out their plan,

- the time delay between the sabotage event and the occurrence of consequences (owing to the depletion of reserve capacity within the system), and

- the time required for a mitigation response team to repair or reconfigure the system to avoid consequences.

In order to complete both the graphical analysis and its related mathematical implications, it is important that some nomenclature be established (these terms are similar in definition to those used in the EASI tool).

Attack-Phase Variables

P_{Di} Probability of detection at the i^{th} security layer.
T_i Adversary delay time (task time) at the i^{th} security layer.
T_D Total adversary delay time for all security layers on the path (the sum of all relevant T_i from the point of detection to target access).
T_R Time required for the response force to interrupt the adversary, given that detection occurs at any point before the adversary reaches the target.

Sabotage-Phase Variables

P_S Probability that the sabotage is successful, given that it is attempted.
T_S Adversary time required to accomplish sabotage, given target access.
T_{RS} Additional time beyond T_R required for the response force to interrupt the sabotage event, given that the adversary gains access to the target. This may be necessary because it may take longer for the responders to interrupt the adversaries if they are inside a building containing the asset compared with a situation where the responders can engage the adversaries before access occurs.

Mitigation and Consequence Phase Variables

P_{DS} Probability that the sabotage event is detected immediately when it occurs.
P_{DC} Probability that the resultant consequence is detected immediately when it occurs.
T_{DS} Time to detect the sabotage event, given that it is not detected immediately when it occurs.
T_C Time delay between the sabotage event and the occurrence of the resultant consequence.
T_{DC} Time to detect the onset of consequences, given that consequences are not detected immediately when they occur.
T_M Time required to mitigate the sabotage damage or the resultant consequence.
T_{RM} Time required for the *response force* to mitigate the sabotage damage or the resultant consequence, *given* that they are present at the sabotage site *if* there is something that they can do that is quicker than T_M

Note that for this discussion, each probability is a point estimate value and each time is a probability distribution (assumed to be a normal distribution for ease of computation in the EASI methodology). On the basis of this nomenclature, the event tree for the mitigation problem can be represented (Figure 40) as an expansion of the EASI event tree (the conditions for the "special mitigation tree" will be discussed as an extension in a later section).

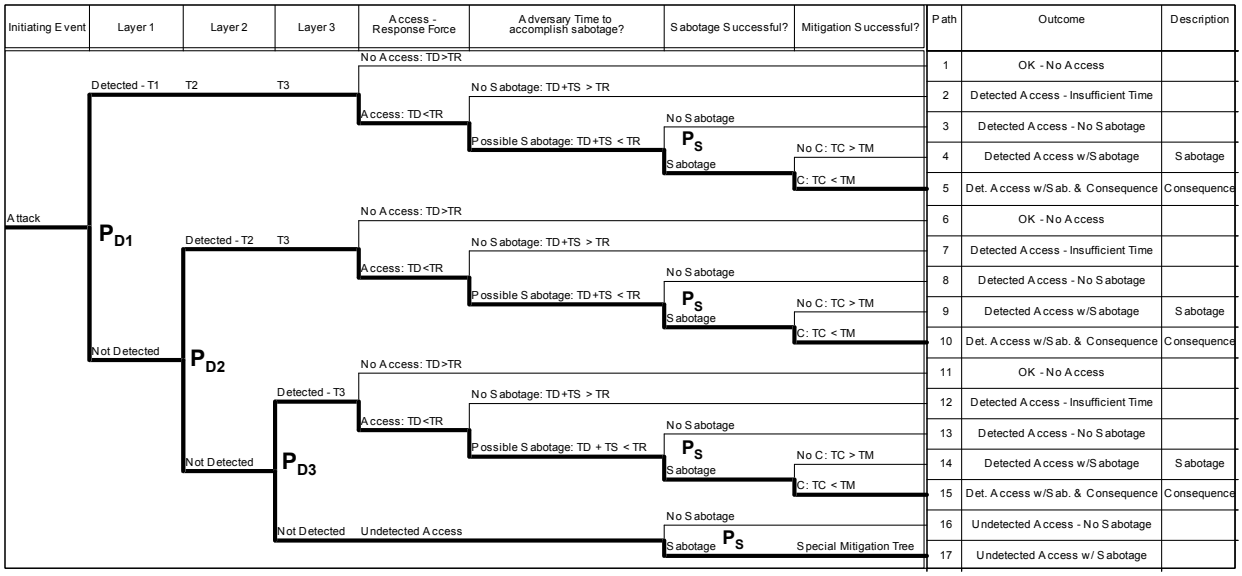

Figure 40. Example Event Tree

In this event tree, the point estimate detection probabilities and the task time probability distributions are as described for the EASI model previously. The remainder of the event tree model compares the remaining barrier delay times with the responder time given that detection takes place at a particular security layer. This delay time comparison is accomplished probabilistically, based on convolution of normal distributions for all delay times and responder times to determine a resulting probability that the adversaries can get through the last barrier before the responders can interrupt them.

The results of this "race" are then combined with the calculated probability of adversary detection at exactly the noted layer as derived from the event tree model for detection and non-detection at particular layers. The probabilities are then combined to determine the likelihood that the adversary will be successful on each event tree path. These probabilities are finally summed to determine the overall probability of adversary success for the modeled scenario. Based on this event tree model, the probability of adversary access P_{AA} (which is but one part of the event tree model) can be written as:

$$P_{AA} = \sum_{i=1}^{n} \left\{ \left[\prod_{j=1}^{i-1} \left(1 - P_{Dj}\right) \right] \cdot P_{Di} \cdot P\left[\left(\sum_{k=i}^{n} T_k \right) < T_R \right] \right\}$$

In this equation, the first sum represents the aggregation over all event tree paths and the product represents the cumulative likelihood of nondetection along the path in which detection occurs at exactly the i^{th} layer.[15] The second sum merely aggregates the barrier delay times for all security layers including and after the i^{th} layer, and this cumulative delay is compared with the responder interruption time. This equation assumes that there are *(n-1)* layers of protection, with the n^{th} layer representing the target itself. By convention, $P_{Dn} = 1.0$ and $T_n = 0.0$ in order to properly capture the situation in which the adversary accesses the target undetected.

[15] This formulation assumes that detection occurs at the *beginning* of the adversary's task time during the i^{th} layer. Similar formulae can easily be constructed assuming different detection times during the adversary's activities at the i^{th} protection layer, but this added complexity detracts from the remainder of the mitigation analysis discussion, so it is omitted here.

One can think of target access as the primary result of the basic EASI methodology. The method can be extended to represent situations in which delayed consequences can occur and mitigation is possible. The first extension simply takes the method to the point of sabotage by modifying two terms in the EASI formula. First, one must consider the time it takes for the adversary to accomplish the actual sabotage task, denoted T_S. There is the probability that the adversaries succeed in their sabotage task, P_S, which is likely to be near unity unless the sabotage task is particularly difficult or complex. Note that the adversaries can begin their sabotage task prior to the arrival of the responders, so the above equation can be extended to represent the probability of adversary sabotage P_{AS} as:

$$P_{AS} = \sum_{i=1}^{n} \left\{ \left[\prod_{j=1}^{i-1} \left(1 - P_{Dj}\right) \right] \cdot P_{Di} \cdot P\left[\left(\sum_{k=i}^{n} T_k + T_S \right) < T_R \right] \cdot P_S \right\}$$

The extension of this equation to consider mitigation can be somewhat more complex, depending upon the assumptions one makes regarding the detection of the sabotage. The simplest assumptions mathematically are that the defenders detect the sabotage immediately as it occurs (regardless of whether they have detected the adversaries at earlier stages of their attack). It is further assumed that the sabotage mitigation efforts do not begin until the sabotage event occurs. The time from the sabotage event until the consequence is manifest in the system is denoted as T_C, while the time required for mitigation of the sabotage event is denoted as T_M. If the mitigation task can be accomplished before the consequence becomes manifest, then no consequence will be seen, otherwise the consequence will occur. If T_C and T_M are normally distributed in the same manner as the other delay times, the probability that the adversary will produce consequences P_{AC} under these assumptions can be written as:

$$P_{AC} = \sum_{i=1}^{n} \left\{ \left[\prod_{j=1}^{i-1} \left(1 - P_{Dj}\right) \right] \cdot P_{Di} \cdot P\left[\left(\sum_{k=i}^{n} T_k + T_S \right) < T_R \right] \cdot P_S \cdot P\left(T_C < T_M\right) \right\}$$

The only change in this equation from the previous equation is the addition of the final probabilistic term to consider the race between the mitigation tasks and consequence achievement. This equation is valid for all pathways in the access event tree as long as the above assumptions are recognized. The duration of consequences T_{AC} is a distribution formed by the convolution of the difference $(T_M - T_C)$ and retaining only that portion where the difference is greater than zero.

A slightly more complex mitigation analysis can be performed if one does not assume that the sabotage is detected immediately. In this case, a second event tree can be constructed to represent the special case where the adversary achieves not only undetected access to the target location, but is not detected until some time after the sabotage event actually occurs. The basis for this additional complexity is embodied in the "special mitigation event tree" that was referred to above and is now presented in Figure 41.

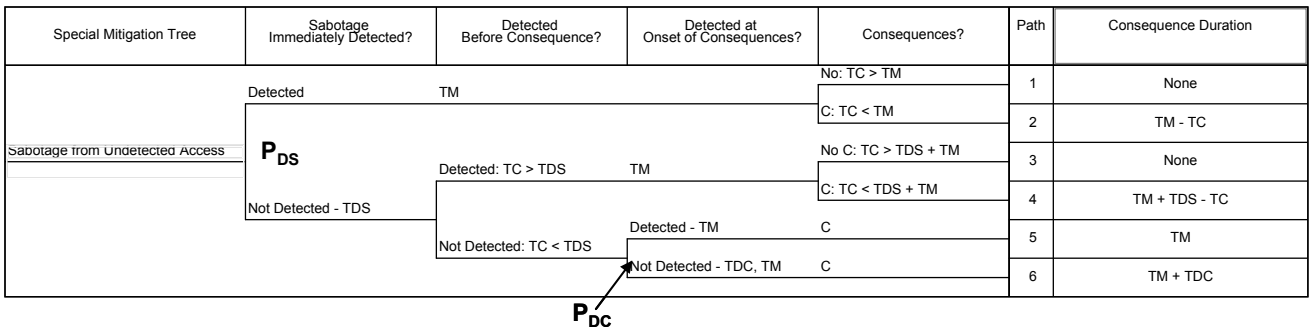

Figure 41. Second Event Tree; Adversary is Not Detected Until After Sabotage Event

One may reasonably assume that the sabotage is immediately detected in those situations where the adversary is detected and the response force is summoned because the defenders have been alerted to the presence of intruders at the site. Thus, the above equation for P_{AS} is reasonable for all cases except where $i=n$. This last term must be modified to account for the probability that the sabotage is not immediately detected. Let the probability of immediate sabotage detection be P_{DS}, and T_{DS} be a distribution for the time required to detect a sabotage event, *given* that it is not detected immediately when it occurs. Under these assumptions, the above equation may be rewritten as:

$$P_{AC} = \sum_{i=1}^{n-1}\left\{\left[\prod_{j=1}^{i-1}\left(1-P_{Dj}\right)\right]\cdot P_{Di}\cdot P\left[\left(\sum_{k=i}^{n}T_k + T_S\right) < T_R\right]\cdot P_S \cdot P\left(T_C < T_M\right)\right\} +$$

$$\prod_{j=1}^{n-1}\left(1-P_{Dj}\right)\cdot P_S \cdot \left[P_{DS}\cdot P\left(T_C < T_M\right) + \left(1-P_{DS}\right)\cdot P\left(T_C < T_{DS} + T_M\right)\right]$$

Under these assumptions, the distribution for the duration of consequences is the same as above for everything except the situation in which the sabotage is not immediately detected. T_{AC} for that case is the convolution of the distributions $(T_{DS} + T_M - T_C)$ and retaining only that portion of the distribution where the argument is greater than zero. If we further assume that the consequences are detected immediately at their onset, this resulting distribution is further limited to set all values greater than T_M to be equal to T_M.

By extending this method, we can consider the possibility that the mere fact of adversary access to the target may make the responders' job of interrupting the adversary more difficult and time-consuming. For example, perhaps the target is inside a building, and the adversaries can use the security features or construction of the building to defend themselves against the response force, or, the characteristics of the target area could require the responders to use less forceful tools and tactics against the adversary force lest they actually cause the consequence by their actions taken to neutralize the adversaries. If either or both of these are true, then one should postulate an additional time required for the responders to interrupt adversaries that are inside the target location, and we denote this time as T_{RS}.

In the most general case, one must evaluate each detection point to determine whether detection at that point enables timely response before or after the adversary gains access to the target. Since these times are statistically distributed, it is likely that this most general situation can only be evaluated using simulation techniques such as Monte Carlo discrete event simulation.

However, for situation where the response force is particularly slow (which may be the case for many infrastructure analyses), closed form solutions are possible. If the response force is unlikely to arrive before the adversary gains access to the target, a reasonable approximation for P_{AS} can be written as:

$$P_{AS} = \sum_{i=1}^{n} \left\{ \left[\prod_{j=1}^{i-1} \left(1 - P_{Dj}\right) \right] \cdot P_{Di} \cdot P\left[\left(\sum_{k=i}^{n} T_k + T_S \right) < \left(T_R + T_{RS}\right) \right] \cdot P_S \right\}$$

Under these assumptions, an approximation for P_{AC} for the case when sabotage events are immediately detected can be written as:

$$P_{AC} = \sum_{i=1}^{n} \left\{ \left[\prod_{j=1}^{i-1} \left(1 - P_{Dj}\right) \right] \cdot P_{Di} \cdot P\left[\left(\sum_{k=i}^{n} T_k + T_S \right) < \left(T_R + T_{RS}\right) \right] \cdot P_S \cdot P\left(T_C < T_M\right) \right\}$$

while the approximation for P_{AC} when sabotage events are not immediately detected is:

$$P_{AC} = \sum_{i=1}^{n-1} \left\{ \left[\prod_{j=1}^{i-1} \left(1 - P_{Dj}\right) \right] \cdot P_{Di} \cdot P\left[\left(\sum_{k=i}^{n} T_k + T_S \right) < \left(T_R + T_{RS}\right) \right] \cdot P_S \cdot P\left(T_C < T_M\right) \right\} +$$

$$\prod_{j=1}^{n-1} \left(1 - P_{Dj}\right) \cdot P_S \cdot \left[P_{DS} \cdot P\left(T_C < T_M\right) + \left(1 - P_{DS}\right) \cdot P\left(T_C < T_{DS} + T_M\right) \right]$$

The distribution for the duration of consequences unchanged from that presented for the case where T_{RS} is not considered.

5.7.2 Methodologies for Evaluating Multiple-Target Cut Sets

The previous section described how the probabilities for access, sabotage, and consequences can be calculated for a single-target cut set using closed-form mathematical formulae under various sets of common assumptions. This assumes that sabotage to a single asset is sufficient to cause the consequence of concern. For multiple-target or multiple-asset cut sets, some of these probabilities are more difficult to evaluate, and most notable among them is the probability that the consequence of concern will be realized. This section describes algorithms that can be used for a couple of common sets of assumptions to evaluate multiple-target cut sets in the presence of mitigation.

The simplest case to consider is the situation where mitigation is not present, as unrealistic as this may be for infrastructure security. In this case, the correct combination of sabotage events will inevitably lead to consequences, and it is only the time of the onset of those consequences that is uncertain. Assuming that the cut sets are minimal,[16] all targets must be sabotaged in order for the consequence to be realized. Thus, we are looking at the compound probabilistic event in which all sabotage events in the cut set occur. Probabilistic independence of the events can occur under a number of circumstances, such as:

- The adversary uses multiple attack teams and assigns one team to attack each target in the cut set, or

[16] That is, each cut set represents one group of sabotage conditions that are *necessary and sufficient* for realization of the consequences. Thus, all conditions in the cut set *must* occur for the consequences to be realized, *and* the consequences *must* occur if all of the conditions in the cut set occur (in the absence of mitigation).

- The adversary uses fewer attack teams, but the time required for the defenders to respond to each attack is so long that the attackers are likely to have accomplished their goal and left before the response force arrives, or

- The list of possible cut sets is extensive enough that completion of one attack by the adversaries does not provide the defenders with enough information to respond preemptively to thwart other events in the cut set.

If independence can be argued for the attacks in the multiple-target cut set, then one can simply compute the probability for adversary sabotage P_{AS} for each attack in the cut set and multiply them (i.e., combine them using "AND" logic) to estimate the probability for the cut set.

Independence cannot be argued for the events in the cut set when the adversary success probability for a subsequent attack in the cut set can change *given* that another attack in the cut set has already been carried out. In this case, the analyst must examine the events in the cut set to determine the order of attacks that would be most advantageous to the adversary. The analyst then computes the independent P_{AS} for the first event and estimates conditional values for P_{AS} for subsequent events. These conditional values may be based on revised response time estimates T_R that account for either the distraction of the response force caused by the first attack (increasing T_R) or for the heightened state of alert or preemptive response for the response force caused by the attack (decreasing T_R). One may also compute independent values for P_{AS} for each attack and use expert judgment to adjust the values for subsequent attacks in the cut set. When the effect of an initial attack on security responders cannot be easily established *a priori*, assuming independence of the various P_{AS} within each cut set can be a useful screening tool.

Adding mitigation to this situation brings forth significant additional complications for the analysis process. Recall that, under the definition of a minimal cut set, all sabotage events within the cut set have to be present *simultaneously* to realize the consequences of concern. For this initial discussion, let us neglect the delayed onset of consequences, which was denoted T_C in the previous section. In this case, consequences occur only during the time interval between the *last* sabotage event and completion of the *first* mitigation event – regardless of which members of the cut set they are associated with. In other words, the first repair, which ends the consequence, may be on *any* of the assets in the cut set. Therefore, in order to accurately relate the probability of consequences for multiple-target cut sets, one must compute outage start times and outage end times as distributions for all events in the cut set and compute the appropriate maxima and minima during the convolution process. Such complicated dependencies are extremely difficult to represent as closed-form mathematical formulae, but can be accomplished readily through Monte Carlo sampling algorithms.

Consider now a situation where parallel simultaneous attacks occur on all assets in a multiple target cut set. The method for constructing the probability of consequence occurrence for the overall cut set is as follows:

1. Construct a distribution for the time at which the adversary causes sabotage for each event/asset in the cut set, denoted as T_{AS}.[17] One must also have at hand the existing distribution for the duration of the mitigation task for each asset T_M.

2. Randomly draw an observation from the T_{AS} and T_M distribution for each asset. Determine the time at which the asset will be restored T_{AR} for this observation by adding the selected values for T_{AS} and T_M.[18]

3. Select the largest value of T_{AS} over all assets ($T_{AS,\,max}$) and the smallest value of T_{AR} over all assets ($T_{AR,\,min}$). If $T_{AR,\,min} < T_{AS,\,max}$, then no consequences occur as one asset in the cut set was restored before the last asset was sabotaged. Otherwise, the consequence was realized, and the duration for which consequences are achieved is $T_{AC} = T_{AR,\,min} - T_{AS,\,max}$.

4. Repeat steps 2 and 3 for a large number of observations. The fraction of observations in which consequences are realized represents P_{AC}, the conditional probability of consequences occurring given the fact that *all* assets in the cut set are successfully sabotaged. The population of T_{AC} across all observations corresponds to a distribution for the conditional consequence duration given that consequences occur.

This method can be easily extended to represent situations in which there is a delay from the accomplishment of the final sabotage until the realization of consequences for the case where that delay does not depend on the order in which sabotage events occur. If this consequence time is T_C, then the inequality in step 3 above becomes $T_{AR,\,min} < (T_{AS,\,max} + T_C)$, and the consequence duration is represented as $T_{AC} = T_{AR,\,min} - T_{AS,\,max} - T_C$. These changes simply account for the additional delay T_C between the accomplishment of the last sabotage and the realization of consequences. T_C may be either deterministic or probabilistic, and if it is probabilistic, its distribution must also be sampled in step 2 of the above algorithm.

The more general case in which T_C varies depending upon the order in which the adversaries accomplish their sabotage objectives is beyond the scope of this paper, but can be approximated by judicious selection of T_C in this simpler case. If one sabotage event would usually take longer or otherwise be the last one accomplished, T_C should be selected to correspond to that event/ asset. Otherwise, one should select T_C to be the value that corresponds to the shortest time from sabotage to consequence, although this does introduce some conservatism into the computations of P_{AC} and T_{AC}.

The potential for non-simultaneous attacks introduces an additional complexity to the above algorithm. If the attacks are non-simultaneous but not necessarily sequential, one can postulate an adversary delay time T_{AD} for each asset or attack between some arbitrary starting time for the overall attack and the beginning of the attack on the specific asset. In this case, the value selected for $T_{AS,\,max}$ in step 3 above must be the largest value of the quantity ($T_{AD} + T_{AS}$) over all

[17] Development of this distribution is nontrivial, and is discussed in the following section.

[18] Note that one cannot simply develop a distribution and use for T_{AR} because the restoration event is not independent of the sabotage event. Developing such a distribution could cause the random sampling scheme to produce such nonphysical results as "the asset is restored before it is sabotaged," and will certainly produce an unrealistic distribution of the repair time over the population of random observations used in the Monte Carlo analysis.

assets, and the value of $T_{AR,\,min}$ must be the smallest value of the quantity ($T_{AD} + T_{AS} + T_M$) as computed on an asset by asset basis. T_{AD} may be different for each asset in the cut set, and may be either deterministic or probabilistic. If it is probabilistic, its distribution must also be sampled in step 2 of the above algorithm.

In order to consider sequential attacks, the analyst must either postulate an order in which the attacks occur, or assess that the adversary has no preference in attack order and thus select that order randomly. Often the analyst will note specific advantages for the adversary that cause one attack order to be preferred over the others, and if this advantage is present, the analyst should assume that the adversary will seek to exploit it and order the attacks in that way during the analysis. For sequential attacks, T_{AD} should be interpreted as the time from the completion of one sabotage event to the initiation of the next attack. Also, the astute observer will note that using the independent values for T_{AS} in a sequential attack may be significantly conservative in that it assumes that the adversary begins to attack each successive asset in an undetected state whereas in reality the detection and response at one asset may make detection a foregone conclusion at later assets in the attack process (e.g., the responders could arrive just as the first sabotage event is completed and actually chase the adversaries to their next target, obviating the need for further detection at that target). Thus, the computations of P_{AC} and T_{AC} from methods that use independent values for T_{AS} should be viewed as worst case or upper bound values. It may be possible to obtain more refined estimates for these values from further analyses, but the added complexities are large (a new event tree would need to be developed, for example) and beyond the scope of this paper. Furthermore, the results of the simple worst-case analysis should be useful for pointing analysts to the particular cut sets that are worthy of more in-depth consideration.

The method for assessing sequential attacks is as follows:

1. Construct a distribution for the time at which the adversary causes sabotage for each event/asset in the cut set, denoted as T_{AS}. One must also have at hand the existing distributions T_M and T_{AD} for each asset and T_C for the cut set. Determine the order in which the cut set assets will be attacked, numbered 1 through n.

2. Randomly draw an observation from the T_{AD}, T_{AS} and T_M distribution for each asset and the T_C distribution for the system. Since each of these values is referenced to the start of the specific activity, and not to the start of the overall attack scenario, we must compute the sabotage time for each asset in the cut set. The sabotage time for the i^{th} asset will be denoted as T_{Si}, and is computed as

$$T_{Si} = \sum_{j=1}^{i \le n} \left(T_{AD,j} + T_{AS,j} \right)$$

3. Determine the time at which the asset will be restored T_{ARi} for this observation by adding the selected values for T_{Si} and T_{Mi}.

4. Select the largest value of T_{Si} ($T_{AS,\,max}$) and the smallest value of T_{ARi} ($T_{AR,\,min}$). If $T_{AR,\,min} < (T_{AS,\,max} + T_C)$, then no consequences occur as one asset in the cut set was

restored before the last asset was sabotaged. Otherwise, the consequence was realized, and the duration for which consequences are achieved is $T_{AC} = T_{AR,\,min} - T_{AS,\,max} - T_C$.

5. Repeat steps 2, 3 and 4 for a large number of observations. The fraction of observations in which consequences are realized represents P_{AC}, the conditional probability of consequences occurring given the fact that *all* assets in the cut set are successfully sabotaged. The population of T_{AC} across all observations corresponds to a distribution for the conditional consequence duration given that consequences occur.

The development of the *probability* of the adversary accomplishing a sabotage event was described in the first section of this paper. It can be represented as a single closed-form equation, although the computation of the probability term $P\left[\left(\sum_{k=i}^{n} T_k + T_S\right) < \left(T_R + T_{RS}\right)\right]$ from the distributions for T_k, T_S, T_R and T_{RS} may require Monte Carlo analysis, depending on the form of these distributions.

5.7.3 Development of Time for Adversary Sabotage (T_{AS})

The computation of the *time* of the sabotage event itself is far more difficult. Recall that there can be several paths through the event tree that result in a successful sabotage event. The conditional probability of sabotage for a scenario in which the adversary is detected in exactly the i^{th} layer of protection is[19]

$$P_{ASi} = \left[\prod_{j=1}^{i-1}\left(1 - P_{Dj}\right)\right] \cdot P_{Di} \cdot P\left[\left(\sum_{k=i}^{n} T_k + T_S\right) < \left(T_R + T_{RS}\right)\right] \cdot P_S$$

The distribution of sabotage times for this situation is a difficult conditional convolution that likely requires a Monte Carlo analysis. For the simple case in which there is no response by the defender, the distribution for time of sabotage after the start of the attack on the asset can be written as

$$T_{AS} = \sum_{k=1}^{i-1} T_k + \sum_{k=i}^{n} T_k + T_S = \sum_{k=1}^{n} T_k + T_S$$

where detection occurs at the i^{th} task, so the first sum represents the time spent by the adversary prior to detection, and the second sum represents the time spent by the adversary in defeating security protection layers after detection.

Most traditional vulnerability analyses are concerned with task times after detection – under the assumption that the adversary is trying to complete the task as quickly as possible without fear of detection – because the analysis is seeking to determine whether the remaining adversary task time is sufficient to allow for a timely reaction of the response force to interrupt the adversary. However, when considering sabotage events with consequence mitigation, the element of time

[19] Recall that the case where $i=n$ represents the situation where the adversary arrives at the target undetected. Recall also that all times T in these equations are viewed as distributions.

can be critical both before and after detection because the mitigation clock starts running upon the achievement of the first sabotage event and consequences occur on the completion of the last sabotage event. Thus, the task time before detection can be important for all types of multiple-target attacks when mitigation is present.[20]

If one estimates T_{AS} using the formula above and uses task delay times where the adversary is not trying to avoid detection, the resulting value for T_{AS} is be conservatively short (maybe unrealistically so). However, developing multiple estimates for each T_k and propagating each through the event tree to obtain a realistically weighted value for T_{AS} is a complex and time-intensive process that may not be worth the effort for the following two reasons:

- The response force arrival times for infrastructures are often so long that an adversary may not realistically need to avoid detection for *any* part of the attack in order for the sabotage attack to be successful.

- For most multiple-target attacks that are non-simultaneous, the adversaries may not seek stealth in attacks after the first in the hopes of completing all attacks as quickly as possible in order to obtain the maximum possible outage duration from the attack.

Thus, for many multiple-target infrastructure attacks, estimating T_{AS} using the above equation is a reasonable approximation. A more accurate treatment of this variable would be required for multiple-target attacks against facilities in which the response force arrives fast enough to have a high probability of interrupting the adversary during the attack process.

A reasonable next step toward achieving a more accurate estimate of T_{AS} involves a rudimentary treatment of rapid adversary attacks. While the event tree treatment recognizes the possibility of multiple detection points, real attacks are often planned to include a "practical detection point" (PDP). The PDP represents the point in the attack plan before which the adversary believes that they will not be detected and after which they will assume that they are detected. Often, the PDP is associated with the first easily detectable event in the attack plan, such as the use of overt force or explosives. As an alternative, many vulnerability analysis methods make use of a "critical detection point" (CDP) which represents the last detection point in a facility's defenses for which the response force can likely respond in time to provide timely interruption of the adversary's attack. These points can be used to estimate T_{AS} as follows. The sabotage event only occurs if the adversary task time after detection is shorter than the response time. Thus, if the PDP or CDP occurs at the beginning of the i^{th} adversary task, and, as before, the adversary and responder task times are represented as probabilistic distributions, a distribution for T_{AS} can be estimated using the following Monte Carlo approach, as follows.

[20] The possible exception to this statement is well-coordinated parallel simultaneous attacks in which the separate adversary teams plan their attacks so that they are *detected* simultaneously rather than being *initiated* simultaneously. Such sophisticated attacks are beyond the scope of this paper, but may be modeled by neglecting the first sum from the equation for T_{AS} above. For nonsimultaneous attacks, one must also add the delay time between attacks T_{AD} to this equation to obtain an accurate estimate for T_{AD}.

1. Randomly draw an observation for the task time for the response times T_R and T_{RS}, the sabotage time T_S, and for each adversary activity T_k (including those activities both before and after the PDP or CDP).

2. Determine whether the sabotage would be successful if these adversary and responder times actually occurred. If $T_R + T_{RS} > \sum_{k=i}^{n} T_k + T_S$, then sabotage occurs, and this observation contributes to the *conditional* distribution for T_{AS} (recall that T_{AS} represents the time at which sabotage occurs *given* that the sabotage event is successful). Otherwise, this observation is neglected because the characteristics associated with this observation have already been captured in P_S, the probability that the sabotage is successful.

3. For this observation, $T_{AS} = \sum_{k=1}^{i-1} T_k + \sum_{k=i}^{n} T_k + T_S = \sum_{k=1}^{n} T_k + T_S$. This is the same formula used to compute T_{AS} previously, but now it is applied only to those observations where timely interruption by the response force does not occur.

4. Repeat steps 1, 2 and 3 for a large number of observations. The population of T_{AS} across those observations where sabotage occurs (i.e., where timely interruption does not occur) corresponds to a distribution for the sabotage time T_{AS}.

For more complex or highly coordinated attacks, it may be necessary to construct the entire attack timeline for a large number of observations using Monte Carlo discrete event simulation to obtain an accurate picture of the interactions between the attack timeline, the response force timeline, and the mitigation timeline. Such simulation can be done efficiently based on the event trees shown earlier in this section. In its simplest form, the discrete event simulation produces the following results for each Monte Carlo observation:

- Is sabotage successful (yes or no)?
- Does a consequence (e.g., outage) occur (yes or no)?
- If the consequence occurs, what is its duration before it is terminated by mitigation actions?

More efficient simulation options are available that produce probabilistic answers to the "yes or no" questions posed above should these computations become too time-consuming. [Wyss and Duran 2001, Wyss et al. 2004]

5.8 Conditional Risk Set Developer

The proposed methodology helps identify why vulnerabilities exist and what causes them. The end goal is a means for determining the most cost-effective options for reducing the "risk" due to malevolent attacks that exploit those vulnerabilities. The Risk (R) is equal to:

$$R = P_A * P_{S|A} * C$$

In the general case, the Probability of Attack (P_A) is either: a) unknown; or b) considered "sensitive information" because it may be based on non-public information. As such, this report focuses on the "conditional risk", which is the risk *given* that an attack occurs. In that case, the Conditional Risk for a given threat ($R_{C,T}$) is defined as:

$$R_{C,T} = P_{S|A,T} * C$$

$P_{S|A,T}$ is the vulnerability estimate for a given threat (T), which should be determined independently of the consequence. Thus, for a selected asset, three parameters define the conditional risk: adversary capability (or threat), asset vulnerability to this threat, and consequence linked to the asset. A 3-dimensional plot is difficult to utilize because the adversary capability would require a continuum descriptor combining both cyber and physical attributes. In addition, the consequence and asset vulnerability ranges estimated by the Belief/Plausibility function create a visual challenge for the risk analyst.

The CPSAM software creates a data tree with all of the data elements and risk parameter estimates (Figure 42) that can be expanded to assess needed detail at various levels. The CPSAM uses 2-dimensional plots of the asset insecurity and consequence with color bars to represent the assigned threats (Figure 43).

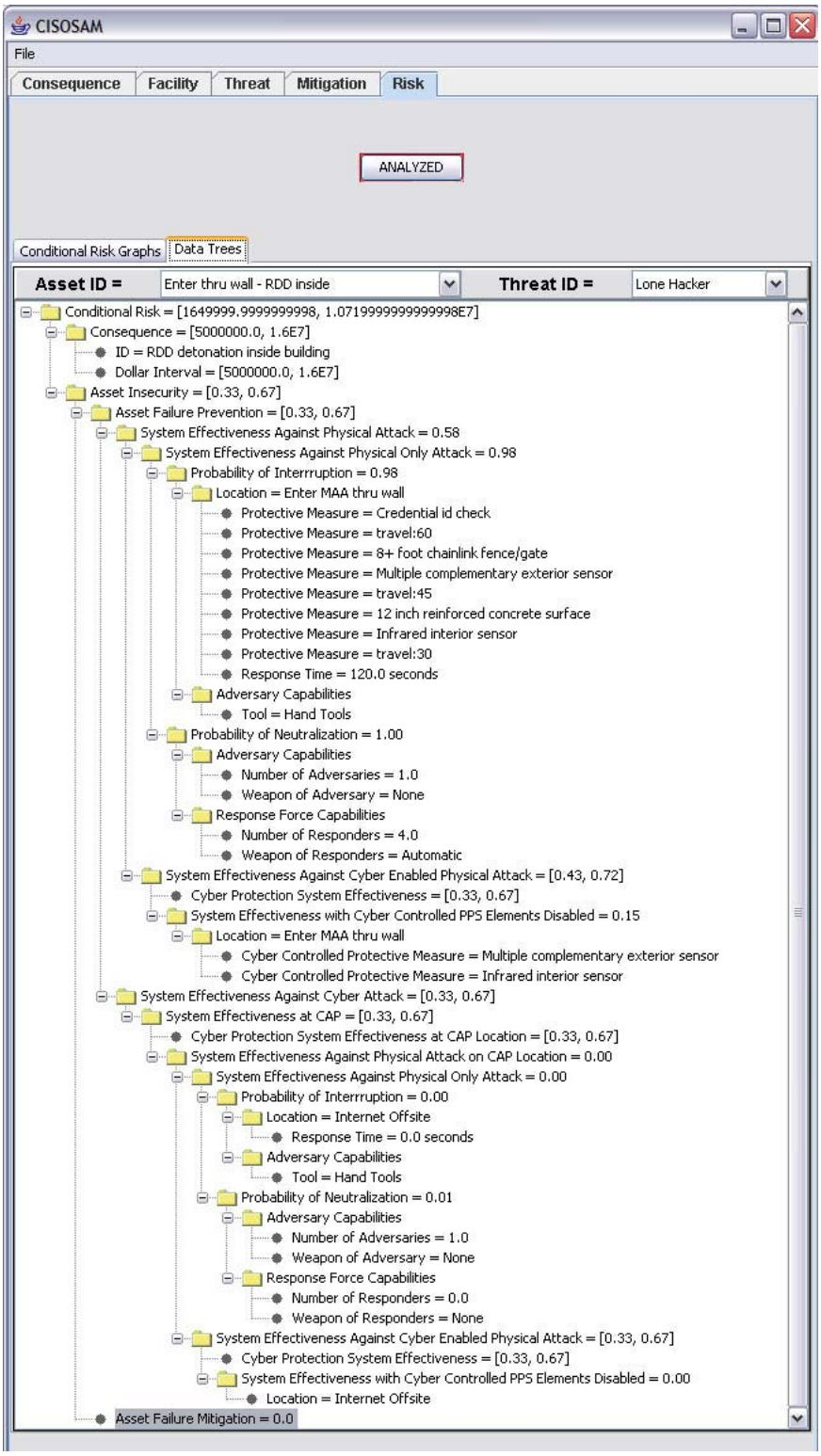

Figure 42. Data Tree Showing Risk Parameter Estimates

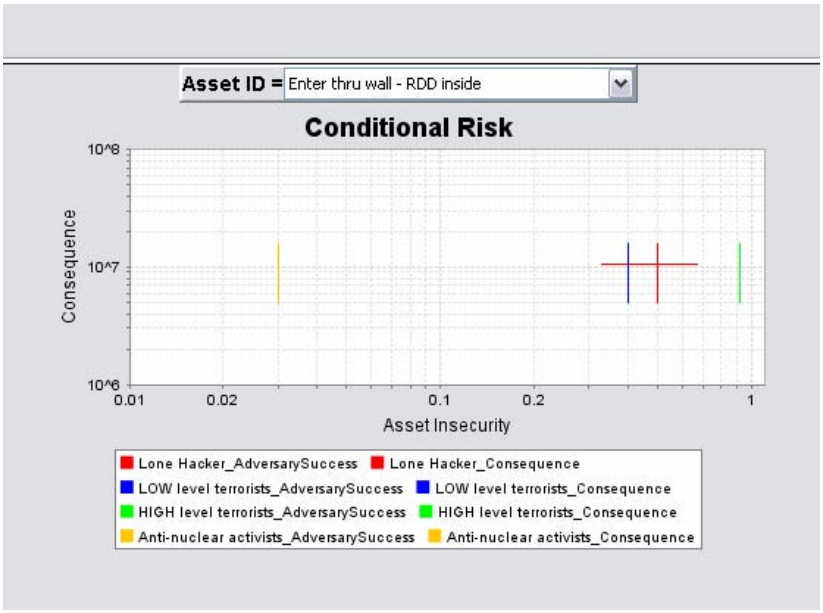

Figure 43. Example of Conditional Risk Set Visualization

6. CPSAM Analyst Software Implementation

6.1 Architecture of CISOSAM Software

The software is a standard Java application developed using the NetBeans 5.0 Integrated Development Environment (IDE). The Java 2 Standard Edition 5.0 (J2SE5.0) virtual machine is required to run the code as the software uses features new to the Java language, such as generics, which were not present in earlier versions of Java. NetBeans is a free product available from Sun Microsystems, and NetBeans 5 comes bundled with J2SE5.0.

Figure 44 shows the CISOSAM code in the NetBeans environment. CISOSAM is a NetBeans project that is a Java application.[21] The "Files" panel in Figure 44 shows the top-level directory structure of the project. The *build.xml* file is used by the ant build tool within NetBeans to build the project; this file is automatically generated by NetBeans. The *manifest.mf* file is used to deploy the application as a jar file. The other files are application-specific files needed by the application. The Examples directory contains saved example data files (saved in xml format) that can be opened for use in the application. The remaining directories—*build*, *dist*, *nbproject*, *src*, and —are automatically generated by NetBeans for a project that is a Java application, as follows:

- "*build*" contains the .class files for the virtual machine, created by compiling the .java source files.
- "*dist*" contains the jar'd application.
- "*nbproject*" contains files used by NetBeans to manage the project.
- "*src*" contains the .java text source files.
- "*test*" contains JUnit test files; the JUnit test harness is part of NetBeans. JUnit tests were not created in this application.

Figure 44 also shows the source code for *Cisosam.java*, which contains the main method that is called automatically when the application is run.

Figure 45 shows selected directories and files under dist and source, and the source code for *manifest.mf*. In *manifest.mf* the Main-Class, *gov.sandia.cisosam.Cisosam*, specifies the fully qualified name of the class file containing the main method; this is used for ease of executing the jar'd application from the java command line. The Class-Path is ".", which is the path specified by the Java System property *System.getProperty*("*java.user.dir*"); this is the String formatted path to the directory from which the Java virtual machine is invoked. *dist* contains an executable file, *jdk-1_5_0_07-windows-i586-p.exe*, that will install J2SE5.0 if the client does not already have that Java virtual machine. The datapackage subdirectory under *src* contains the non-GUI part of a code called *BeliefConvolution* that convolutes numeric variables using the

[21] Projects other than applications can be developed in NetBeans, such as: applets, servlets, JavaServerPages (JSP), and webapps.

belief/plausibility of uncertainty. [Darby 2006] The org subdirectory under *src* contains source code for *JFreeChart* and *JGraph*, two graphical libraries used in the application.

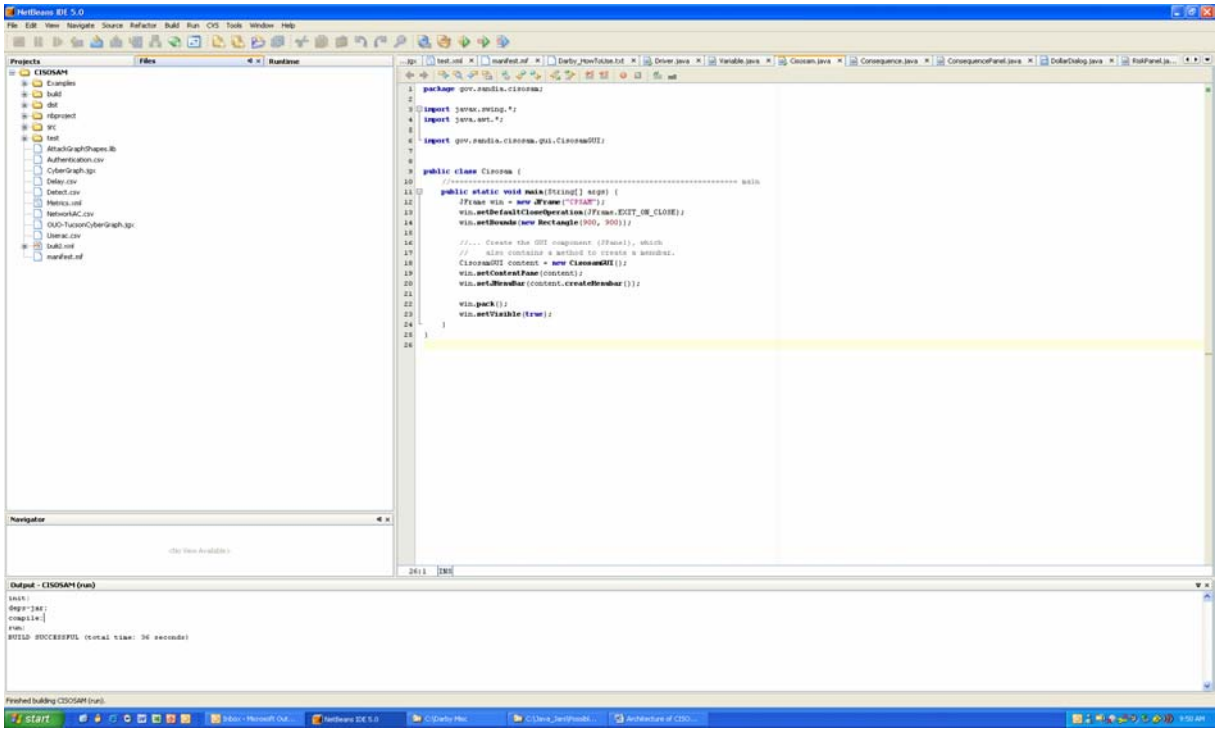

Figure 44. CISOSAM in NetBeans 5.0

Since the source code for these two libraries is included, the jar'd CISOSAM application will contain these libraries without having to include the jar'd version of each of these libraries in the deployment.measure of uncertainty [Darby 2006]. The *org* subdirectory under *src* contains source code for *JFreeChart* and *JGraph*, two graphical libraries used in the application. Since the source code for these two libraries is included, the jar'd CISOSAM application will contain these libraries without having to include the jar'd version of each of these libraries in the deployment.

The built-in guibuilder in NetBeans was not used to develop the GUI for most of the files used in CISOSAM; the GUI code was custom-developed by the developer. However, the NetBeans gui-builder was used for specific code used in CISOSAM, such as the *DollarDialog JPanel* shown in Figure 46. The application has no custom-coded multithreading code beyond that used in libraries such as *JGraph*.

The application saves data by creating an xml file from code objects. Similarly, the application reads (opens) data by parsing a previously saved xml file and creating code objects; the parser is a DOM parser, not a SAX parser. The code for reading/writing xml is in *CisosamGUI.java*, as shown in Figure 47.

Figure 48 shows a top-level, simple unified modeling language (uml) description of the application. The solid arrows indicate association and the dashed arrows indicate dependence.

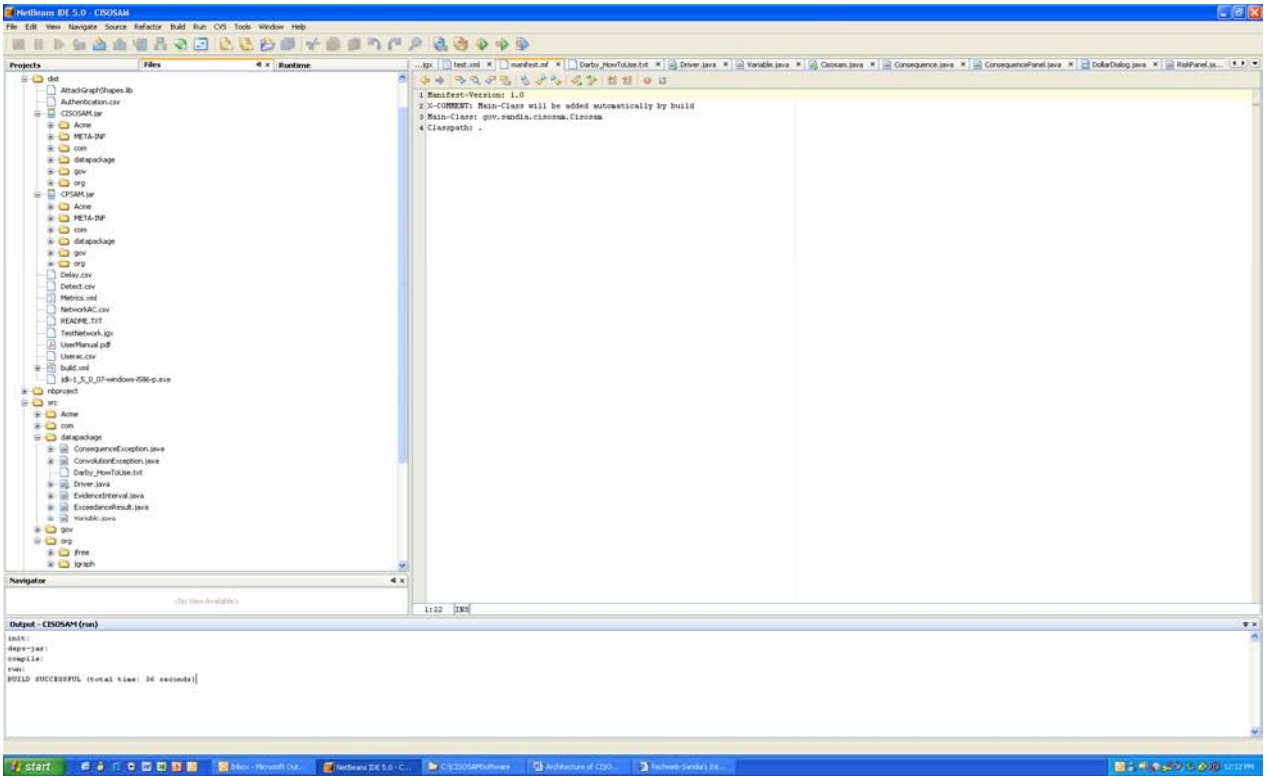

Figure 45. Structure of *dist* and *src* and *manifest.mf*

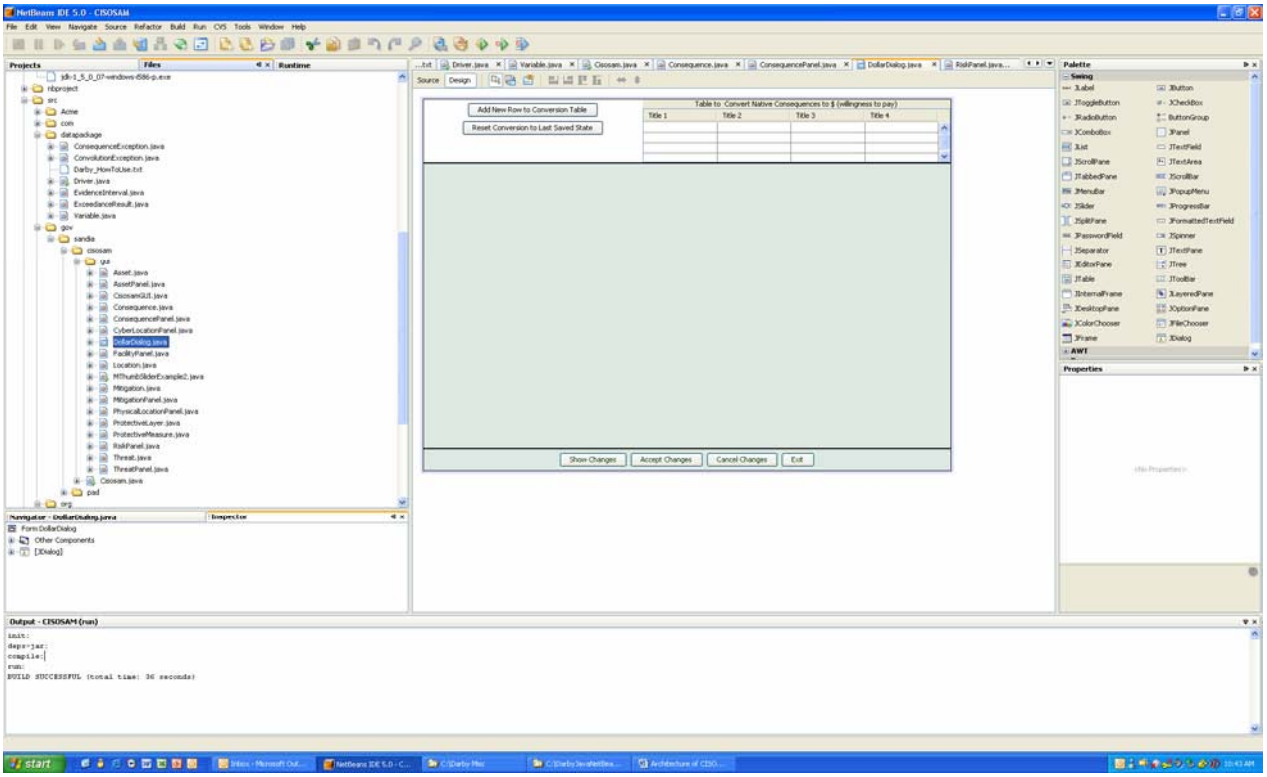

Figure 46. Example of NetBeans GUI Builder

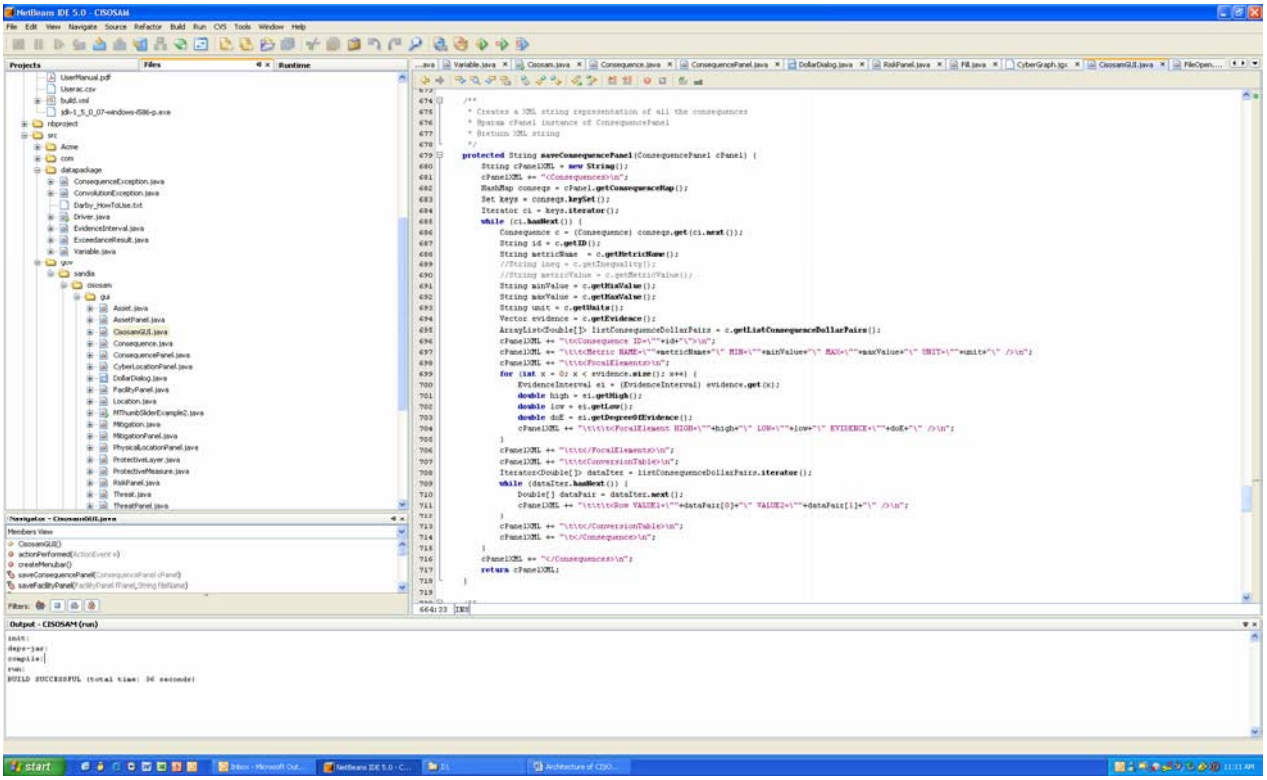

Figure 47. Example of Code to Save in xml

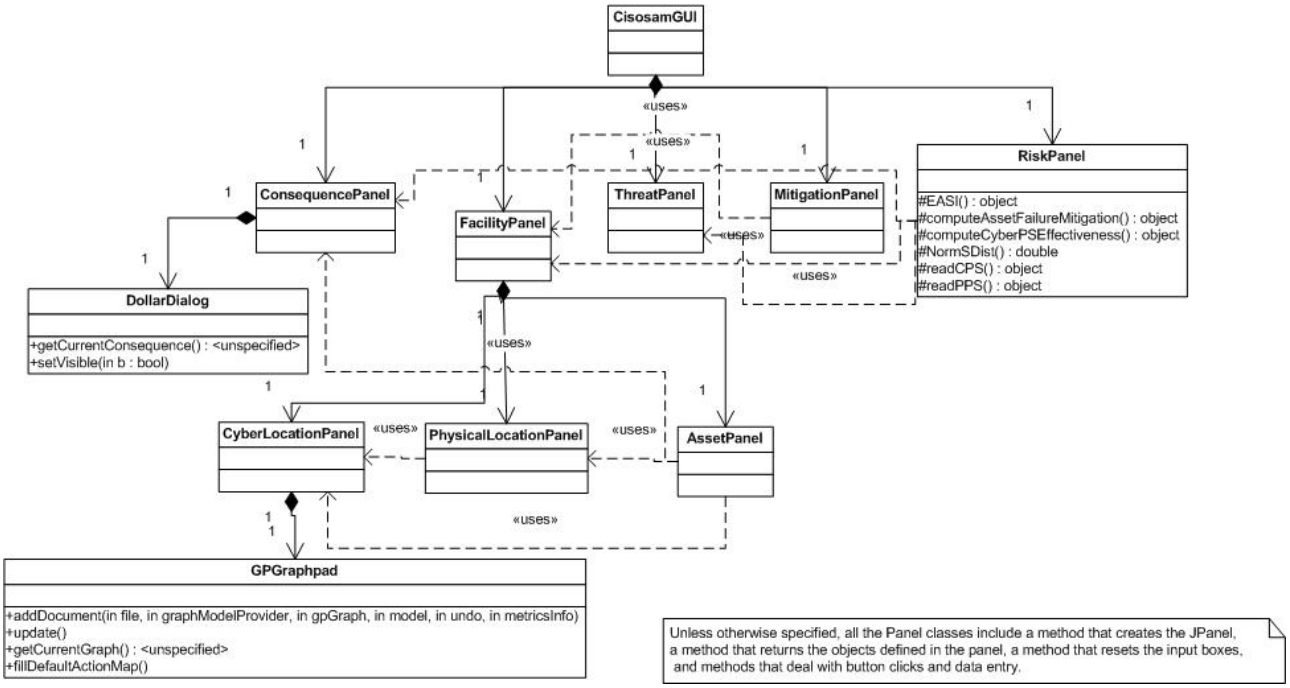

Figure 48. Simple UML Diagram of CISOSAM Application

The *RiskPanel* class calls methods that evaluate the risk. Figure 49 shows the JFrame gui for the application with the *RiskPanel* selected using the Risk tab. The major methods of interest for the risk calculation are noted on the class diagram for *RiskPanel* in Figure 48. *EASI()* estimates the probability that a sufficient number of response force personnel will interrupt the adversary at some point before the adversary completes their task. *computeAssetFailureMitigation()* computes the asset failure mitigation index.

computeCyberPSeffectiveness() computes the cyber protection system effectiveness index. *NormSDist()* computes the probability that the observed value of a standard normal random variable will be less than or equal to a specified value. *readCPS()* creates a HashTable for the cyber protection system file. *readPPS()* creates a HashTable for the physical protection system file.

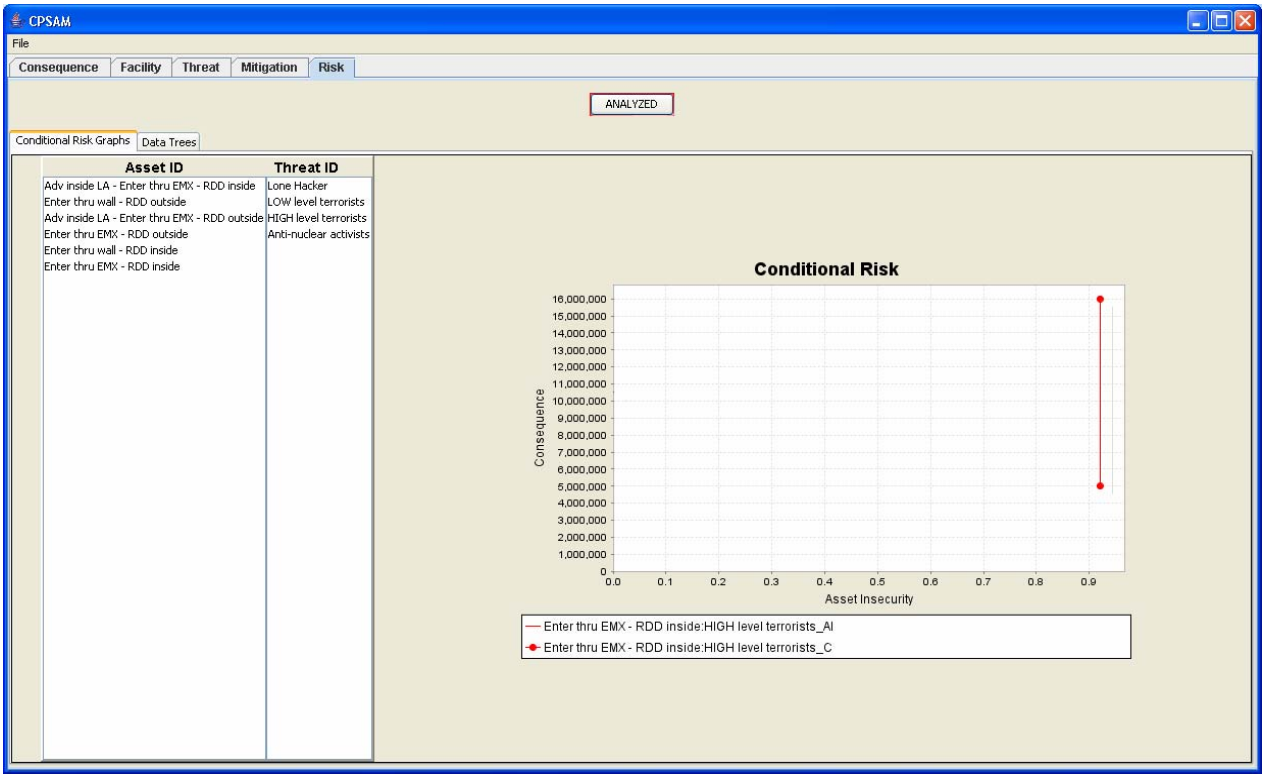

Figure 49. CISOSAM GUI JFrame

7. Application of CPSAM Analysis Tool

To test the CPSAM Analysis Tool software, two test cases were formulated based on real-world examples. The first test case is based on a portion of a real US city's water infrastructure system. The second is a fictional high-security facility that resembles NNSA's domestic high-security facilities. The two test cases were selected to exercise the tool as completely as possible; the water utility example is intended to demonstrate the tool's applicability to examining cyber and physical security of large, networked infrastructures, while the high-security facility example shows its usefulness in examining the many high-consequence, high-security facilities in operation throughout the country.

7.1 Water Utility

Over the past decade, the Security Systems and Technology Center at Sandia has developed several Risk Assessment Methodologies (RAMs) for the assessment of security of critical infrastructures. The Environmental Protection Agency (EPA) funded the creation and application of RAM-W, a methodology geared specifically toward the assessment of the security of municipal water systems. Since its creation, RAM-W has been applied to many city water infrastructures. In order to exercise the CPSAM Analysis Tool software, the documentation and reports pertaining to the application of the RAM-W process to a particular city were examined and used as references to model a portion of the real-world municipal system. For classification reasons, the identity of the specific city is not disclosed and the names of the facilities have been altered.

7.1.1 Facilities and Infrastructure System Layout

Like most networked infrastructures, the city water system examined was very complex and consisted of dozens of facilities, control stations, and access points. To simplify the first test case of the tool, a single "arm" of the system was examined. This arm consisted of all steps of the water system from beginning to end, from ground water collection and storage, to water treatment, to storage and eventual distribution of treated water to customers. To fully incorporate the cyber security aspects of the system, the central command and control center for the system was also included in the analysis. A diagram of the system and the four facilities analyzed appears in Figure 50.

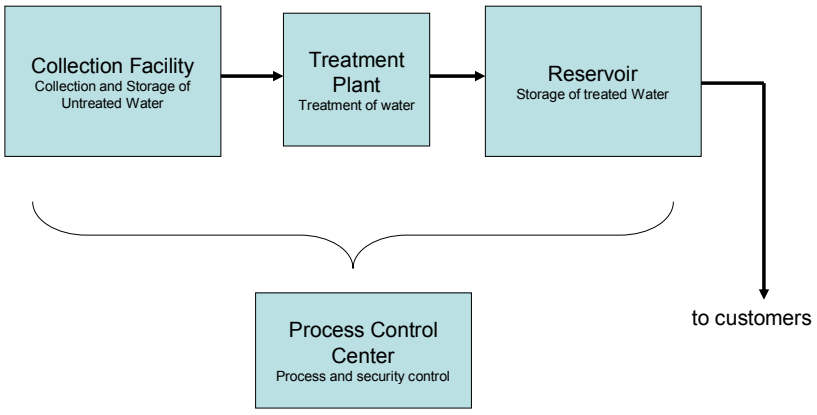

Figure 50. Schematic of Simplified Water Utility Infrastructure

A notional layout of the four facilities were generated, including the cyber and physical security safeguards in place. These layouts, seen in Figure 51 through Figure 54, could then be used to input the necessary information into the CPSAM analysis software.

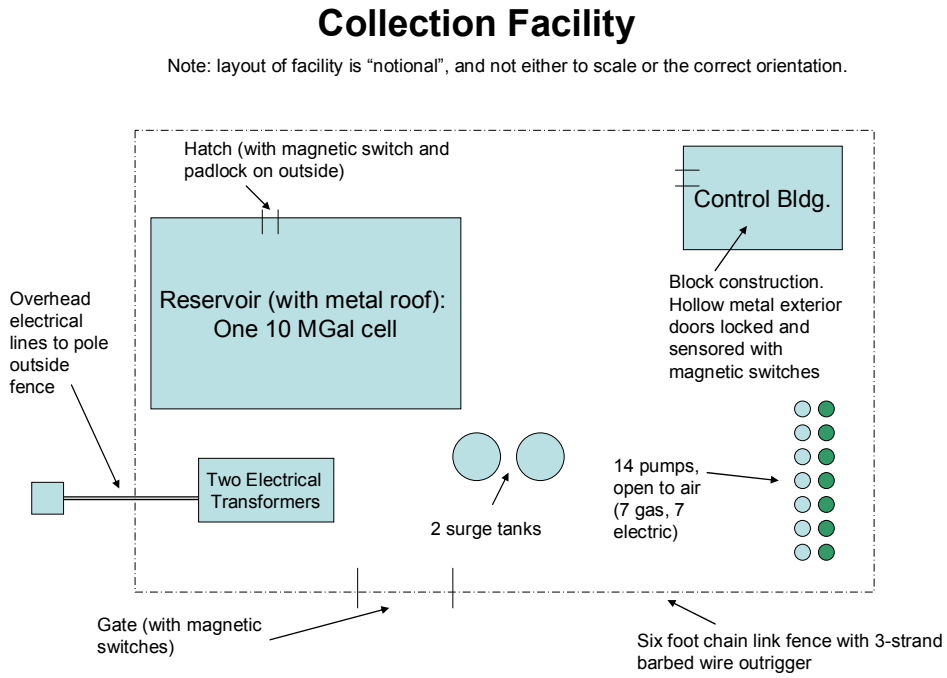

Figure 51. Notional Layout of Collection Facility

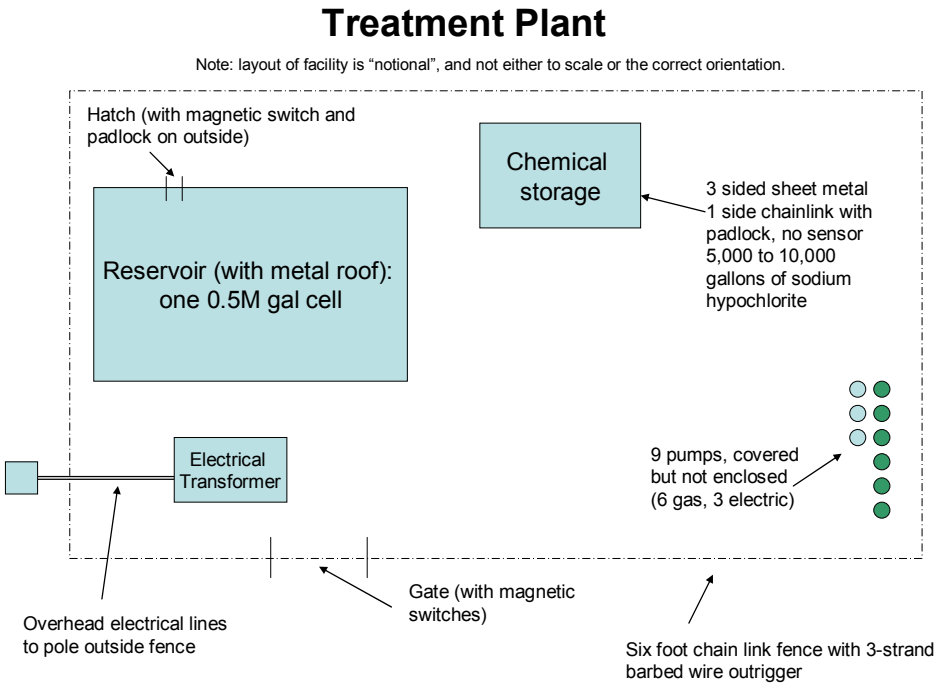

Figure 52. Notional Layout of Treatment Plant

Reservoir

- Note: layout of facility is "notional", and not either to scale or the correct orientation.

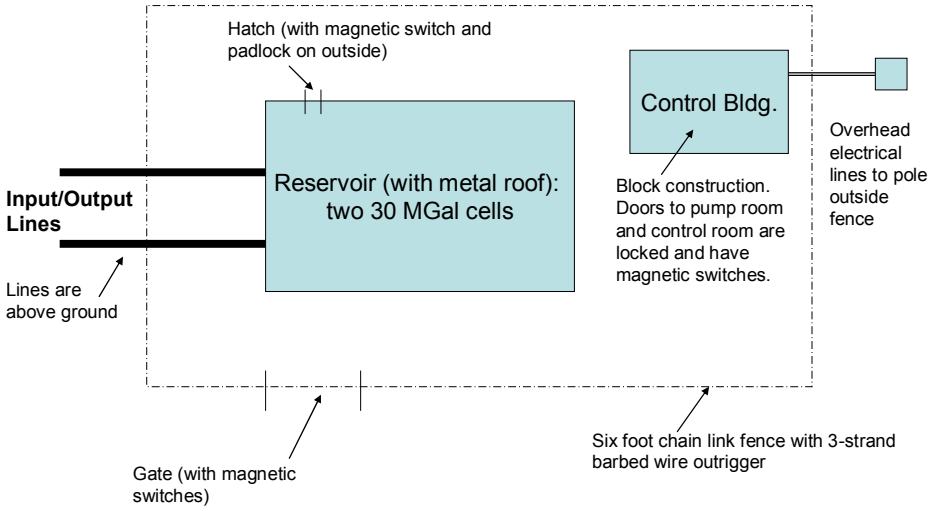

Figure 53. Notional Layout of Reservoir Facility

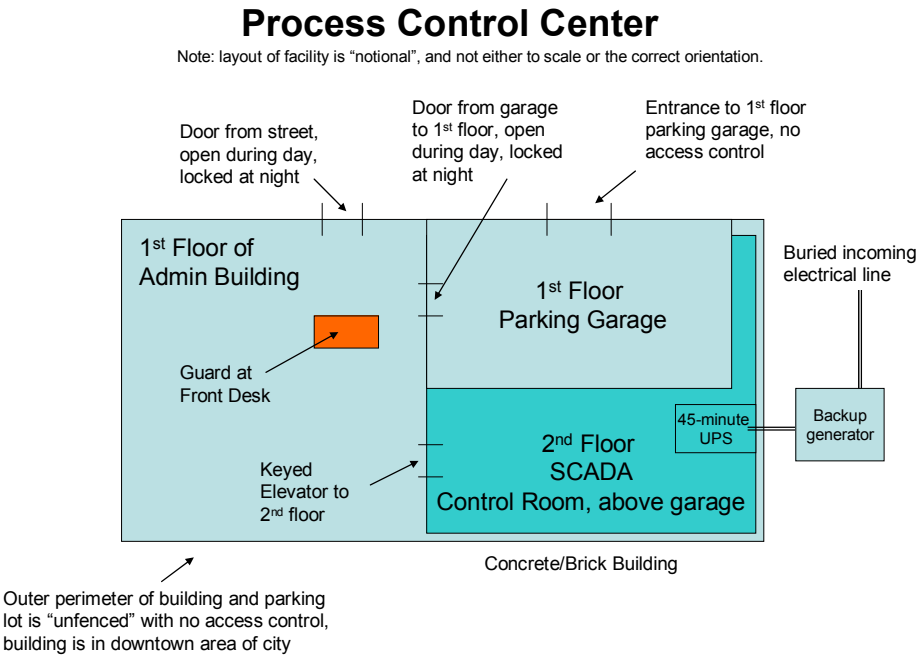

Figure 54. Notional Layout of Process Control Center

Finally, the overall network architecture of the SCADA and the cyber-security controls were determined using information available in the RAM-W report. Assumptions based on expert opinion were made to fill in any gaps in necessary data. The cyber security primitives were found to be relatively weak, with the best security measures rating only a Category II. Refer to Section 5.6.2.2 for further details on the cyber security primitive category ratings.

The network topology is shown in Figure 55 and the security primitive ratings are described below.

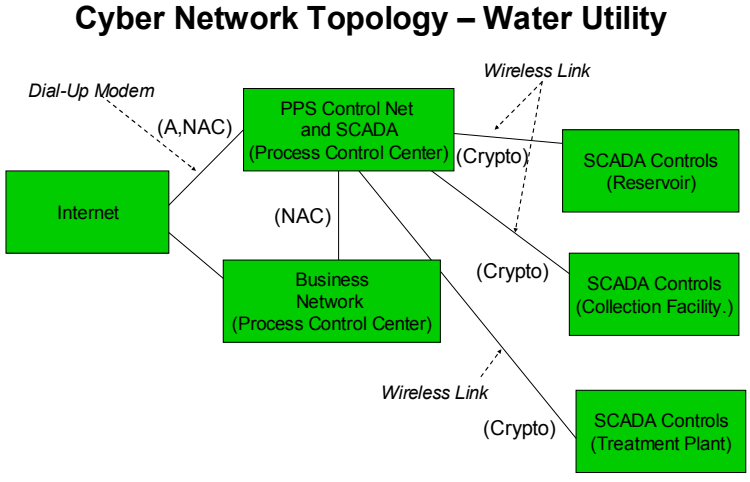

Figure 55. Water Utility Network Topology

The water utility had relatively poor cyber security at its process control center. The Authentication security primitive (password strength) was considered to be Category II which is "weak passwords with no periodic changes". At the time this scenario was analyzed with the CPSAM tool, only the authentication security primitive was available.

The Network Access Control (NAC) security primitives were considered to be a mix of Category I and Category II. There was remote login via password-protected dial-up connections to the SCADA LAN at the process control center. This is NAC Category I for that access path. However, there was a firewall between the administrative LAN and the SCADA LAN which was likely Category II for that access path. That category is "remote logins are allowed from the Internet but IP address filtering and port blocking is used".

User Access Control (UAC) security primitive was optimistically modeled as Category II. In that case, physical access in monitored and rights are assigned to individual users. This is somewhat optimistic since the dial-up modem uses a shared account.

The Cryptography security primitive was modeled as Category I because the system used plaintext communications over wireless channels. This is somewhat pessimistic since some of the communications between the control center and the PCS devices used VLANs over a private Intranet. Those links would be Category II for this security primitive.

The Data Integrity, Data Aging, and Logging/Monitoring/Auditing security primitives were all modeled as Category I, which means that nothing was implemented for these three security primitives.

The System Management security primitive was modeled as Category II. This entails backups and current security patches to the operating system and applications.

7.1.2 Consequence Inputs

The consequences of concern to the water utility, and the resulting approximate economic damages, are from the RAM-W report. The consequences of concern and the primary facilities where those concerns are paramount include:

- Interrupt or Reduce Ability to Tap Untreated Water — Collection Facility
- Interrupt or Reduce Ability to Treat Water Supply — Treatment Plant
- Interrupt or Reduce Ability to Store Treated Water — Reservoir
- Interrupt or Reduce Ability to Distribute Water — Process Control Center

These consequences are not exclusive to particular sites; the damaging or disabling of any site will likely have consequences in all four areas. In addition, the consequences are described in relative categories of "High", "Medium", or "Low", depending upon the length of the outage, the number of customers impacted, resulting health impacts, and the damage to the overall system. Figure 56 shows the inputs for the consequence: "Medium Loss in Ability to Store Treated Water". Based on available information, this consequence ranges between $1 million and $10 million. The monetary cost is the only information available, however, so there is no evidence to

assign greater weight to any particular portion of that spectrum. As such, the degree of evidence across the range is equivalent at 1.0. Also, there is no need to "Set Willingness to Pay" because this consequence is already measured in dollars. These features are, however, used in the High-Security Facility example described in Section 7.2.

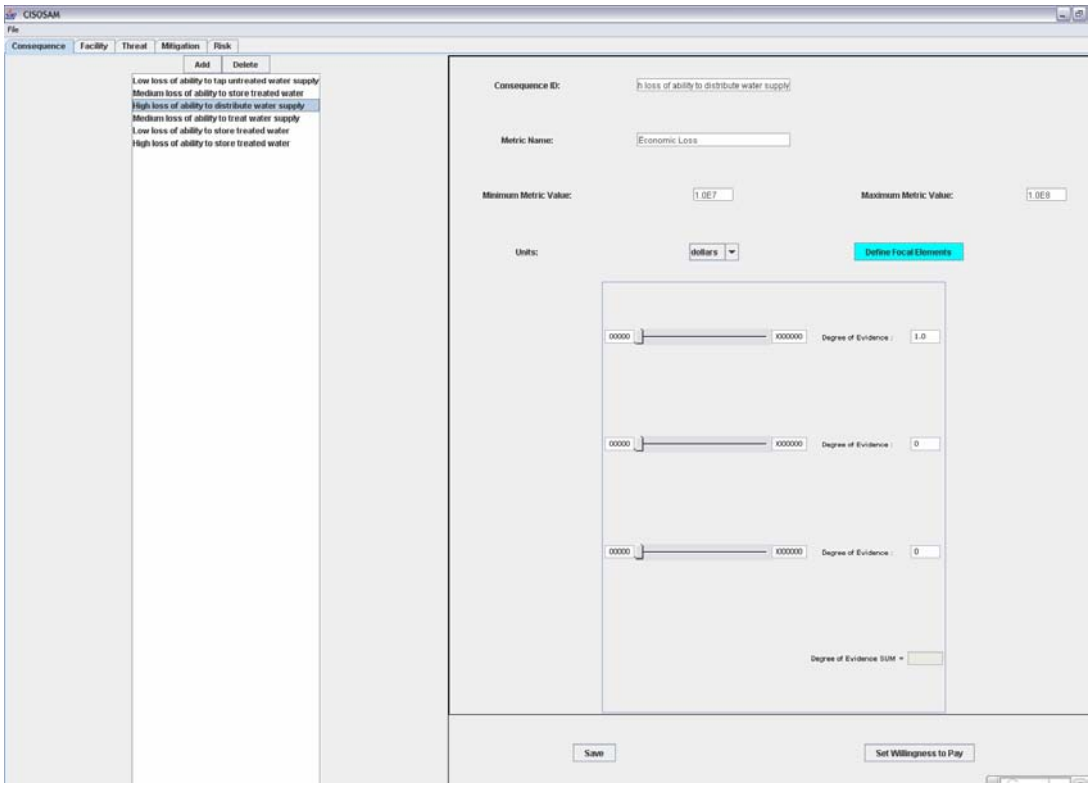

Figure 56. Water Utility Consequence Screen Inputs

A subset of the consequences analyzed for this test case is listed in Table 22. This list is merely a small group of the overall possible consequences; the consequences listed are in terms of economic damage, but the analysis could also be conducted by converting native units such as the duration of the outage or the number of water system users impacted to dollars using the Willingness to Pay feature on the Consequence input screen.

Table 22. Water Utility Consequences Analyzed

Consequence ID	Min value	Max value
High loss of ability to store treated water	$10 million	$100 million
Medium loss of ability to store treated water	$1 million	$10 million
Low loss of ability to store treated water	$1 thousand	$1 million
High loss of ability to distribute water supply	$10 million	$100 million
Low loss of ability to tap untreated water supply	$1 thousand	$1 million
Medium loss of ability to treat water supply	$1 million	$10 million

7.1.3 Cyber Inputs

The information on the physical and cyber security systems in the diagrams above, taken from the RAM-W report, were then input into the CPSAM software. The first step on the Facility input tab is to input the Cyber Location information, i.e., the cyber network topology and security primitive categories. To input this information into the CPSAM software, a graph similar to the one in Figure 55 is created, and the security primitive categories are attached to the network links.

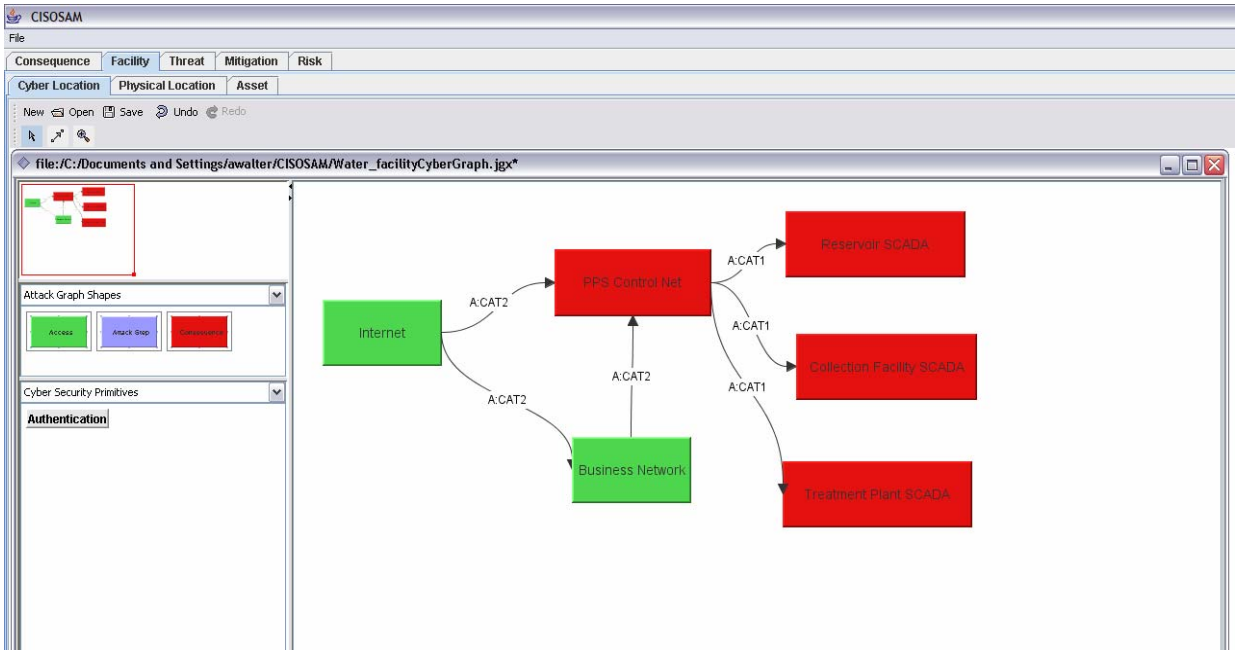

Figure 57. Water Utility Cyber Location Screen Inputs

In the case shown in Figure 57, the authentication primitive (passwords) is the only one utilized between the networks. The links from the Internet and the Business Network to the SCADA Control Network are given Category II ratings (weak passwords, no periodic changes), while the links from the SCADA Control Network to the SCADA networks at the individual facilities have Category I authentication (no passwords). A cyber attack need only breach the central SCADA Control Network, via the Internet and/or Business Network and weak passwords, to gain access to the SCADA and security controls of all the sites. This model follows the RAM-W report, which shows the relatively antiquated SCADA and security control network of the water utility.

7.1.4 Physical Inputs

The physical protection features of each facility – including locks, doors, access controls and intrusion sensors – are input into the Facility/Physical Location screen. Each facility can be input multiple times with slight variations to the Physical Location screen inputs to show the result on security effectiveness of degradation or upgrades to physical protection features. Also, changes can be made to show how effective the security system is if the adversary starts at different points in the facility, such as an insider having access to bypass the first layer of

security but not the second. The software tool has the potential to handle hundreds of these variations on a single facility or set of facilities.

An example screenshot of the Reservoir facility, modeled from the adversaries starting outside the perimeter fence and with all parameters from Figure 53, is seen in Figure 58.

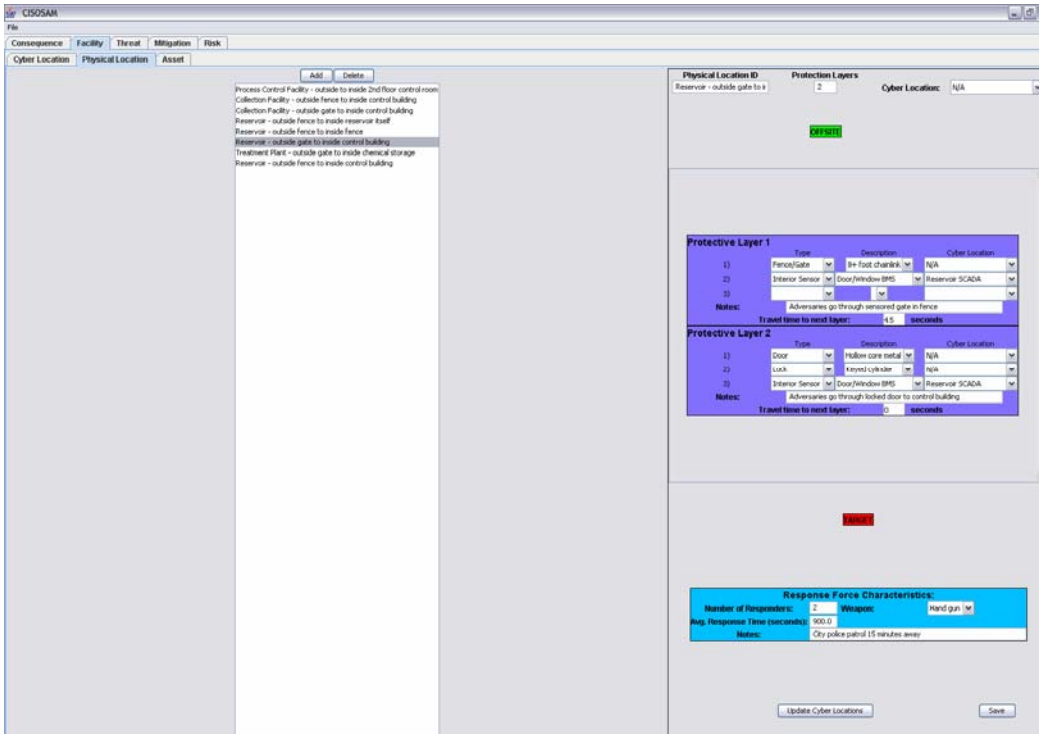

Figure 58. Physical Location Screen Input for Reservoir Control Room Access

Each physical security feature can be tied to a particular cyber network feature that can control it. In the figure above, the Interior Sensor – Door/Window BMS is controlled by the PPS Control Net through the Reservoir SCADA. This means that if the adversaries have sufficient cyber-skills to break into the PPS Control Net they can disable that Door BMS (balanced magnetic switch) sensor. By turning off this sensor, a cyber-enabled physical attack would likely have a greater chance of succeeding because the detection value of the sensor is lost.

Also on this screen, the user inputs the travel time between layers and the characteristics of the response force. In this example, the travel times between the first layer and the second layer is 45 seconds, while the travel time between the second layer and the target is 30 seconds. The response force characteristics are set to the probable response times stipulated in the RAM-W report, which is a city police patrol of two officers, armed with standard sidearms, arriving 15 minutes after the first alarm.

7.1.5 Asset Inputs

The asset input screen allows the user to combine all the necessary information inputted at prior screens – a physical location, a cyber location, a consequence, and a total task time – to define a

target asset. The physical location selection is the physical path the adversary must take to reach the asset – and the physical security safeguards that must be bypassed or overcome with physical or cyber means. The cyber location selection lists all the cyber locations previously specified on the cyber input screen. If the target asset is not cyber-controlled (i.e., the consequence of concern for the target asset cannot be accomplished through cyber means alone), the cyber location selection is left as N/A. Finally, a consequence must be associated with a successful attack on the asset and the total task time it would require to successfully complete the attack task once the adversaries have gained access to it. All of this information is then saved as an "asset". A large list of assets can be made based on the many permutations and combinations of physical locations, cyber locations, consequences, and task times. These assets are compared to the threats defined on the next screen to determine the security effectiveness of the facility. Figure 59 shows an example of one of the assets defined for the water system.

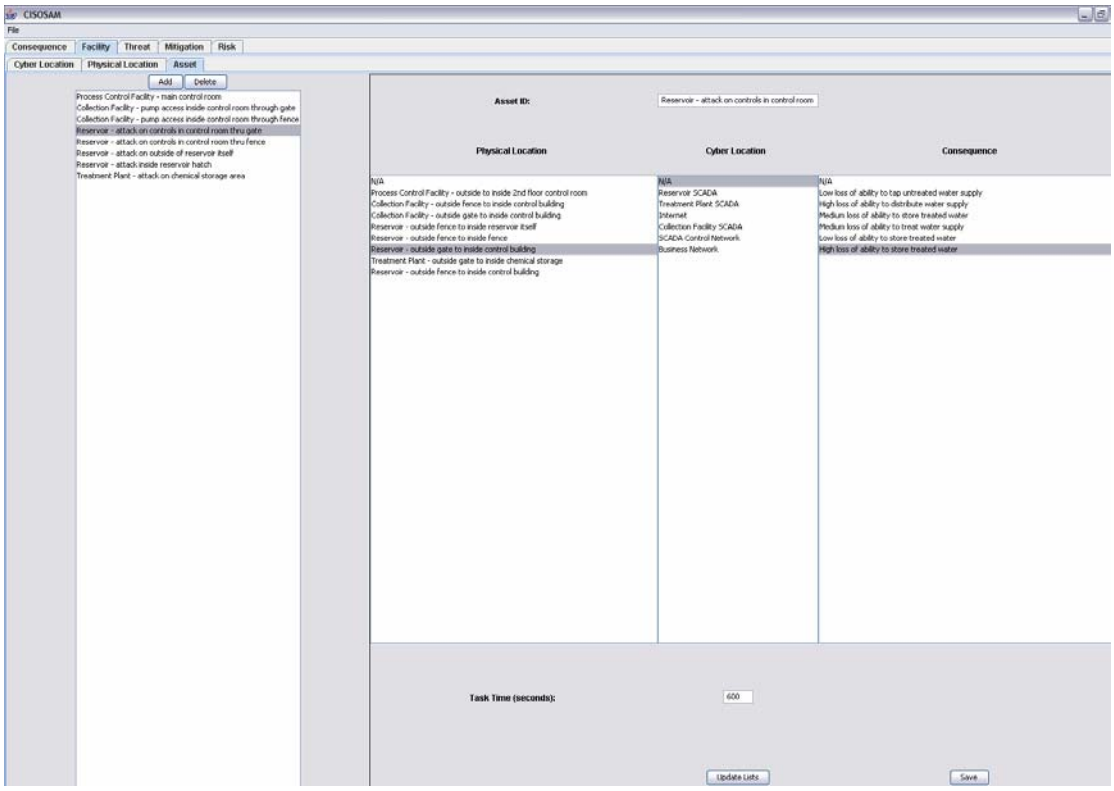

Figure 59. Asset Input Screen, Attack on Reservoir Control Building

Table 23 shows the list of assets input for the sample analysis. Many additional assets are possible; only a small group was selected to simplify the analysis and results. Also, a standard 600-second (10-minute) task time was selected for each asset as a starting point for the analysis. For this analysis, each cyber location is given a value of "N/A", meaning that the asset is not cyber-controlled and a consequence of concern cannot be caused by cyber means alone.

Table 23. List of Defined Assets for Water Utility

Asset Name	Physical Location	Cyber Location	Consequence	Task time (sec)
Process Control Facility – main control room	Process Control Facility – outside to inside 2nd floor control room	N/A	High loss of ability to distribute water supply	600
Collection Facility – pump access inside control room through gate	Collection Facility – outside gate to inside control building	N/A	Low loss of ability to tap untreated water supply	600
Collection Facility – pump access inside control room through fence	Collection Facility – outside fence to inside control building	N/A	Low loss of ability to tap untreated water supply	600
Treatment Plant – attack on chemical storage area	Treatment Plant – outside gate to inside chemical storage	N/A	Medium loss of ability to treat water supply	600
Reservoir – attack on controls in control room thru gate	Reservoir – outside gate to inside control building	N/A	High loss of ability to store treated water	600
Reservoir – attack on controls in control room thru fence	Reservoir – outside fence to inside control building	N/A	High loss of ability to store treated water	600
Reservoir – attack on outside of reservoir itself	Reservoir – outside fence to inside reservoir itself	N/A	Medium loss of ability to store treated water	600
Reservoir – attack inside reservoir hatch	Reservoir – outside fence to inside reservoir itself	N/A	Medium loss of ability to store treated water	600

7.1.6 Threat Inputs

Four threats are defined using the threat input screen seen in Table 24. These represent a categorical grouping of the threats that may be expected of a water utility target. Representing the most sophisticated attacks, the "high-level terrorist" threat combines a high level of physical attack expertise, equipment, and automatic weapons with a moderate amount of cyber skills. (See Figure 60 for an input screen of the high-level terrorist threat.) The "low-level terrorist" threat represents a smaller group of individuals, with less equipment, expertise, few cyber skills, and handguns. The "disgruntled employee" threat incorporates the lowest physical-only attack, with a single individual's equipment limited to power tools, no cyber skills, and no weapons. Finally, the "Lone Hacker" represents a fairly skilled hacker with very little inclination to physically attack a facility.

Table 24. Threats Analyzed for Water Utility

Threat name	Number of Adversaries	Tools	Cyber Threat	Weapon
High-level terrorists	4	Vehicle	Category III	Automatic
Low-level terrorists	2	Power tools	Category VI	Hand Gun
Disgruntled employee	1	Power tools	Category VI	None
Lone hacker	1	Hand tools	Category IV	None

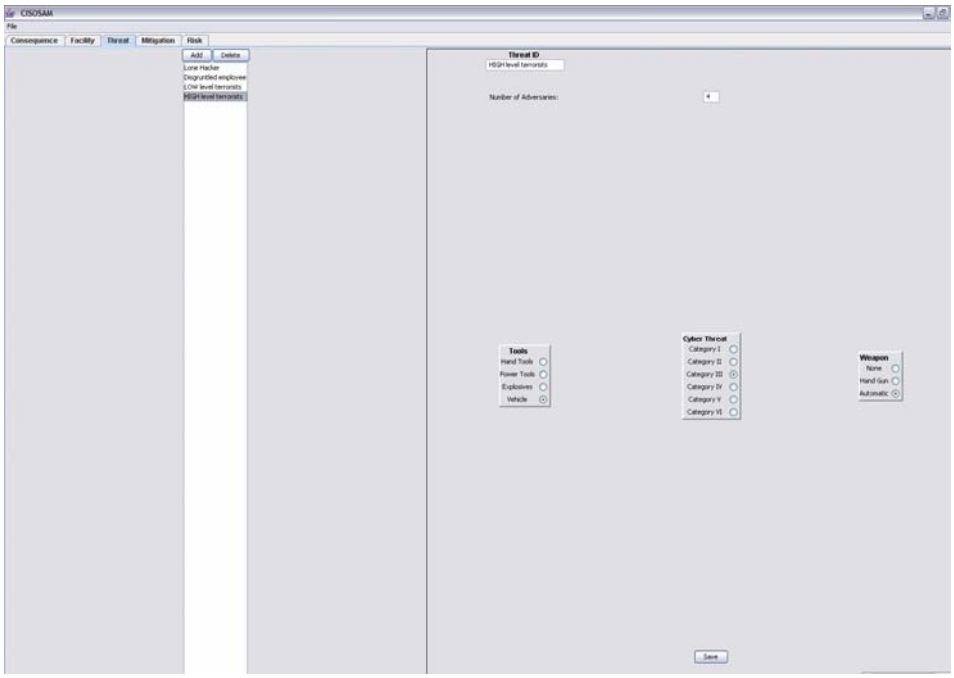

Figure 60. Threat Input Screen, High-level Terrorist Input

7.1.7 Mitigation Inputs

The RAM-W report did not include information on the mitigation of consequences after a successful attack. As such, this feature was not included in the initial test-case analysis.

7.1.8 Risk Results

Using the inputs described above, the risk to the municipal water infrastructure facilities was analyzed. The results are conditional risk, and the assumed probability of attack is equal to 1.0.

Figure 61 shows the Conditional Risk Plot for the Process Control Facility – main control room asset. The plot shows graphically what the Data Trees tab shows numerically: the cyber and physical security systems are largely effective against the Lone Hacker and Disgruntled Employee threats. The Asset Insecurity for each of these two threats is 0.17 (conversely, the

System Effectiveness for each is 0.83). The plot shows that the system effectiveness for the low- and high-level terrorist threats is much less, with asset insecurity for the low-level terrorists at 0.68 and the asset insecurity for the high-level terrorist threat at 0.95.

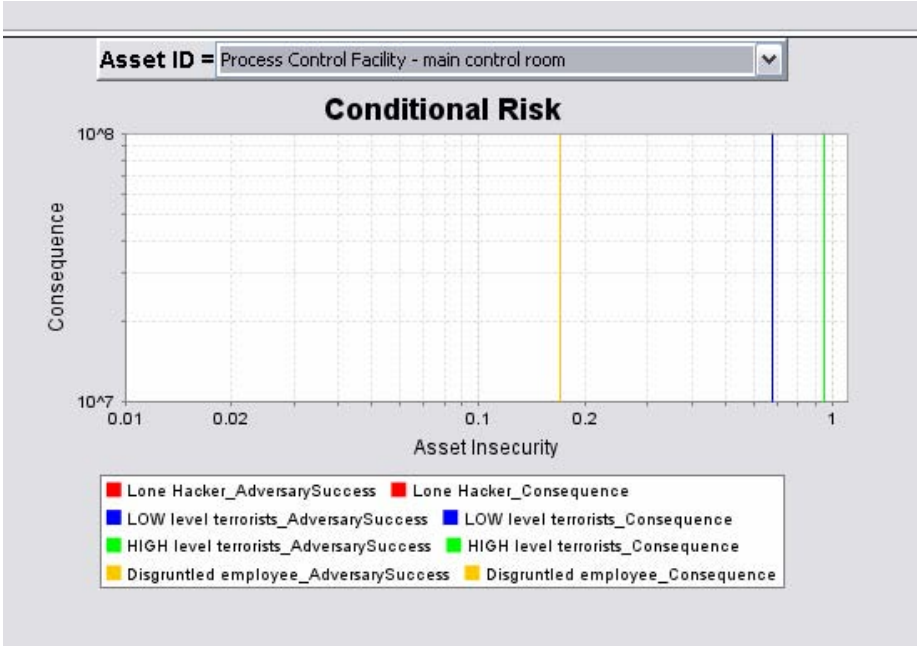

Figure 61. Conditional Risk Plot for Process Control Facility

Similar numerical and plot output for the other assets allows analysts and decision-makers to determine which assets are least secure when attacked with combined physical and cyber methods, and which have the greatest consequences associated with them.

Information of potential value to decision-makers gleaned from this simple analysis includes:

- Protecting against the highest-level threat, high-level terrorists, may not be possible and may require acceptance of that risk by decision-makers, and a subsequent review of the facility's mitigation plans. The Asset Insecurity for all assets against this threat ranges from 0.95 to 1.0. Even when the adversaries are detected and the response force arrives in time, such as with the Process Control Facility asset, the responding police patrol does not have sufficient weaponry or staffing to stop the well-armed adversaries from accomplishing their task.
- As currently designed, the system is not very effective against low-level threats, such as the Disgruntled Employee. While having little to no cyber skills and no inclination toward violence, this threat is able to enter most facilities and complete the task because the response times to most of the remote, unmanned facilities are very lengthy. At the two facilities with shorter response times, the Treatment Plant and the Process Control Facility (5 minutes and 2 minutes respectively), this low-level threat was largely ineffective in its task. Decreasing the response time to the other facilities, or dramatically increasing the task time necessary to cause a consequence of concern, will have similar propitious effects, as the software tool can show.

- As currently modeled – with assumptions based on best available information – CoCs cannot be caused by cyber means alone, which stymies the Lone Hacker threat, because they cannot accomplish their goals through only cyber means. However, if certain CoCs can be caused via the computer network, increased cyber security protections should be installed to prevent the Lone Hacker threat from successfully causing a consequence.
- With additional combinations of threats, the tool shows that a skilled hacker working in concert with a moderately skilled, non-violent physical attack team may be just as successful at attacking the water infrastructure as a full-blown, highly skilled and violent terrorist team. With a skilled hacker being able to turn off most of the physical intrusion detection sensors, an individual or team could enter the unmanned sites without risk of detection and spend as much time as necessary to complete their task. The only facilities where this is not true are those that are manned 24 hours a day, such as the Process Control Facility. This shows that cyber-enabled physical attacks can be a concern for facility providers even if a cyber-only attack cannot directly cause a consequence of concern.

7.2 High-Security Facility

The Sample Uranium Reprocessing Facility (SURF) is a fictional high-security facility that stores, handles, and processes highly enriched uranium. The facility layout and security safeguards are loosely based upon those present in real-world, high-security facilities of a similar nature. However, because the facility is fictional, more flexibility is given to test various inputs to the CPSAM software tool. Assumptions and inputs were changed based upon the circumstances that would most fully exercise the tool.

7.2.1 Facility Layout

Like most high-security facilities, the SURF has "defense-in-depth" of concentric rings of security layers. As shown in the figures below, SURF has essentially three layers of protection: the Limited Area, the Protected Area, and the Material Access Area. The first layer, the Limited Area, is surrounded by an eight-foot-high, barbed wire-topped, chainlink fence. The fence does not have sensors, but the Limited Area has random guard patrols at all times and several gates with identification (ID) checks during operational hours. The gates are locked at night and have balanced magnetic switch (BMS) detection and vehicle barriers in place when not in use.

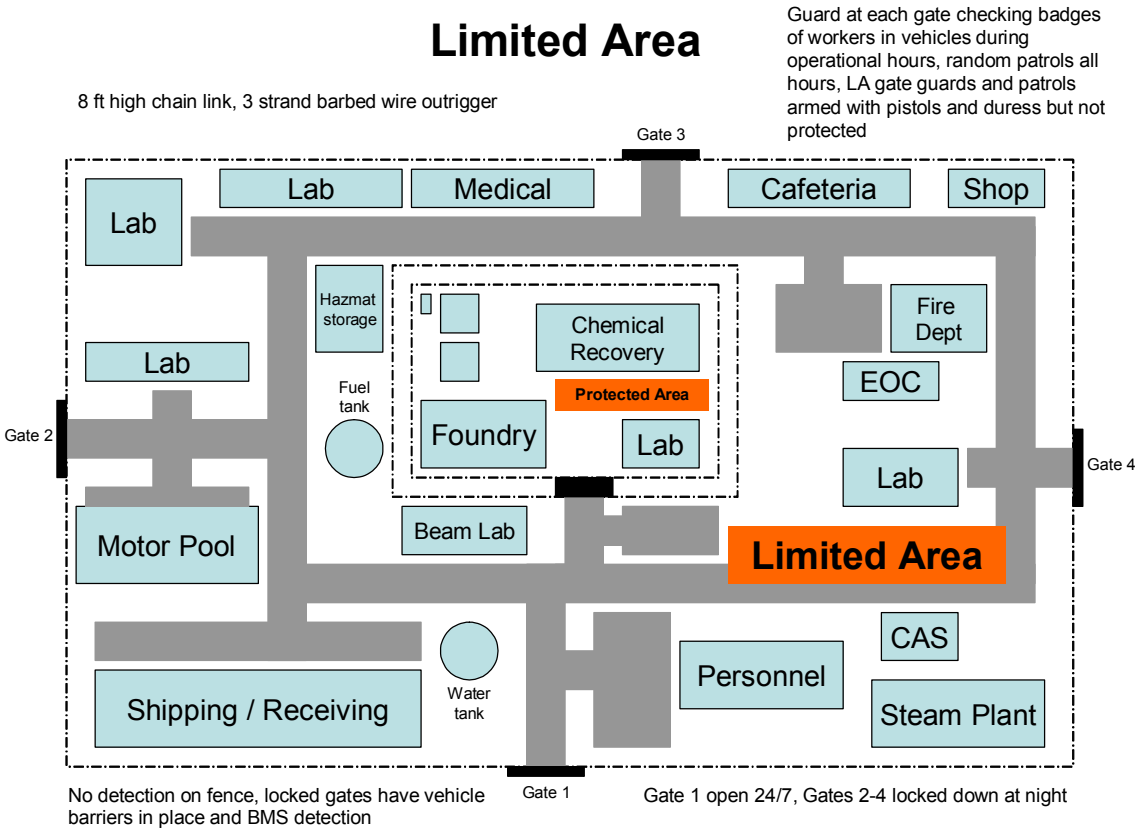

Figure 62. SURF Limited Area Layout

Inside the Protected Area (shown in Figure 63), another layer of defense exists around several critical buildings. The Protected Area is surrounded by a double PIDAS fence, which consists of an isolation zone with various intrusion detection technologies. Entry and exit to the Protected Area is accomplished through a personnel portal. Occasional official vehicles requiring access to the area are screened in a nearby vehicle portal. All personnel and vehicles are subjected to various tests for explosives, metals, and special nuclear material (SNM).

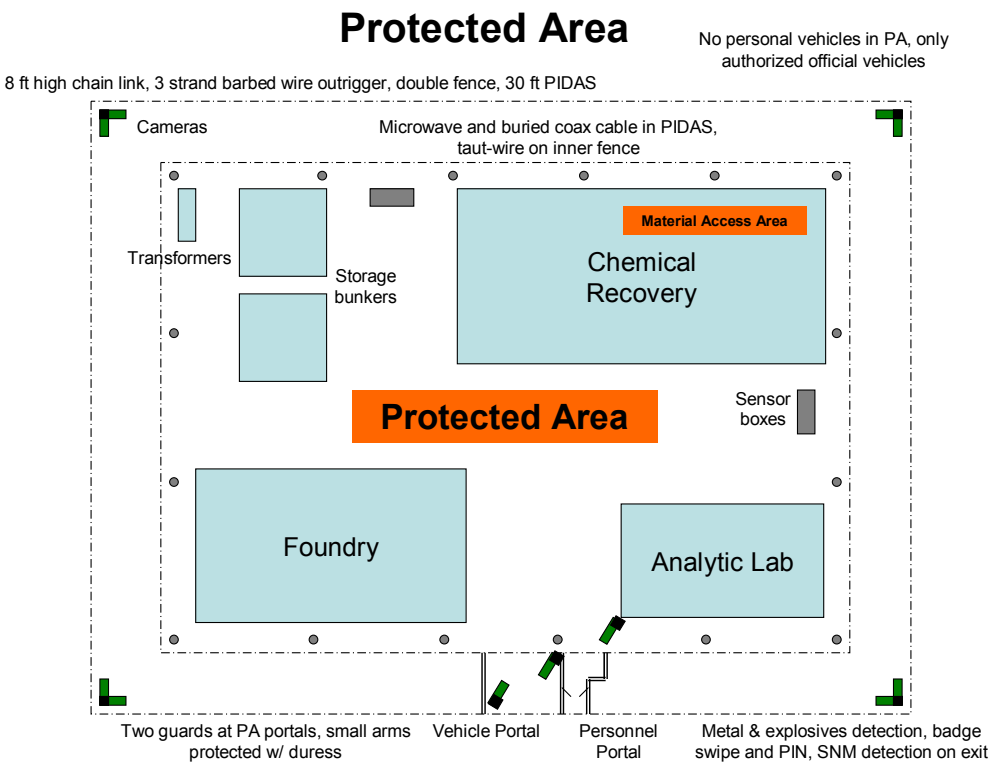

Figure 63. SURF Protected Area Layout

Finally, certain buildings within the Protected Area are given another layer of protection. Figure 64 shows the Material Access Area (MAA) of the Chemical Recovery Building. The doors and access points to the MAA have sensors, with guards and further identification verification at the entry point. A fortified vault is also present for storage of the special nuclear material (SNM) during non-operational hours.

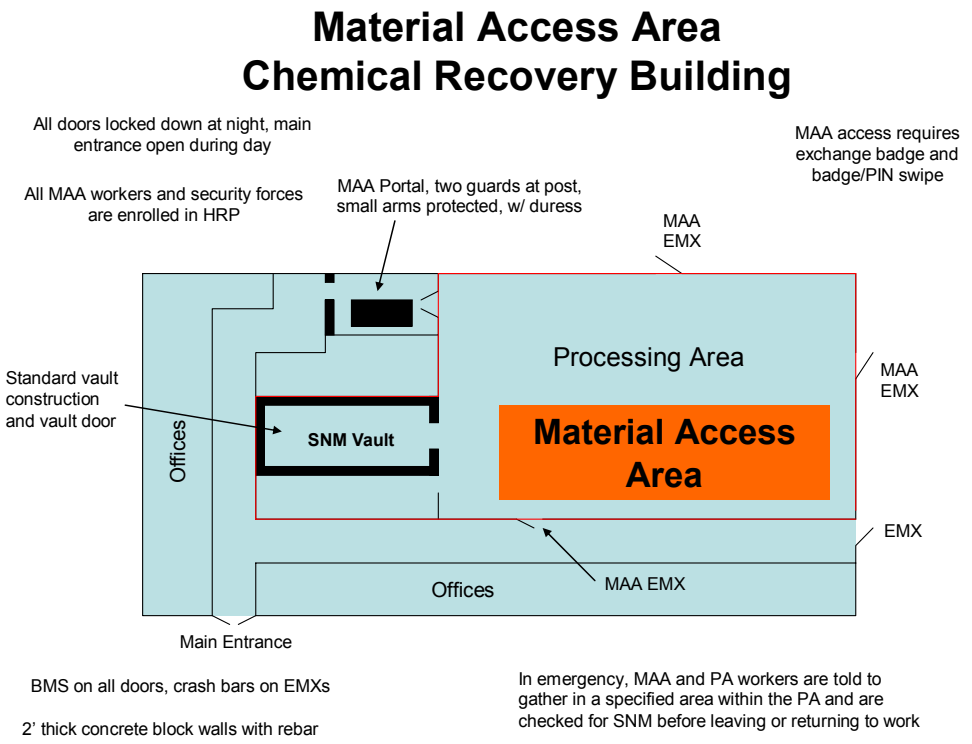

Figure 64. SURF Material Access Area Layout

With the physical layout of the SURF established, the cyber layout was examined. In most high-security facilities, the networks controlling classified information and processes, the security control network, and the open Internet or business networks are completely separate. With no physical or cyber connections between the networks, they are "air-gapped". The cyber network topology shown in Figure 65 shows no connections between the three networks present at the SURF.

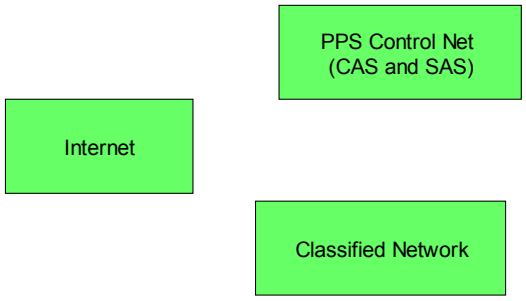

Figure 65. SURF "Air-gapped" Network Topology

Air-gapped systems enhance cyber security because an adversary must gain physical access to the security network – usually only possible from within the facility – to alter or bypass physical security safeguards controlled by the network. But, if an insider adversary were to gain access to the security network, or simply bridge the network such that the "air gap" was bridged and the security network could be accessed from outside the facility, a cyber-enabled physical attack could be devastating. Figure 66 shows a simple diagram of the compromised insider-bridged SURF networks.

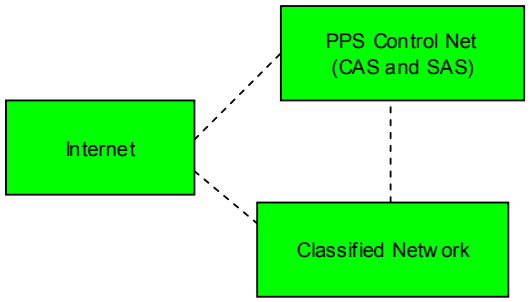

Figure 66. SURF Insider-bridged Network Topology

Each of these cases was analyzed to examine how the security effectiveness would change given partial or full cyber-access to the entire security control network.

7.2.2 Consequence Inputs

The possible consequences of an attack on a facility such as SURF are numerous and widely varied. Consequences could include theft of classified documents/material, theft of various forms of SNM, employee health impacts or deaths, contamination of facilities or surrounding populations from radiological dispersal device (RDD) detonations, political and social impacts to the nation, and more. Many of these consequences are not direct economic damage or reconstruction costs and cannot be easily defined in units of dollars. The CPSAM software helps elicit from the user the information needed to convert the more nebulous, qualitative consequence units into dollars.

For the SURF test case described, the two consequences of concern modeled are RDD detonation inside the facility and RDD detonation outside the facility. Two likely goals of an adversary attacking a facility like SURF would be to acquire SNM and either detonate it within the building – thus contaminating the building and requiring extensive cleanup – or take it outside the building to detonate an RDD and contaminate much of the site and perhaps the surrounding countryside and population.

In this model, cleanup costs of a detonation within the building can be quantified fairly easily in terms of dollars: a minimum value of cleaning up a single, lightly contaminated room all the way up to a maximum value of removing and replacing the entire contaminated building. Because the adversary team would have to hand-carry the explosives needed into the building, it is

considered more likely that they would only be capable of accomplishing a small RDD. This information is translated into the software tool via the "Degree of Evidence" inputs. A greater degree of evidence is given to the lower end of the scale than the higher as seen in Figure 67.

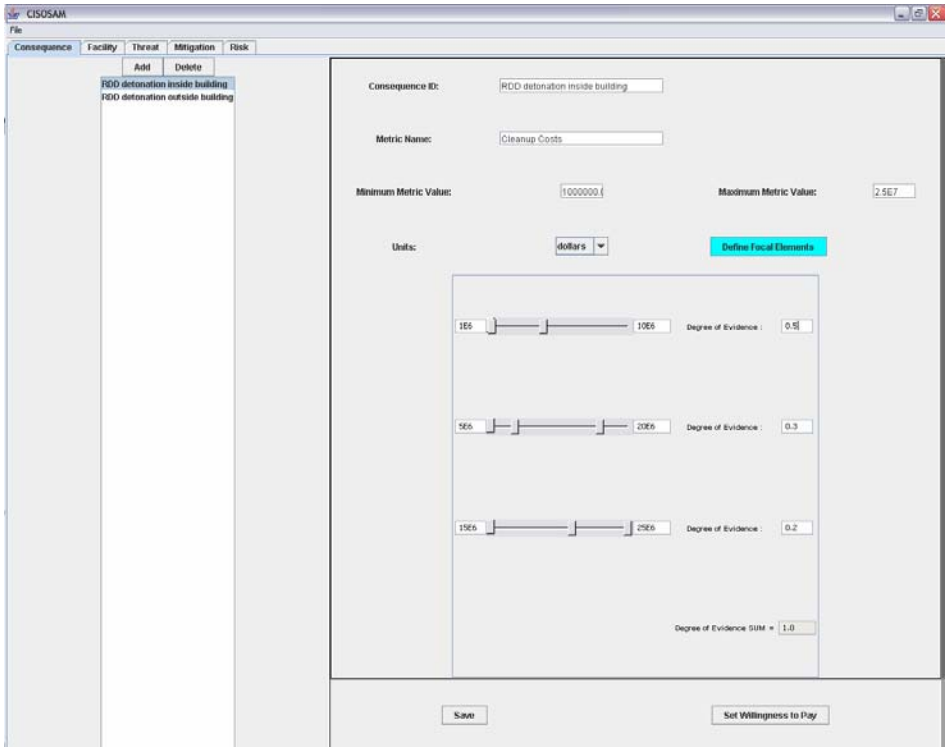

Figure 67. Degree of Evidence Inputs for RDD Inside Consequence

Total costs for an RDD detonated outside a building are more difficult to quantify. Perhaps easier to quantify are the number of people affected by the contamination. As modeled, the detonation would likely affect between 50 (the number of people working in the immediate vicinity of the Protected Area onsite) and 5,000 (the number of people working on the entire site plus the civilian population living in the immediate vicinity of the SURF). Again, it is deemed more likely that the number of people affected would be at the lower end of this scale, so the Degree of Evidence is set higher for the lower end. But because the unit of measure used for this consequence is people, the "Set Willingness to Pay" feature must be used to convert to dollars. This conversion would likely be hotly debated with a discussion between many stakeholders; for this analysis, a set of somewhat arbitrary numbers was used to make a conversion graph (Figure 68) that would likely be similar to real-world numbers.

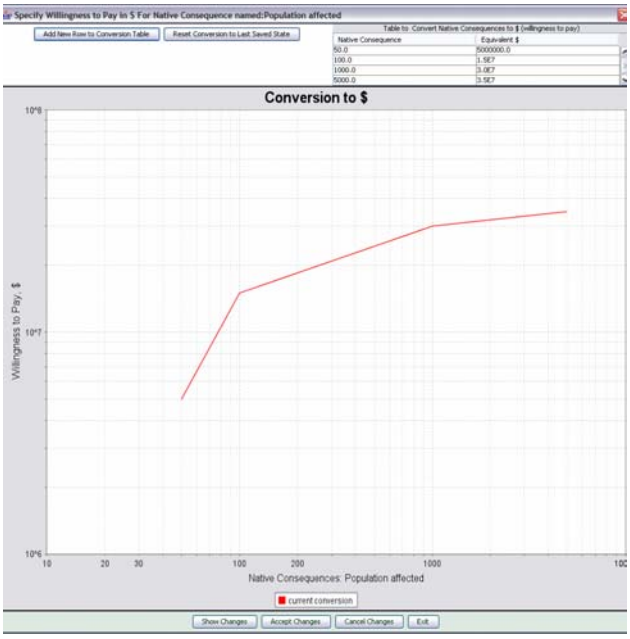

Figure 68. Willingness to Pay Conversion Plot for RDD Outside Consequence

7.2.3 Cyber Inputs

The inputs for the Cyber Location screen are based upon information described in Figure 65 and Figure 66. For the air-gapped cyber topology, the boxes representing the different networks are kept unlinked. For the insider-bridged topology, a link is established between the Internet access point and the PPS Control Net. A fairly high cyber security posture is established between the two, as it will require significant resources – either the recruitment/placement of one or more insiders, or stealth/subterfuge to access the facility – to successfully bridge the networks. This will obviously require adversaries to have a very high level of cyber security expertise to accomplish, and may limit the effectiveness of adversaries with lesser cyber skill.

7.2.4 Physical Inputs

Three physical locations are defined for the SURF analysis. All final target locations are within the Chemical Recovery Building's MAA, though different adversary paths are taken to reach it. The first defined physical location starts with the adversaries already within the Limited Area, with the assumption made that they could bypass the relatively light security surrounding the Limited Area. In this case, the adversaries must still negotiate the security of the Protected Area and MAA layers by proceeding through the PIDAS fence and an emergency exit door, respectively. The second case requires the adversaries to go through the same portions of the PA and MAA security layers, while starting outside the Limited Area fence. These two will be compared to see how much value the Limited Area security is adding to the system. Finally, the third physical location defines an adversary path from outside the entire facility to within the MAA by first traversing the outer fence, the PIDAS, and then breaching the wall of the MAA. The second and third case will allow a comparison on MAA breaching methods.

7.2.5 Asset Inputs

With two consequences and three physical locations defined, a total of six assets were modeled. It was decided that no consequence could be caused by cyber-means alone, so the Cyber Location for all assets is N/A. Task times were based on the consequence selected: the RDD inside the building would likely take less time to construct and accomplish and is given a task time of five minutes (300 seconds) while an RDD detonated outside the building would necessarily take more effort to successfully accomplish and is given a task time of 10 minutes (600 seconds).

Table 25 List of Defined Assets for SURF

Asset Name (SNM Sabotage)	Physical Location	Cyber Location	Consequence: RDD Detonation	Task time (sec)
Enter through EMX – RDD outside	Enter MAA through EMX	N/A	**Outside** facility	600
Enter through wall – RDD outside	Enter MAA through wall	N/A	**Outside** building	600
Enter through EMX – RDD inside	Enter MAA through EMX	N/A	**Inside** building	300
Enter through wall – RDD inside	Enter MAA through wall	N/A	**Inside** building	300
Adv inside LA – Enter thru EMX – RDD outside	Adv inside LA – Enter MAA through EMX	N/A	**Outside** building	600
Adv inside LA – Enter thru EMX – RDD inside	Adv inside LA – Enter MAA through EMX	N/A	**Inside** building	300

7.2.6 Threat Inputs

The threats analyzed for the SURF (Table 26) are similar to those analyzed for the Water Utility, but because the facility is hardened the adversary teams are given greater numbers.

Table 26 Threats Analyzed for SURF

Threat Name	Number of Adversaries	Tools	Cyber Threat	Weapon
High-level terrorists	8	Vehicle	Category III	Automatic
Low-level terrorists	4	Explosives	Category VI	Hand Gun
Anti-nuclear Activists	2	Power tools	Category VI	None
Lone hacker	1	Hand tools	Category IV	None

7.2.7 Mitigation Inputs

Mitigation was not included in the test-case analyses.

7.2.8 Risk Results

First, an assessment of the value of the Limited Area protection layer to the overall security system was made. By comparing the "Adv inside LA – Enter thru EMX – RDD outside" asset and the "Enter thru EMX – RDD outside" asset, it was seen that there was no significant change in the final risk results when the adversaries started from outside the Limited Area or within the Limited Area. Because the Limited Area had so little detection value, its layer of security was providing no benefit to the final security effectiveness.

Also, an evaluation was made of the effect of the adversary path going through the wall of the MAA versus through the Emergency Exit (EMX). It was found that this only affected the lower level threats. The Anti-nuclear Activists and Lone Hacker threats had relatively low Asset Insecurity values of 0.02 when going through the EMX; when forced to go through the wall of the MAA, this value dropped to zero. This is likely because the equipment available to these threats, i.e., power tools and/or hand tools, are not sufficient to breach the wall.

The risk for the SURF was also analyzed against two cyber security configurations: air-gapped and insider-bridged. On the air-gapped configuration, the cyber skills of the adversary meant very little; only the physical attack attributes were important to the final result. Accordingly, the system is very effective against the Lone Hacker and Anti-nuclear Activists threats – which have very little physical capability – as seen in Figure 69. The terrorist threats with much greater physical assault capability are much more effective at defeating the security system.

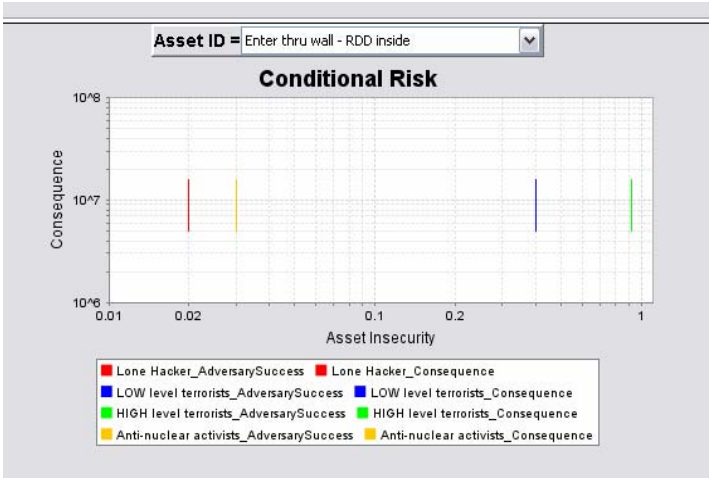

Figure 69. SURF Air-gapped Cyber Configuration Results

When the insider-bridged cyber-security topology is analyzed, the threats with significant cyber-skills should be more successful in penetrating the overall security system. Figure 70 shows the same asset and threats as the previous figure, but with an insider-bridged link between the Internet and the PPS Control Net. This link is given a Category 4 authentication protection to simulate the difficulty in accomplishing this bridge. The only threat that has appreciably improved its chance of success in attacking the target is the Lone Hacker, which has increased from a 0.02 asset insecurity to a 0.43. The low-level terrorists and Anti-nuclear Activists do not have the

requisite cyber capabilities to take advantage of a potential insider-bridged cyber topology to shut down elements of the PPS.

Interestingly, the high-level terrorist threat results do not change between the two scenarios, remaining constant at 0.92. This result shows that this high-level threat does not rely upon its considerable cyber skills to accomplish its mission; it is just as successful at defeating the security system through purely physical assault as with a combined cyber and physical attack.

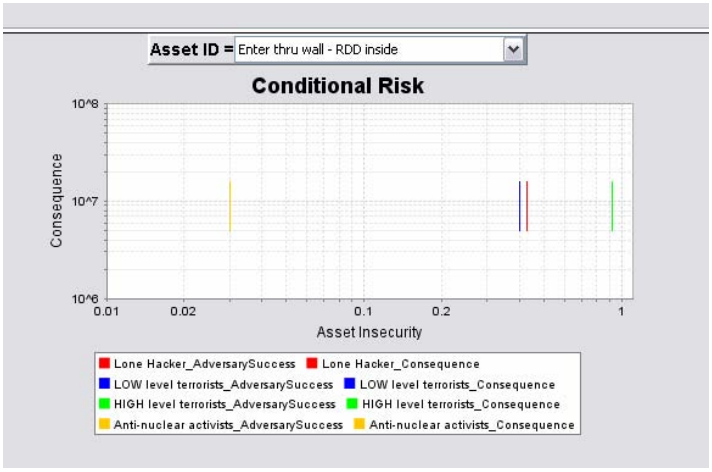

Figure 70. SURF Insider-bridged Cyber Configuration Results

Figure 71 shows the results if an asset is modified to be cyber-controlled and the insider-bridged cyber topology is kept. It shows that the Lone Hacker threat now has the greatest chance of successfully causing a consequence through cyber-only means. Because the mathematics used by the software to calculate asset insecurity of cyber-only attack is based on belief and plausibility, the asset insecurity result is now an interval instead of a point estimate. And, because the cyber-only interval covers higher values than the point estimate for a physical-only and cyber-enabled physical attack (see Figure 71), the cyber-only attack is the best option for the Lone Hacker.

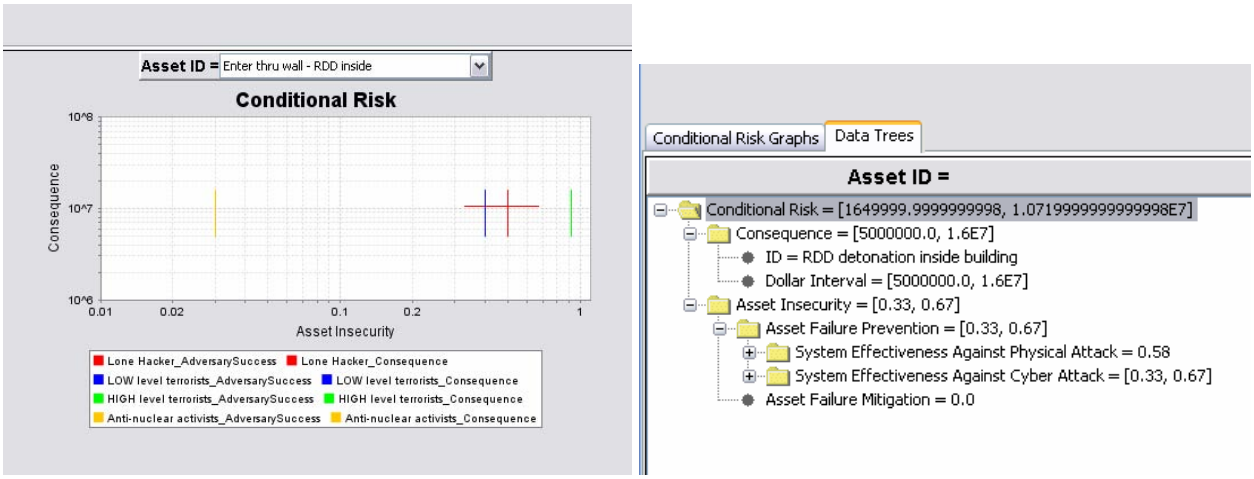

Figure 71. SURF Cyber-controlled Consequence Results

8. Summary and Next Steps

Security assessment methods, engineering failure analysis methods and risk assessment methods all have similar purposes that seek a better understanding of system failures that cause significant consequences. Physical-security assessment methods and cyber-security assessment methods also share this same purpose. However, differing historic approaches to achieve these purposes provided significant challenges to develop a risk assessment methodology that supports analysis of integrated physical and cyber security elements within critical infrastructure systems.

Analytic methods for engineering failure analysis provide a foundation for what system components and sequences of events must fail to cause a specific consequence. Yet, security assessments require additional steps to understand whether an adversary capability can overcome security systems that protect critical components. Physical security analysis uses a "detect, delay and respond" method to evaluate physical system effectiveness; however, cyber security analysis typically uses "best practices" or "red team engagements" to evaluate cyber system effectiveness. When path-based approaches (e.g., attack graphs) are used to evaluate cyber protection systems [Lippmann and Ingols, 2005; Amenaza Technologies], those analyses typically lack the quantitative details found in PPS analyses. Risk analysis requires integration of consequence and vulnerability estimates to gauge the potential impacts to critical infrastructure. Integrating select aspects from each of these methods was the key to achieve the project goal.

Through this LDRD project, the physical security and cyber security team members researched historical approaches, retained valuable aspects of those approaches, and developed a truly integrated cyber/physical security assessment methodology. Within this report, we have attempted to communicate the most important outcomes of this work, which were

- to achieve a better understanding of the cyber/physical interfaces and implications for unidentified vulnerabilities, and

- to provide a tool for decision makers that shows integrated and comprehensive risk results for "blended" security systems that can contain both cyber and physical elements.

The project team recognized that not all security systems are of similar sophistication, nor should they be. Security systems for low consequence impacts or where mitigation might provide adequate risk management could be evaluated with a best practices or screening analysis method. A best practices questionnaire analysis tool (CICSTART) was developed to evaluate both physical and cyber security practices. Conversely, high consequence impacts or difficult to mitigate risks often have more sophisticated security systems, which require functional or engagement style security systems analysis. This project created a functional style security assessment method (CPSAM) that integrates cyber and physical security systems as a software application.

The CPSAM functional risk assessment methodology combines the fundamentals of physical protection systems (e.g., detect, delay and respond) with cyber protection system primitives that are based on opportunistic pathway analysis. The methodology begins with a fundamental risk principle where the analysis selects specific consequences of concern (CoC) so resources are not wasted looking at inconsequential impacts. Specific key asset failures that could lead to those consequences are identified by external analysis methods as these are often tailored to the complexities of the specific infrastructure. The capabilities of the adversary attacking the facility are contrasted with the protective features at the facility to estimate the likelihood of adversary success. The key cyber-physical security integration step occurs in the portion of the vulnerability assessment model, where the performance of protection elements that are cyber-controlled are turned off to account for the likelihood that an attacker could penetrate the cyber protection system. Since there is insufficient data to support a probabilistic approach to cyber security assessment, a novel application of a broader mathematical tool (Belief and Plausibility) was developed to support vulnerability estimates for attacks that include cyber elements.

While the CPSAM is an operational alpha-version software product, additional developments of the methodology and software features are needed in the following areas:

- Develop user interfaces to efficiently elicit the data needed to apply the model.
- Develop graphics to display risk values for different types of attacks and different consequences of concern.
- Formalize the application of the methodology to multiple targets.
- Automate the identification of target sets using appropriate engineering process models for various types of infrastructures.
- Develop an improved method for assessing mitigation effectiveness.
- Improve the techniques used to evaluate the effectiveness of the cyber protection systems.

This page intentionally left blank.

Appendix A:
Belief as Measure of Uncertainty

This appendix summarizes the mathematics of the belief/plausibility measure of uncertainty and also summarizes belief/plausibility for fuzzy sets. The information in this appendix is excerpted from a Sandia technical report. [Darby 2006]

The axioms for belief require that the number of focal elements (defined later) for a universe of discourse be countable. Sections A.2 and A.3 of this report address discrete sets. Section A.4 discusses intervals of real numbers.

A.1 Value and Uncertainty

A random variable is a real-valued function defined on a sample space. [Dougherty 1990] The values for a random variable can be represented as a set of all possible numerical values; for example, $X = \{x \mid x \text{ an element of } [0,1]\}$.[22] The uncertainty for a random variable can be expressed by assigning a "likelihood" to events in its set. Therefore, a complete description of the random variable consists of two parts: (1) the set of all possible values and (2) an uncertainty measure on that set. A random vector is a combination of random variables and the random vector has a set of values (tuples). The name convolution is used to denote the combination of uncertainties of random variables into an uncertainty for a function defined on the random vector.

Consider two discrete random variables with ranges defined as follows: [23]

$$\begin{aligned} X &= \{x_i \mid i = 1,2,...,n\} \\ Y &= \{y_j \mid j = 1,2,...,m\} \\ X\,x\,Y &= \{< x, y > \mid x \in X, y \in Y\} \end{aligned} \qquad \text{(Eqn. A-1)}$$

where x and y are real numbers. The random vector is the Cartesian product X x Y.
A subset of the Cartesian product X x Y is called a relation. In the remainder of this discussion, reference to the random variable X means the range for X.

We are interested in a function defined on a random vector that maps to the set of real numbers, $f: X$ x $Y \rightarrow$ Reals. For example we may wish to perform addition, $X + Y$, or multiplication $X * Y$.[24] Let $f(x,y) = z$. The mapping f produces the solution:

---

[22] To be precise, X is the range for its corresponding random variable; see Section A.4. The set contains all possible unique outcomes for the random variable. The elements of the set are mutually exclusive.

[23] <> denotes a tuple whereas {} denotes a set; a tuple is an ordered collection and elements can be repeated, a set is an unordered collection and elements cannot be repeated. Uppercase is used for a random variable and lowercase is used for a value of the random variable; for example, X is a random variable and x is a specific value for X.

[24] Let E and F be events and let P(E) and P(F) be the probability (frequency) for these events. P(E) and P(F) can be considered as random variables over [0, 1] and the probability (frequency) for combinations of these events must be evaluated using the axioms of a probability measure. For union of E and F, P(E ∪ F) = P(E) + P(F) – P(E ∩ F). If E and F are mutually exclusive P(E ∩ F) = 0; if X and Y are independent P(E ∩ F) = P(E)*P(F).

$$Z = \{z \mid f(x,y) = z, x \in X, y \in Y\} \quad \text{(Eqn. A-2)}$$

Note that more than one <*x, y*> can have the same *z*. For example, if *f* is $X + Y$ then <2, 3> and <1, 4> both have $z = x + y = 5$.

Equation A-2 provides the values for the function of interest. We also need to generate a measure of uncertainty for each of these values by convoluting the uncertainties for X and Y. As subsequently discussed, there are measures of uncertainty besides probability, and we will use the name uncertainty to denote a general measure of which probability is one special case.[25] The mathematics for the convolution depends on the measure selected for uncertainty.

An uncertainty distribution is associated with each random variable; the uncertainty distribution specifies a "likelihood" for each value of the random variable.

Denote the power set of *X* as Pow(*X*). Pow(*X*) is defined as the set of all subsets *X* including the null set. For example, the power set of
X = {*a, b, c*} is Pow(*X*) = {null, {*a*}, {*b*}, {*c*}, {*a, b*}, {*a, c*}, {*b, c*}, {*a, b, c*}}. For a finite set with *n* elements the power set has 2^n elements. A general measure of uncertainty, *U*, is a mapping on the power set: U:Pow(X)→[0, 1]. Using the mathematics for the uncertainty measure, a likelihood can be calculated for each event in *X*.

A.2 Types of Uncertainty and Measures of Uncertainty

A.2.1 Ambiguity and Belief/Plausibility

Let *A* be a subset of *X*. A is also called an event for the random variable. The elements of *X* are unique values (mutually exclusive). In general events are not mutually exclusive, since subsets of *X* can have common elements.[26]

Since the sample space has unique elements, and the random variable is a mapping of the sample space to the reals, the value of a random variable will be unique, but there is uncertainty as to this value. This type of uncertainty is called Ambiguity. A measure of ambiguity is called a fuzzy measure in the literature. Ambiguity is the uncertainty in predicting the outcome of a future occurrence, such as the ultimate intensity and point of landfall for a hurricane forming in the Gulf of Mexico.

The most general fuzzy measure of interest for our evaluation of risk is Belief, which can be explained by considering degrees of evidence assigned to the elements of Pow(*X*). Let *m* denote a degree of evidence. *m* is a function defined as follows:

---

[25] To be technically correct, what we call uncertainty here is a fuzzy measure. See section B.2.1.

[26] For example let X = {a, b, c} and let event A = {a, b} and event B = {b, c}. A and B are not mutually exclusive since both contain b. A subset with only one element is called a singleton. Singleton events are mutually exclusive.

$$
\begin{aligned}
& m : Pow(X) \rightarrow [0,1] \\
& m(null) \quad = \quad 0 \\
& \sum_{A \in Pow(X)} m(A) \quad = \quad 1
\end{aligned}
\qquad \text{(Eqn. A-3)}
$$

The elements of Pow(X) for which m is greater than 0 are called the focal elements of X. The focal elements of X are the subsets (events) of X on which the evidence focuses.

In terms of degrees of evidence, Belief (*Bel*) and its dual fuzzy measure Plausibility (*Pl*) are defined as follows for any A and B in Pow(X):

$$
\begin{aligned}
Bel(A) \quad &= \quad \sum_{B|B \subseteq A} m(B) \\
Pl(A) \quad &= \quad \sum_{B|A \cap B \neq 0} m(B)
\end{aligned}
\qquad \text{(Eqn. A-4)}
$$

m(A) represents the evidence that the value of the random variable is *exactly* in A (in A only). *Bel(A)* represents the evidence that the value of the random variable is in A or any subset of A. *Pl(A)* represents the evidence that the value of the random variable is in A, in any subset of A, or any set that overlaps (is not disjoint) with A.

Bel(A) is a measure of the amount of information that implies A^C is false, where A^C is the complement of A. *Pl(A*) is a measure of the amount of information that implies A is true (i.e., does not negate A is true).

One useful interpretation is that *Bel(A)* is a measure of the degree to which *A will* happen, and *Pl(A)* is a measure of the degree to which *A could* happen.

The collection of all the focal elements with non-zero degrees of evidence form the body of evidence.

The ambiguity type of uncertainty is completely specified by the body of evidence.

Two types of ambiguity are of interest. Strife (or Discord) is present if there is more than one focal element. Nonspecificity is present if a focal element is not a singleton.

With a belief/plausibility distribution a random variable X has an expected value interval [$E_*(X)$, $E^*(X)$] given by:

$$
\begin{aligned}
E_*(X) \quad &= \quad \sum_{all\ A_i \subseteq X} \inf(A_i) * m(A_i) \\
E^*(X) \quad &= \quad \sum_{all\ A_i \subseteq X} \sup(A_i) * m(A_i)
\end{aligned}
\qquad \text{(Eqn. A-5)}
$$

where A_i is an element of Pow(X) and m is a degree of evidence.[27]

As an example of Belief and Plausibility consider $X = \{a, b, c\}$ with the body of evidence given in Figure A-1.

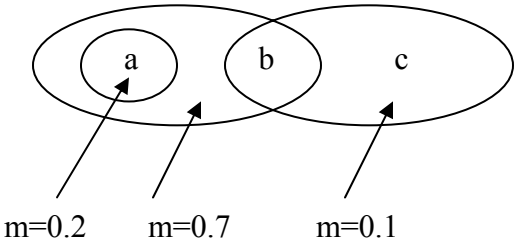

Figure A-1. Example Body of Evidence for Belief and Plausibility

The body of evidence for Figure A-1 is: $\{a\}$ with m = 0.2, $\{a, b\}$ with m = 0.7, and $\{b, c\}$ with m = 0.1. This body of evidence exhibits both Strife and Nonspecificity.

Using equation A-4, Bel and Pl can be evaluated for any element in Pow(X). Of specific interest to us are these fuzzy measures for the singletons: $Bel(\{a\}) = 0.2$, $Bel(\{b\}) = 0$, $Bel(\{c\}) = 0$, $Pl(\{a\}) = 0.9$, $Pl(\{b\}) = 0.8$, and $Pl(\{c\}) = 0.1$. Figure A-2 shows the uncertainty distribution for this case.

Let $a = 8$, $b = 1$, and $c = 6$. Using Equation A-5, the expected value interval $[E_*(X), E^*(X)]$ is [2.4, 7.8].

[27] For a finite set sup (supremum, or least upper bound) is max, and inf (infimum, or greatest lower bound) is min.

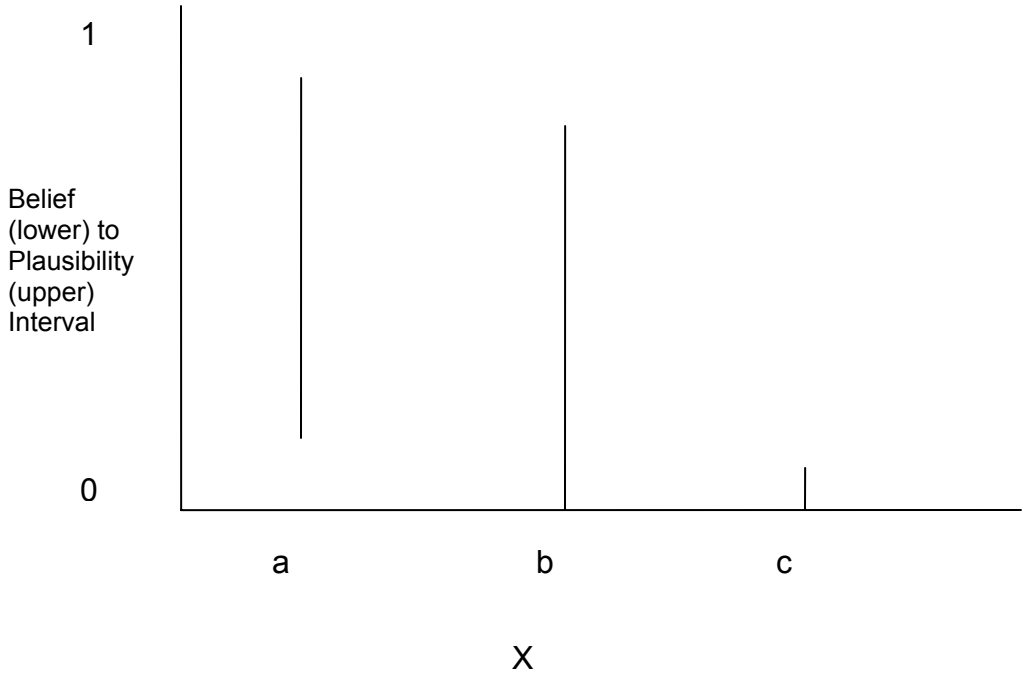

Figure A-2. Uncertainty Distribution for Body of Evidence

A.2.2 Strife and Probability

Probability is a special case of Belief. If the focal elements are singletons, then both Belief and Plausibility reduce to a common fuzzy measure, Probability. For a discrete sample space, a probability measure assigns a degree of evidence to the elements of X (the singletons of Pow(X)), and the degree of evidence for an element, m, is called the probability, p, of the element. The degrees of evidence (probabilities) sum to 1.0.[28]

The expected value, called the mean, of X is:

$$\overline{X} = \sum_{all\ x} x * p(x) \qquad \text{(Eqn. A-6)}$$

Equation A-6 is a special case of equation A-5 where $E_*(X) = E^*(X)$; that is, the expected value interval is a point value.

Figure A-3 is an example body of evidence where probability is the appropriate metric for uncertainty for $X = \{a, b, c\}$.

[28] As discussed earlier, here we are dealing with discrete sets. A probability measure requires that the probability of two disjoint events be the sum of the probabilities of each event. Since the elements of the set are mutually exclusive outcomes the probability of any event defined on the set is the sum of the probabilities of its constituent outcomes. Events are in general not mutually exclusive since they can share outcomes and are therefore not disjoint.

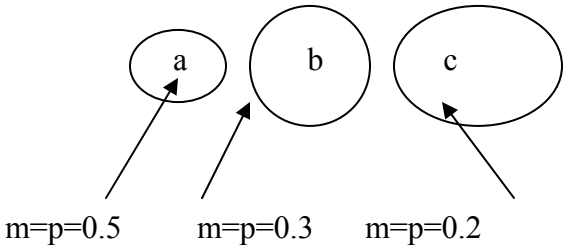

Figure A-3. Example Body of Evidence for Probability

Using either Equation A-4 or Equation A-5 where Bel and Pl are both denoted as Prob: Prob(*a*) = 0.5, Prob(*b*) = 0.3, and Prob(c) = 0.2. Figure A-4 shows the uncertainty distribution for this case.

Let a=4, *b*=13, and *c*=7. Using either Equation A-5 or Equation A-6, the expected value of the random variable is 7.3.

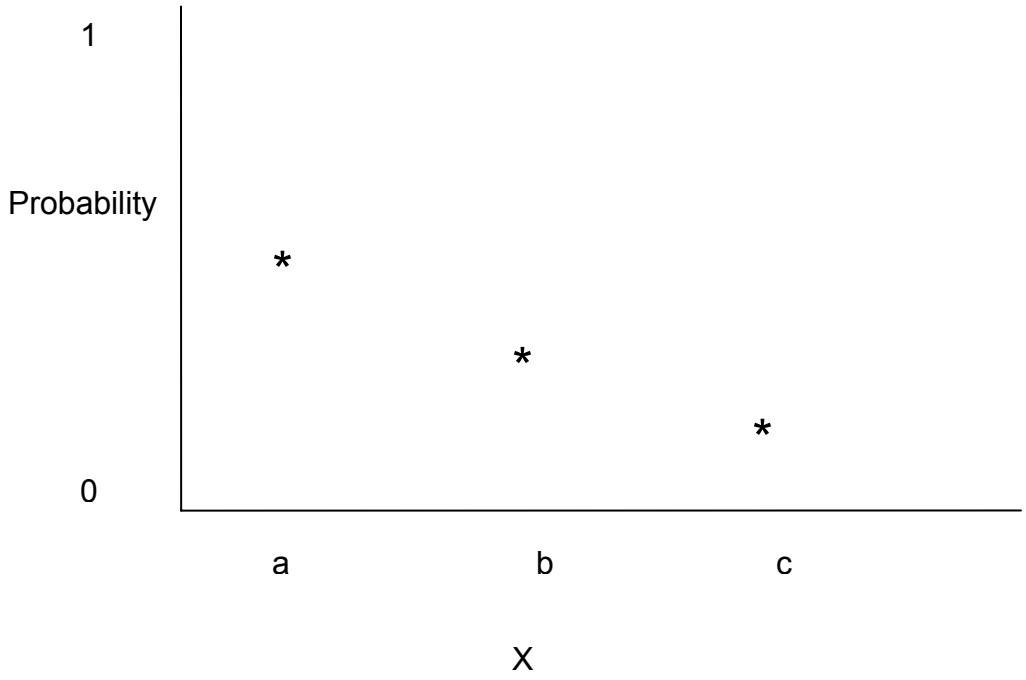

Figure A-4. Uncertainty Distribution for Body of Evidence in Figure A-3

Probability is a special case of Belief/Plausibility where there is no Nonspecificity. Probability considers Strife but does not consider nonspecificity, so it is an inappropriate measure of uncertainty where there is significant nonspecificity.

Probability is well suited to problems where the uncertainty is aleatory (random) such as tossing a cubical die known to have 1 to 6 dots on each side. Probability is not well suited to problems where the uncertainty is epistemic (state of knowledge) such as a case where we do not know how many dots are on each face of the die or even that the die is a cube.

A.2.3 Coherent Evidence and Possibility

Belief/Plausibility become Necessity/Possibility, respectively, if the focal elements are nested. The nested requirement means that for any two focal elements A and B either A is a subset of B or B is a subset of A. Possibility is applicable to situations where the body of evidence is coherent; that is, where nonspecificity dominates over strife. This is in contrast to a situation where a probability metric is applicable for which the evidence is precise but contradictory. It is important to note that necessity/possibility never reduce to probability, but belief/plausibility both reduce to probability for specific evidence.

A possibility distribution can be produced based on the degrees of evidence, and the Possibility and Necessity for any element of the power set can be calculated from the possibility distribution. The possibility distribution π is a mapping on the sample space X: π:X→[0,1].[29] Let x denote an element of X. Let Π denote the Possibility of any event A a subset of X and let N denote the Necessity:

$$\begin{aligned} \Pi(A) &= \max_{x \in A} \pi(x) \\ N(A) &= \min_{x \notin A} (1 - \pi(x)) = 1 - \max_{x \in A^c} \pi(x) = 1 - \Pi(A^c) \end{aligned} \qquad \text{(Eqn. A-7)}$$

where A^c denotes the complement of A.

A simple way to generate the possibility distribution from the degrees of evidence is to order the focal elements by increasing level of nesting; that is if the focal elements are $\{A_i \mid i = 1, 2, \ldots, n\}$ reorder and renumber the focal elements such that $A_1 \subset A_2 \subset \ldots A_n$.

With this rearrangement:

$$\pi(x_i) = \sum_{k=i}^{n} m(A_k) \qquad \text{(Eqn. A-8)}$$

where i denotes a focal element and x_i is any x that is a member of A_i. [Klir and Yuan]

For any function $f:X \rightarrow$Reals, using the Lebesgue-Stieltjes integrals the expected interval for f is:

$$\begin{aligned} E_*(f) &= \sum_i f(x_i)[\Pi(\{x_j \mid f(x_j) \le x_i\}) - \Pi(\{x_j \mid f(x_j) \le x_{i-1}\})] \\ E^*(f) &= \sum_i f(x_i)[\Pi(\{x_j \mid f(x_j) > x_{i-1}\}) - \Pi(\{x_j \mid f(x_j) > x_i\})] \end{aligned} \qquad \text{(Eqn. A-9)}$$

$[E_*(X), E^*(X)]$ is obtained using $f(x) = x$ in Equation A-9.

[29] If we have defined a random variable on the sample space, the random variable can be viewed as transforming the sample space to the reals, and the range of the random variable can serve as a surrogate sample space. [Dougherty, Probability] Therefore, X can be a random variable (a sample space on the reals) for which π specifies a possibility distribution for the values of the range of the random variable.

For a probability distribution the expected value is a point estimate given in Equation A-6. A necessity/possibility distribution (and a belief/plausibility distribution) has an expected value interval, $[E_*(X), E^*(X)]$ instead of a point estimate expected value. This is not surprising since the probability distribution over a random variable represented as a discrete set is a set of points as indicated in Figure A-4 while a necessity/possibility distribution (and a belief/plausibility distribution) over a random variable is a set of intervals as subsequently indicated in Figure A-6 for a possibility distribution (and previously indicated in Figure A-2 for a belief distribution).

As an example of Possibility and Necessity consider the body of evidence in Figure A-5 on $X = \{a, b, c\}$.

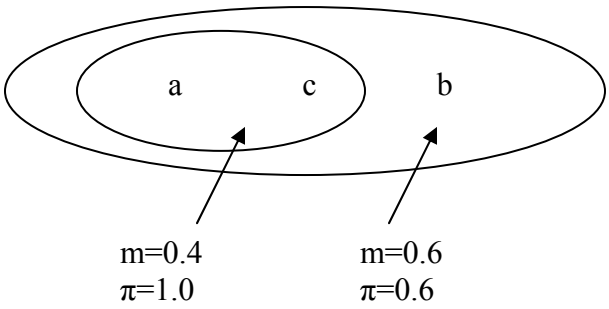

Figure A-5. Example Body of Evidence for Possibility and Necessity

Using Equations A-7 and A-8: $\Pi(a) = 1.0$, $\Pi(b) = 0.6$, $\Pi(c) = 1.0$, $N(a) = 0$, $N(b) = 0$, $N(c) = 0$. Figure A-6 shows the uncertainty distribution for this case.

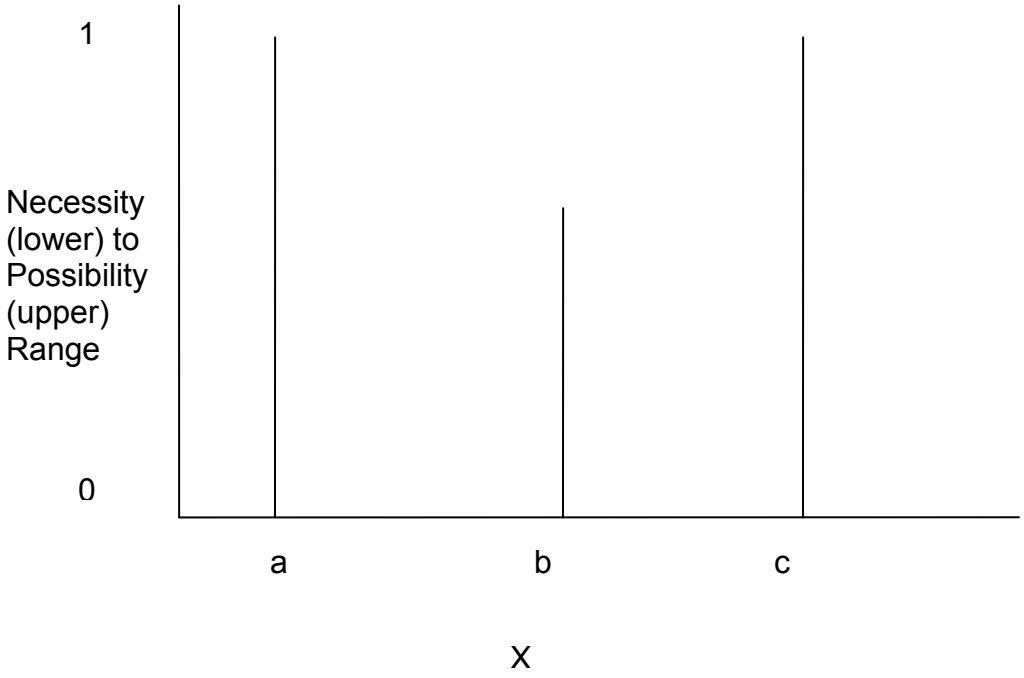

Figure A-6. Uncertainty Distribution for Body of Evidence

Let $a = 3$, $b = 8$, and $c = 2$. Using either Equation A-5 or A-9, the expected value interval $[E_*(X), E^*(X)]$ is [2, 6].

Consider the assignment of uncertainty for the frequency of an attack using a possibility metric. Using the random vector presented earlier in Section A.2.2, $F_A = \{0, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$. Since there are on the order of a million targets and on the order of 1 attack per year, for a generic target the body of evidence justifies assignment of a degree of evidence of 1.0 to the subset $\{0, 10^{-5}\}$. (A generic target is one for which specific evidence of adversary intent is not available.) If there is evidence based on intelligence that a specific target is more likely to be attacked, then a body of evidence such as that given in Figure A-7 can be produced, where the possibilities are calculated using Equation A-8.

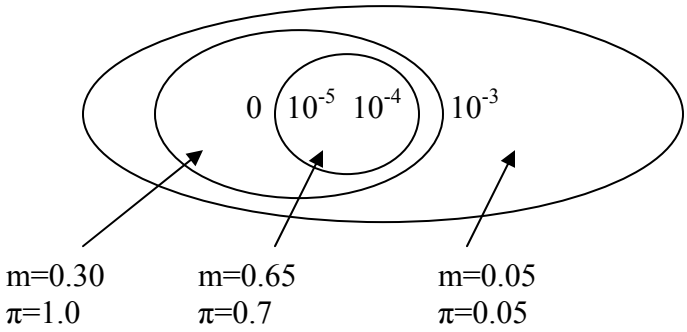

Figure A-7. Example Possibilistic Model for Threat Frequency

A.2.4 Vagueness and Fuzzy Sets

Sections A.2.1 through A.2.3 discussed various measures of uncertainty that address ambiguity; such measures are called fuzzy measures. This section addresses another type of uncertainty, Vagueness.

Whereas ambiguity deals with the uncertainty related to which value of a random variable is likely to occur, Vagueness deals with the uncertainty of how to categorize a *known* value of a random variable. Vagueness can be modeled using the concept of fuzzy sets. Note that a fuzzy measure is a different concept from a fuzzy set; a fuzzy measure addresses ambiguity while a fuzzy set addresses vagueness.

A fuzzy set extends the concept of a traditional set, called a crisp set, to include partial membership. For example for the variable $X = \{a, b, c\}$ a crisp subset is $A = \{a, b\}$. Each element in X is either completely in A or not; a and B are in A and c is not in A. A fuzzy set can have members with partial membership, for example $F = \{1/a, 0.3/b\}$ is a fuzzy subset of X for which element has a total membership and element b has partial membership of degree 0.3. Fuzzy sets are useful for modeling linguistic concepts. For example, consider the random variable for the frequency of an attack $F_A = \{0, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$ and consider the following fuzzy sets for F_A: "unlikely", "credible", and "likely".[30] Figure A-8 provides a possible definition of these fuzzy sets.

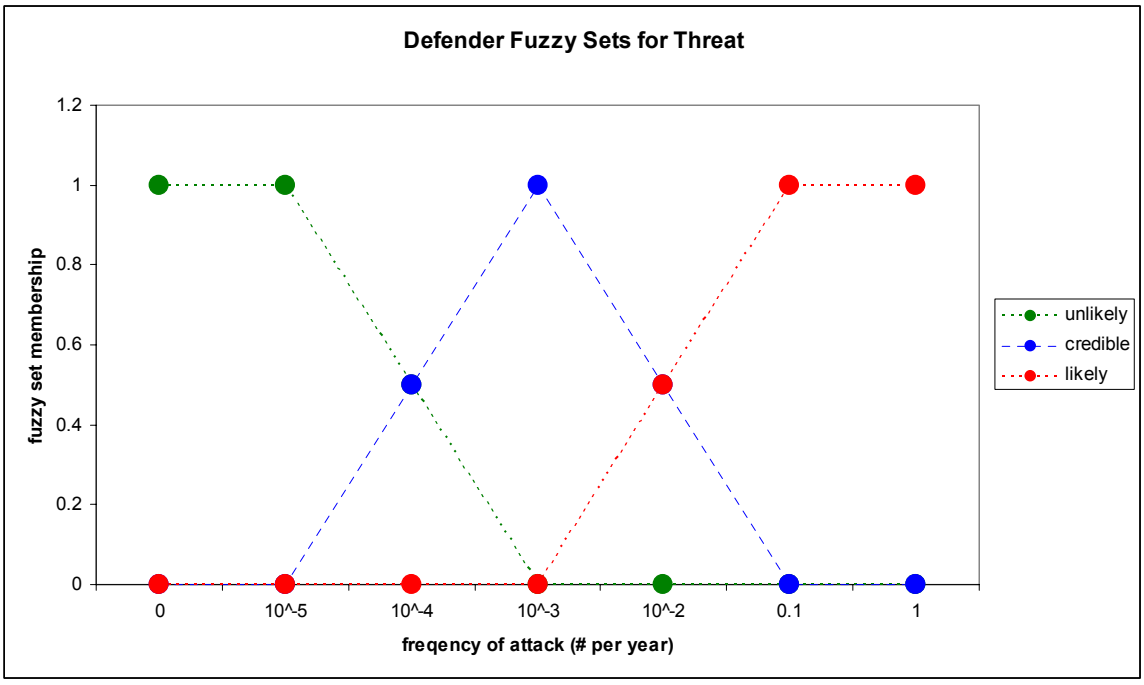

Figure A-8. Fuzzy Sets for Frequency of Attack

30 "unlikely" is a set since it is a subset of F_A and it is a fuzzy set since not every element in "unlikely" has degree of membership of 1.0; for example 10^{-4} has degree of membership of 0.5 in "unlikely". Similarly, "credible" and "likely" are fuzzy sets of F_A.

A.2.5 Probability of a Fuzzy Set

As developed in Section A.2.2 probability is a fuzzy measure that addresses a special type of ambiguity called strife. Probability can be extended to address vagueness and such an extension will be called fuzzified probability.

For a set X let $p(x)$ be the probability of element x in a discrete set X. Let F be a fuzzy set defined on X by the degree of membership $\mu_F(x)$. The probability of the fuzzy event F is:

$$P(F) = \sum_{x \in X} p(x) * \mu_F(x) \qquad \text{(Eqn. A-10)}$$

Equation A-10 weights the probability of x by the degree of membership of x in the fuzzy set of interest, F.

A.2.6 Possibility of a Fuzzy Set

As developed in Section A.2.3 possibility is a fuzzy measure that addresses ambiguity when the body of evidence is coherent. Possibility can be extended to address vagueness and such an extension will be called "fuzzified" possibility.

For a set X let $\pi(x)$ be the possibility for element x in X. Let F be a fuzzy set defined on X by the degree of membership $\mu_F(x)$. The possibility of the fuzzy event F is: (Dubois and Prade 1988)

$$\Pi(F) = \sup_{x \in X} \min[\pi(x), \mu_F(x)] \qquad \text{(Eqn. A-11)}$$

Equation A-11 is an extension of Equation A-7 to fuzzy sets; it weights the possibility for x by the degree of membership of x in the fuzzy set of interest, F.

A.2.7 Belief for a Fuzzy Set

The development in this section follows the approach discussed by Yager. [Yager 1986] Given fuzzy sets A and B for a universe of discourse X, the possibility for fuzzy set B given fuzzy set A is: [31]

$$\Pi(B \mid A) = \max_{x \in X} \{\min[\mu_A(x), \mu_B(x)]\} \qquad \text{(Eqn. A-12)}$$

where $\mu_A(x)$ is the degree of membership of the element x in the fuzzy set A.

---

[31] If A and B are crisp, "given A" means event A occurs, so $\Pi(A) = 1$ and $N(A) = 1$. $\Pi(B|A) = 1$ if A and B are not disjoint, otherwise $\Pi(B|A) = 0$. $N(B|A) = 1$ if B is a subset of A, otherwise $N(B|A) = 0$.

If A and B are fuzzy, "given A" means that the possibility distribution is determined by the degrees of membership of A; specifically, $\pi(x) = \mu_A(x)$ for all x in the universe of discourse X and: $\Pi(A) = \max_{x \in X} \mu_A(x)$, *and* $N(A) = \min_{x \in X} (1 - \mu_A(x))$. For a normalized fuzzy set (one with at least one element with a degree of membership of 1), $\Pi(A) = 1$ and $N(A) = 1$ since the fuzzy event A is the sure event. $\Pi(B|A)$ is evaluated as the overlap between fuzzy event B and the sure fuzzy event A using Equation A-12. [Dubois and Prade, Sections 1.4 and 1.7]

The necessity for B given A is $1 - \Pi(B^c|A)$ where

B^c is the fuzzy complement of B; that is, $B^c \equiv 1 - B$. This necessity can be expressed as:

$$N(B \mid A) = 1 - \max_{x \in X} \{\min[\mu_A(x), 1 - \mu_B(x)]\} \qquad \text{(Eqn. A-13)}$$

where $\mu_{Bc}(x)$ has been taken as $1 - \mu_B(x)$.

For "fuzzy focal elements" (fuzzy sets with evidence) A_i over X and for any fuzzy set B in X:

$$\begin{aligned} Pl(B) &= \sum_i [m(A_i) \cdot \Pi(B \mid A_i)] \\ Bel(B) &= \sum_i [m(A_i) \cdot N(B \mid A_i)] \end{aligned} \qquad \text{(Eqn. A-14)}$$

Equation A-14 reduces to Equation A-4 if the focal elements A_i and B are crisp sets.

If the focal elements A_i are crisp and B is fuzzy, Equation A-14 reduces to:

$$\begin{aligned} Pl(B) &= \sum_i \left[m(A_i) \cdot (\max\{\mu_B(x) \mid x \in A_i\})\right] \\ Bel(B) &= \sum_i \left[m(A_i) \cdot (1 - \max\{1 - \mu_B(x) \mid x \in A_i\})\right] \end{aligned} \qquad \text{(Eqn. A-15)}$$

A.3 Convolution

Section A.2 summarized various metrics for uncertainty, specifically: Belief/Plausibility, Necessity/Possibility, Probability, Fuzzy Sets, Fuzzified Probability, and Fuzzified Possibility.

As discussed in Section A.1, it is necessary to convolute the uncertainty measures for random variables to produce an uncertainty measure for a function defined on a random vector. Sections A.3.1 through A.3.4 assume crisp sets; Sections A.3.5 and A.3.6 address fuzzy sets.

A.3.1 Probabilistic Convolution

As presented in Section A.2, the values for a function $f:X$ x $Y \rightarrow$Reals defined on the random vector X x Y are given in Equation A-2 repeated here:

$$Z = \{z \mid f(x,y) = z, x \in X, y \in Y\} \qquad \text{(Eqn. A-2, repeated)}$$

Where $f(x,y) = z$. Let $p(x)$ and $p(y)$ be probability distributions over X and Y, respectively. The probability distribution $p(z)$ is:[32]

[32] Since the x elements are mutually exclusive and the y elements are mutually exclusive the <x,y> tuples are mutually exclusive and the probabilistic sum is an algebraic sum as indicated in the equation. As previously stated, this section deals with sets of discrete elements.

$$p(z) = \sum_{all\ x \in X, y \in Y} p(x,y) \mid f(x,y) = z\} \qquad \text{(Eqn. A-16)}$$

where *p(x,y)* is the joint probability distribution over *X* and *Y*. If *X* and *Y* are independent random variables than *p(x,y) = p(x) * p(y).*

A.3.2 Possibilistic Convolution

As presented in Section A.2, the values for a function *f:X* x *Y*→Reals defined on the random vector *X* x *Y* are given in equation A-2 repeated here:

$$Z = \{z \mid f(x,y) = z, x \in X, y \in Y\} \qquad \text{(Eqn. A-2, repeated)}$$

where *f(x,y) = z*. Let π(*x*) and π(*y*) be possibility distributions over *X* and *Y*, respectively. The possibility distribution π(z) is:

$$\pi(z) = \sup_{all\ x \in X, y \in Y} \{\pi(x,y) \mid f(x,y) = z\} \qquad \text{(Eqn. A-17)}$$

where π (x,y) is the joint possibility distribution over X and Y. If X and Y are non-interactive random variables (in the possibilistic sense) then π (x,y) = min[π (x), π (y)]. This is the "min" definition of noninteraction.

A.3.3 Convolution for Belief

As presented in Section A.2, the values for a function f:X x Y→Reals defined on the random vector X x Y are given in Equation A-2 repeated here:

$$Z = \{z \mid f(x,y) = z, x \in X, y \in Y\} \qquad \text{(Eqn. A-2, repeated)}$$

For the random vector X x Y each degree of evidence can be considered a binary relation R.[33] That is, R is a subset of X x Y with non-zero m.

Using Equation A-18 belief and plausibility for *z = f(x, y)* are:

$$\begin{aligned} Bel(z) &= \sum_{R \subseteq X\,x\,Y \mid z = f(x,y)\ and\ R \subseteq \{<x,y>\}} m(R) \\ Pl(z) &= \sum_{R \subseteq X\,x\,Y \mid z = f(x,y)\ and\ \{<x,y>\} \cap R \neq null} m(R) \end{aligned} \qquad \text{(Eqn. A-18)}$$

Following Equation A-5, the expected value interval for *f* is:

[33] A binary relation is defined as a subset of the Cartesian product X x Y. For example, if X = {a, b} and Y = {p, q} then X x Y = {<a, p>, <a, q>, <b, p>, <b, q>} and R = {<a, p>, <a, q>, <b, q>} is a binary relation on X x Y.

$$
\begin{aligned}
E_*(f:X\,x\,Y) &= \sum_{all\ R\subseteq X\,x\,Y} \inf[f(R)] * m(R) \\
E^*(f:X\,x\,Y) &= \sum_{all\ R\subseteq X\,x\,Y} \sup[f(R)] * m(R) \\
where \quad f(R) &= \{f(x,y) | < x,y > \in R\}
\end{aligned}
\qquad \text{(Eqn. A-19)}
$$

Let C denote any subset of X x Y. Using Equation A-4:

$$
\begin{aligned}
Bel(C) &= \sum_{R\subseteq X\,x\,Y | R\subseteq C} m(R) \\
Pl(C) &= \sum_{R\subseteq X\,x\,Y | R\bigcap C\neq null} m(R)
\end{aligned}
\qquad \text{(Eqn. B-20)}
$$

For each R, let R_X denote the projection of R on X and let R_Y denote the projection of R on Y.

$$
\begin{aligned}
R_X &= \{x \in X | < x,y > \in R \ for\ some\ y \in Y\} \\
R_Y &= \{y \in Y | < x,y > \in R \ for\ some\ x \in X\}
\end{aligned}
\qquad \text{(Eqn. A-21)}
$$

Define the marginal degrees of evidence m_x, the projection of m on X, and m_y, the projection of m on Y as:

$$
\begin{aligned}
m_X(A) &= \sum_{R|A=R_X} m(R)\ for\ all\ A \in Pow(X) \\
m_Y(B) &= \sum_{R|B=R_Y} m(R)\ for\ all\ B \in Pow(Y)
\end{aligned}
\qquad \text{(Eqn. A-22)}
$$

where $R|A=R_X$ means all relations R such that the projection of R onto X (R_X) is equal to A.

For any focal elements A and B in X and Y, respectively, the marginal bodies of evidence are said to be noninteractive if and only if:

$m(A\ x\ B) = m_x(A) * m_Y(B)$, and
$m(R) = 0$ for all $R \neq A$ x B.[34]

This is the "product" definition of noninteraction.

The "min" definition of possibilistic noninteraction discussed in Section A.3.2 is not a special case of the "product" definition of noninteraction. Even if the focal elements of X and Y are nested, if X and Y and are noninteractive using the product definition, the focal elements of X x Y may not be nested. That is, the product definition of noninteraction does not preserve nesting of focal elements. [Klir and Yuan, Section 7.3]

[34] The requirement that m(A x B) = m_x(A)*m_Y(B) means that for any focal elements A in X and B in Y, there is a focal element in X x Y formed by A x B with degree of evidence equal to m_x(A)*m_Y(B). The requirement that m(R) = 0 for all R ≠ A x B means that any focal element in X x Y is a Cartesian product of focal elements in X and Y.

Probabilistic independence is a special case of the product definition of noninteraction. For a probability measure, a degree of evidence is a probability for an element of the sample space, so any focal elements A and B are singletons of X and Y (call them a and b) and A x B has one element {<a, b>} with a probability P(a)*P(b).

In this report, unless stated otherwise noninteraction means the product definition of noninteraction.

Independence and noninteraction are discussed at length in a report by Ferson, et al. [Ferson et al. 2004]

As an example of convolution for belief/plausibility let $X = \{x_1, x_2\}$ and $Y = \{y_1, y_2, y_3\}$. Assume the bodies of evidence for X and Y given in Figure A-9.

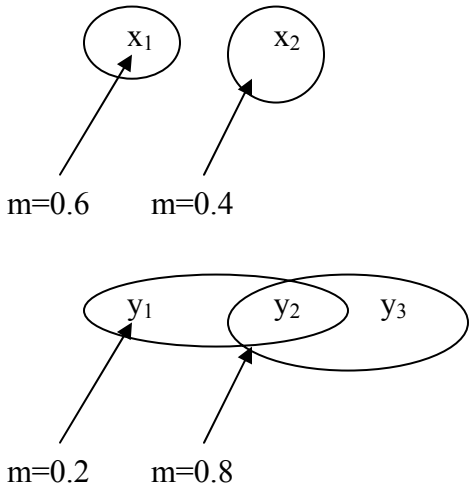

Figure A-9. Bodies of Evidence for X and Y

The random vector X x Y = {<x_1,y_1>, <x_1,y_2>, <x_1,y_3>, <x_2,y_1>, <x_2,y_2>, <x_2,y_3>}. The binary relations and the projections of those relations for the focal elements of X x Y are given in Table A-1.

Table A-1. Binary Relations and Projections for X x Y for Figure A-9

R	R_X	R_Y
$R_1 = \{<x_1,y_1>, <x_1,y_2>\}$	$R_{1X} = \{x_1\}$	$R_{1Y} = \{y_1, y_2\}$
$R_2 = \{<x_1,y_2>, <x_1,y_3>\}$	$R_{2X} = \{x_1\}$	$R_{2Y} = \{y_2, y_3\}$
$R_3 = \{<x_2,y_1>, <x_2,y_2>\}$	$R_{3X} = \{x_2\}$	$R_{3Y} = \{y_1, y_2\}$
$R_4 = \{<x_2,y_2>, <x_2,y_3>\}$	$R_{4X} = \{x_2\}$	$R_{4Y} = \{y_2, y_3\}$

The marginal degrees of evidence are given in Table A-2.

Table A-2. Marginal Degrees of Evidence

A an element of Pow(X)	$m_X(A)$	B an element of Pow(Y)	$m_Y(B)$
null	0		
$\{x_1\}$	$m(R_1) + m(R_2)$		
$\{x_2\}$	$m(R_3) + m(R_4)$		
$\{x_1, x_2\}$	0		
		null	0
		$\{y_1\}$	0
		$\{y_2\}$	0
		$\{y_3\}$	0
		$\{y_1, y_2\}$	$m(R_1) + m(R_3)$
		$\{y_1, y_3\}$	0
		$\{y_2, y_3\}$	$m(R_2) + m(R_4)$
		$\{y_1, y_2, y_3\}$	0

Assuming the bodies of evidence for X and Y are noninteractive:

$$m(R_1) = m(\{x_1\})*m(\{y_1, y_2\}) = 0.12$$
$$m(R_2) = m(\{x_1\})*m(\{y_2, y_3\}) = 0.48$$
$$m(R_3) = m(\{x_2\})*m(\{y_1, y_2\}) = 0.08$$
$$m(R_4) = m(\{x_2\})*m(\{y_2, y_3\}) = 0.32$$

and the body of evidence for X x Y is:

$$m(\{<x_1, y_1>, <x_1, y_2>\}) = 0.12$$
$$m(\{<x_1, y_2>, <x_1, y_3>\}) = 0.48$$
$$m(\{<x_2, y_1>, <x_2, y_2>\}) = 0.08$$
$$m(\{<x_2, y_2>, <x_2, y_3>\}) = 0.32$$

Using this body of evidence with Equation A-4, the belief and plausibility distributions for each element of X x Y can be calculated as summarized in Table A-3.

Table A-3. Belief and Plausibility for Elements of X x Y

Element	Belief	Plausibility
$<x_1,y_1>$	0	0.12
$<x_1,y_2>$	0	0.60
$<x_1,y_3>$	0	0.48
$<x_2,y_1>$	0	0.08
$<x_2,y_2>$	0	0.40
$<x_2,y_3>$	0	0.32

Note that unlike degrees of evidence or probability, Plausibility and Belief over the elements do not have to sum to 1.0.

Assume that $x_1 = 1$, $x_2 = 2$, $y_1 = 4$, $y_2 = 3$, and $y_3 = 2$. Let the function of interest on the random vector X x Y be $z = f(x, y) = x + y$. Table A-4 lists the <x, y> tuples and f(x, y) for each tuple.

Table A-4. Tuples and Functional Values

<x, y>	z = x + y
$<x_1, y_1> = <1, 4>$	5
$<x_1, y_2> = <1, 3>$	4
$<x_1, y_3> = <1, 2>$	3
$<x_2, y_1> = <2, 4>$	6
$<x_2, y_2> = <2, 3>$	5
$<x_2, y_3> = <2, 2>$	4

The belief and plausibility for each element of f (each unique z in Table A-4) can be calculated using Equation A-14 with the body of evidence previously calculated. Since none of the R is a subset of any <x, y> tuple, Bel(z) = 0 for all z. For z = 5, the pertinent tuples are $<x_1, y_1>$ and $<x_2, y_2>$. $<x_1, y_1>$ has non-null intersection with R_1 and

$<x_2, y_2>$ has non-null intersection with R_3 and R_4. So from Equation A-14:

$Pl(x + y = 5) = m(R_1) + m(R_3) + m(R_4) = 0.12 + 0.08 + 0.32 = 0.52$. Similarly,
$Pl(x + y = 3) = m(R_2) = 0.48$, $Pl(x + y = 4) = m(R_1) + m(R_2) + m(R_4) = 0.92$,
$Pl(x + y + 6) = m(R_3) = 0.08$.

Table A-5 summarizes the functional values, and the belief and plausibility for each value of the function f.

Table A-5. Values, Belief , and Plausibility

f(x, y) = x + y	Belief	Plausibility
3	0	0.48
4	0	0.92
5	0	0.52
6	0	0.08

The uncertainty distribution for the function of interest is summarized in Figure A-10.

The body of evidence for $f: X \times Y$ is:

$m(\{f(x_1, y_1), f(x_1, y_2)\}) = m(5, 4) = 0.12$
$m(\{f(x_1, y_2), f(x_1, y_3)\}) = m(4, 3) = 0.48$
$m(\{f(x_2, y_1), f(x_2, y_2)\}) = m(6, 5) = 0.08$
$m(\{f(x_2, y_2), f(x_2, y_3)\}) = m(5, 4) = 0.32$

so,

$m(5, 4) = 0.12 + 0.32 = 0.44$
$m(4, 3) = 0.48$
$m(6, 5) = 0.08$

Using Equation A-5, the expected value interval for $f:X \times Y$ can be calculated.

$E_*(f(x, y)) = 4(0.44) + 3*(0.48) + 5(0.08) = 3.6$, and
$E^*(f(x, y)) = 5(0.44) + 4(0.48) + 6(0.08) = 4.6$.

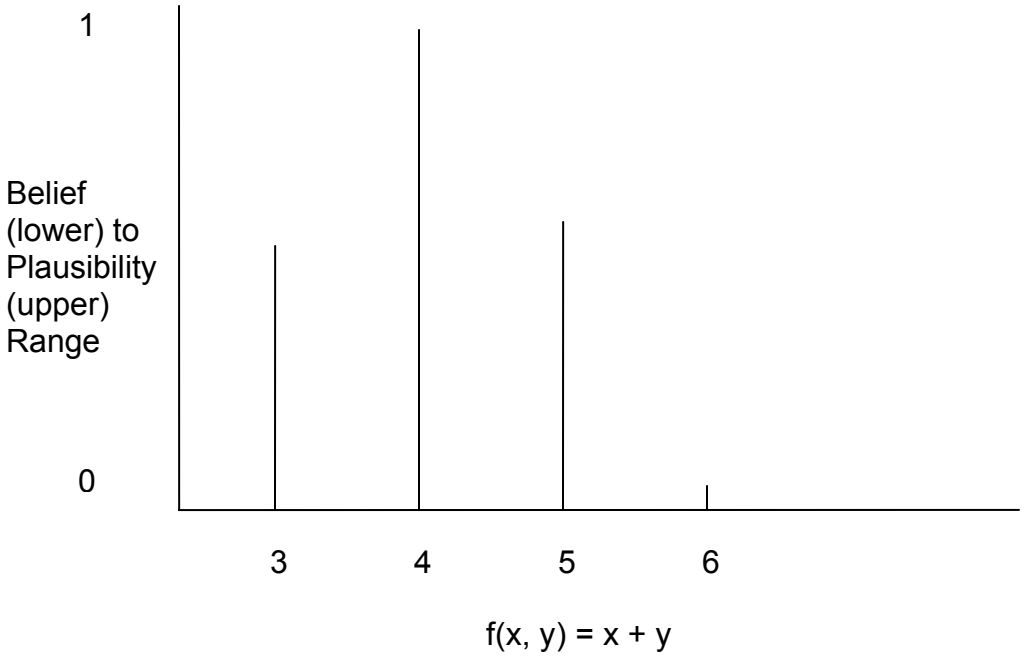

Figure A-10. Uncertainty Distribution for Example Problem

Belief and Plausibility can be calculated for subsets on the functional values. Consider the subset "values of the function $f:X \times Y$ greater than 3" for the example problem. The tuples of $X \times Y$ that form this subset are: $\langle x_1, y_1\rangle$, $\langle x_1, y_2\rangle$, $\langle x_2, y_1\rangle$, $\langle x_2, y_2\rangle$, and $\langle x_2, y_3\rangle$. Using Equation A-20, $C = \{\langle x_1, y_1\rangle, \langle x_1, y_2\rangle, \langle x_2, y_1\rangle, \langle x_2, y_2\rangle, \langle x_2, y_3\rangle\}$, and:

$Bel(A) = m(R_1) + m(R_3) + m(R_4) = 0.52$
$Pl(A) = m(R_1) + m(R_2) + m(R_3) + m(R_4) = 1.0$.

For the subset "all values of $f:X \times Y$" belief and plausibility are both 1.0 as expected.

A.3.4 Expectation Value for a Function of a Random Vector

The previous sections discussed the convolution process for generating the uncertainty distribution for the values of a function on a random vector, $f:X \times Y \rightarrow$ Reals, for different metrics for uncertainty.

The process for calculating the expected interval of a random variable was also summarized, and for a probability metric the expected value interval simplifies to a point value, the mean.

This section discusses some conditions for which the expected value for the values of the function can be directly calculated from the expected values for the constituent random variables.

For a probability measure with f(x, y) = $x + y$ it can be shown that the expected value for X + Y is the sum of the expected values for X and Y, even if X and Y are not independent. That is, E[X + Y] = E[X] + E[Y] in all cases.

For a probability measure with f(x, y) = x·y it can be shown that the expected value for X·Y is the product of the expected values for X and Y, if X and Y are independent. That is, E[X·Y] = E[X] ·E[Y] if X and Y are independent.

As proven in section A.3.4.1, using a belief measure for noninteractive X and Y,
$E_*[X + Y] = E_*[X] + E_*[Y]$, and
$E^*[X + Y] = E^*[X] + E^*[Y]$. Furthermore, if the domain of X and Y are the *non-negative* reals, then $E_*[X \cdot Y] = E_*[X] \cdot E_*[Y]$, and $E^*[X \cdot Y] = E^*[X] \cdot E^*[Y]$.

A.3.4.1 Expectation Value for Noninteractive X and Y

A.3.4.1.1 E(X+Y), X and Y Noninteractive, X and Y Real Numbers

Consider random variables X and Y whose values are real numbers, with X and Y noninteractive as defined in Section A.3.3. Under these conditions, it is asserted that:
$E_*(X + Y) = E_*(X) + E_*(Y)$ and $E^*(X + Y) = E^*(X) + E^*(Y)$.

The proof of this assertion follows.

Consider f(x, y) = x + y. Following the development in Section A.3.3:

$$\begin{aligned} E_*(X+Y) &= \sum_{all\ A_i x B_j \subseteq X\,x\,Y} \inf\{a+b \mid < a,b > \in A_i x B_j\} * m_X(A_i) * m_Y(B_j) \\ E^*(X+Y) &= \sum_{all\ A_i x B_j \subseteq X\,x\,Y} \sup\{a+b \mid < a,b > \in A_i x B_j\} * m_X(A_i) * m_Y(B_j) \end{aligned} \qquad \text{(Eqn. A-23)}$$

where A_i and B_j are the focal elements of X and Y, respectively; that is, A_i and B_j are elements of the power set of X and Y with non-zero degrees of evidence. A_i x B_j is a relation on X x Y.

"sup" is supremum, least upper bound, and "inf" is infimum, greatest lower bound.[35] $m_X(A_i)$ and $m_Y(B_j)$ are the projections of $m(A_i \times B_j)$ onto X and Y, respectively. Noninteraction means that $m(A_i \times B_j) = m_X(A_i) \cdot m_Y(B_j)$.

Let $a_{i\,inf}$, $b_{j\,inf}$, $a_{i\,sup}$, and $b_{j\,sup}$ be defined as follows:

$$\begin{aligned} a_{i\,\inf} &\equiv \inf\{a \mid a \in A_i\} \\ b_{j\,\inf} &\equiv \inf\{b \mid b \in B_j\} \\ a_{i\,\sup} &\equiv \sup\{a \mid a \in A_i\} \\ b_{j\,\sup} &\equiv \sup\{b \mid b \in B_j\} \end{aligned} \qquad \text{(Eqn. A-24)}$$

For any real numbers, positive or negative:

$$\begin{aligned} \inf\{a+b \mid < a,b > \in A_i x B_j\} &= a_{i\,\inf} + b_{j\,\inf} \\ \sup\{a+b \mid < a,b > \in A_i x B_j\} &= a_{i\,\sup} + b_{j\,\sup} \end{aligned} \qquad \text{(Eqn. A-25)}$$

and Equation A-23 can be written as:

$$\begin{aligned} E_*(X+Y) &= \sum_{all\ A_i x B_j \subseteq X\,x\,Y} (a_{i\,\inf} + b_{j\,\inf}) * m_X(A_i) * m_Y(B_j) \\ E^*(X+Y) &= \sum_{all\ A_i x B_j \subseteq X\,x\,Y} (a_{i\,\sup} + b_{j\,\sup}) * m_X(A_i) * m_Y(B_j) \end{aligned} \qquad \text{(Eqn. A-26)}$$

Equation A-26 can be written:

$$\begin{aligned} E_*(X+Y) &= \sum_{all\ A_i} \sum_{all\ B_j} (a_{i\,\inf} + b_{j\,\inf}) * m_X(A_i) * m_Y(B_j) \\ E^*(X+Y) &= \sum_{all\ A_i} \sum_{all\ B_j} (a_{i\,\sup} + b_{j\,\sup}) * m_X(A_i) * m_Y(B_j) \end{aligned} \qquad \text{(Eqn. A-27)}$$

Equation A-27 can be written:

$$\begin{aligned} E_*(X+Y) &= \sum_{all\ A_i} a_{i\,\inf} * m_X(A_i) \sum_{all\ B_j} m_Y(B_j) + \sum_{all\ B_j} b_{j\,\inf} * m_Y(B_j) \sum_{all\ A_j} m_X(A_i) \\ E^*(X+Y) &= \sum_{all\ A_i} a_{i\,\sup} * m_X(A_i) \sum_{all\ B_j} m_Y(B_j) + \sum_{all\ B_j} b_{j\,\sup} * m_Y(B_j) \sum_{all\ A_j} m_X(A_i) \end{aligned} \qquad \text{(Eqn. A-28)}$$

Since it is required that:

[35] Since this section is dealing with discrete sets, inf is equivalent to min and sup is equivalent to max. The conclusions in Section A.3.4.1 are valid for focal element that are intervals of real numbers as well as for focal elements that are sets of discrete numbers.

$$\sum_{all\ A_i} m_X(A_i) = 1$$
$$\sum_{all\ B_i} m_Y(B_j) = 1 \qquad \text{(Eqn. A-29)}$$

Equation A-28 simplifies to:

$$E_*(X+Y) = \sum_{all\ A_i} a_{i\,\inf} * m_X(A_i) + \sum_{all\ B_j} b_{j\,\inf} * m_Y(B_j)$$
$$E^*(X+Y) = \sum_{all\ A_i} a_{i\,\sup} * m_X(A_i) + \sum_{all\ B_j} b_{j\,\sup} * m_Y(B_j) \qquad \text{(Eqn. A-30)}$$

Equation A-30 shows that $E_*(X+Y) = E_*(X) + E_*(Y)$ and $E^*(X+Y) = E^*(X) + E^*(Y)$.

Therefore, if the domain of X and Y are the reals and X and Y are noninteractive, $E_*(X + Y) = E_*(X) + E_*(Y)$ and $E^*(X + Y) = E^*(X) + E^*(Y)$.

A.3.4.1.2 E(X·Y), X and Y Noninteractive, X and Y Non-negative Real Numbers

Consider random variables X and Y whose values are *non-negative* real numbers, with X and Y noninteractive as defined in Section A.2.3. Under these conditions, it is asserted that:

$E_*(X\cdot Y) = E_*(X)\cdot E_*(Y)$ and $E^*(X\cdot Y) = E^*(X)\cdot E^*(Y)$.

The proof of this assertion follows.

Consider $f(x, y) = x\cdot y$. Following Section A.3.3:

$$E_*(X*Y) = \sum_{all\ A_i x B_j \subseteq X\,x\,Y} \inf\{a*b \mid <a,b> \in A_i x B_j\} * m_X(A_i) * m_Y(B_j)$$
$$E^*(X*Y) = \sum_{all\ A_i x B_j \subseteq X\,x\,Y} \sup\{a*b \mid <a,b> \in A_i x B_j\} * m_X(A_i) * m_Y(B_j) \qquad \text{(Eqn. A-31)}$$

Let $a_{i\,inf}$, $b_{j\,inf}$, $a_{i\,sup}$, and $b_{j\,sup}$ be as defined in Equation A-24.

Since we are dealing with non-negative real numbers[36],

$$\inf\{a*b \mid <a,b> \in A_i x B_j\} = a_{i\,\inf} * b_{j\,\inf}$$
$$\sup\{a*b \mid <a,b> \in A_i x B_j\} = a_{i\,\sup} * b_{j\,\sup} \qquad \text{(Eqn. A-32)}$$

and Equation A-31 can be written as:

[36] Equation A-32 is not valid if we allow negative numbers in the domain for X or Y.

$$\begin{aligned} E_*(X*Y) &= \sum_{all\ A_i xB_j \subseteq X\,x\,Y} (a_{i\,\inf} * b_{j\,\inf}) * m_X(A_i) * m_Y(B_j) \\ E^*(X*Y) &= \sum_{all\ A_i xB_j \subseteq X\,x\,Y} (a_{i\,\sup} * b_{j\,\sup}) * m_X(A_i) * m_Y(B_j) \end{aligned} \quad \text{(Eqn. A-33)}$$

Equation A-33 can be written:

$$\begin{aligned} E_*(X*Y) &= \sum_{all\ A_i} \sum_{all\ B_j} a_{i\,\inf} * b_{j\,\inf} * m_X(A_i) * m_Y(B_j) \\ E^*(X*Y) &= \sum_{all\ A_i} \sum_{all\ B_j} a_{i\,\sup} * b_{j\,\sup} * m_X(A_i) * m_Y(B_j) \end{aligned} \quad \text{(Eqn. A-34)}$$

Equation A-34 can be re-written as:

$$\begin{aligned} E_*(X*Y) &= \sum_{all\ A_i} a_{i\,\inf} * m_X(A_i) \sum_{all\ B_j} b_{j\,\inf} * m_Y(B_j) \\ E^*(X*Y) &= \sum_{all\ A_i} a_{i\,\sup} * m_X(A_i) \sum_{all\ B_j} b_{j\,\sup} * m_Y(B_j) \end{aligned} \quad \text{(Eqn. A-35)}$$

Equation A-35 shows that $E_*(X \cdot Y) = E_*(X) \cdot E_*(Y)$ and $E^*(X \cdot Y) = E^*(X) \cdot E^*(Y)$.

Therefore, if the domain of X and Y are the *non-negative* reals and X and Y are noninteractive, $E_*(X \cdot Y) = E_*(X) \cdot E_*(Y)$ and $E^*(X \cdot Y) = E^*(X) \cdot E^*(Y)$.

A.3.4.1.2 Examples

A.3.4.1.2.1 Examples with X and Y Non-Negative

For the example in Section A.3.3, the expected value interval $[E_*(X + Y), E^*(X + Y)]$ was calculated to be [3.6, 4.6]. Since X and Y satisfy the requirements for the assertions, the expected value interval can be calculated as $[E_*(X) + E_*(Y), E^*(X) + E^*(Y)]$.

Specifically,

$E_*(X) = 1 \cdot 0.6 + 2 \cdot 0.4 = 1.4$
$E^*(X) = 1 \cdot 0.6 + 2 \cdot 0.4 = 1.4$
$E_*(Y) = \min(4, 3) \cdot 0.2 + \min(3, 2) \cdot 0.8 = 2.2$
$E^*(Y) = \max(4, 3) \cdot 0.2 + \max(3, 2) \cdot 0.8 = 3.2$
and $[E_*(X) + E_*(Y), E^*(X) + E^*(Y)] = [3.6, 4.6]$.

Similarly, for the example in section 4, the expected value for $(1 - P_E)$ is a probabilistic mean: $0.02 \cdot 0.3 + 0.03 \cdot 0.2 + 0.90 \cdot 0.1 + 0.05 \cdot 0 = 0.102$ and the expected value for C is a probabilistic mean: $0.03 \cdot 10^2 + 0.05 \cdot 10^3 + 0.80 \cdot 10^4 + 0.10 \cdot 10^5 + 0.02 \cdot 10^6 = 3.80 \times 10^4$. The expected value interval for f_A is:

$E_*(f_A) = \min(0, 10^{-5})\cdot 0.70 + \min(0, 10^{-5}, 10^{-4}, 10^{-3})\cdot 0.15 +$
$\min(0, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2})\cdot 0.15 = 0$
$E^*(f_A) = \max(0, 10^{-5})\cdot 0.70 + \max(0, 10^{-5}, 10^{-4}, 10^{-3})\cdot 0.15 +$
$\max(0, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2})\cdot 0.15 = 1.657 \times 10^{-3}$.
$[E_*(1 - P_E) * E_*(C) * E_*(f_A),\ E^*(1 - P_E) * E^*(C) * E^*(f_A)] = [0, 6.4]$.

By convolution, as summarized in Section A.3.3

$[E_*(f_A *(1 - P_E)*C), E^*(f_A *(1 - P_E)*C)] = [0, 6.4]$.

A.3.4.1.2.1 Example with X and Y having Negative Values

Consider X = {-2, 1, 4} and Y = {-6, 0, 5}. Let the focal elements of X be:
{-2, 1} with m = 0.6
{1, 4} with m = 0.4
Let the focal elements of Y be:
{-6, 0} with m = 0.8
{-6, 0, 5} with m = 0.2.
$E_*(X) = 0.6(-2) + 0.4(1) = -0.8$
$E^*(X) = 0.6(1) + 0.4(4) = 2.2$
$E_*(Y) = 0.8(-6) + 0.2(-6) = -6.0$
$E^*(Y) = 0.8(0) + 0.2(5) = 1.0$.

Consider X + Y.
$E_*(X) + E_*(Y) = -6.8$
$E^*(X) + E^*(Y) = 3.2$
Using the *BeliefConvolution* code, $E_*(X+Y) = -6.8$ and
$E^*(X+Y) = 3.2$, so $E_*(X+Y) = E_*(X) + E_*(Y)$ and $E^*(X+Y) = E^*(X) + E^*(Y)$.

Consider X·Y.
$E_*(X) \cdot E_*(Y) = 4.8$
$E_*(X)\cdot E^*(Y) = -0.8$
$E^*(X) \cdot E_*(Y) = -13.2$
$E^*(X) \cdot E^*(Y) = 2.2$.

Using the *BeliefConvolution* code,
$E_*(X\cdot Y) = -13.68$ and $E^*(X\cdot Y) = 8.8$, so
$E_*(X\cdot Y) \neq E_*^*(X) \cdot E_*^*(Y)$ and $E^*(X\cdot Y) \neq E_*^*(X) \cdot E_*^*(Y)$ where E_*^* denotes either E_* or E^* to account for the product of two negative numbers being positive.

A.3.5 Convolution with Fuzzy Sets

The discussion of convolution in Sections A.3.1 through A.3.4 assumed crisp sets. Convolution can also be performed with fuzzy sets. The following discussion is from the paper by Yager, for convolution using the belief/plausibility measure given evidence on fuzzy sets defined by degrees of membership on the reals. [Yager 1986]

Let OP denote any arithmetic operation on real numbers (addition, subtraction, multiplication, division, exponentiation) . Let Z = X OP Y and let S be a subset of Z.

Let A_i and B_j denote the i^{th} and j^{th} focal elements of X and Y, respectively, and assume the focal elements are noninteractive:

$$m(S) = \sum_{\substack{i,j \\ A_i \ OP \ B_j = S}} m(A_i) \bullet m(B_j) \qquad \text{(Eqn. A-36)}$$

If A_i and B_j are fuzzy sets, OP is defined as

$$Z = A_i \ OP \ B_j \qquad \text{(Eqn. A-37)}$$

where for any real number z

$$Z(z) = \max_{\substack{all\ x,y \\ such\ that \\ x\ OP\ y = z}} \{\min[\mu_{A_i}(x), \mu_{B_j}(y)]\} \qquad \text{(Eqn. A-38)}$$

For the special case of A_i and B_j crisp:

$$Z = \{x \ OP \ y \mid for\ x \in A_i \ and \ y \in B_j\} \qquad \text{(Eqn. A-39)}$$

The paper by Yager also addresses operations that are not arithmetic. [Yager 1986]

A.3.6 Linguistic Convolution

This section summarizes a technique for the evaluation of purely linguistic information using a belief measure.

In this approach, X and Y are modeled to a level of detail consistent with the fidelity of the information available. For example if X is the number of deaths from a terrorist attack, we may chose to bin X into subsets such as “Minor,” “Moderate,” “Major” and “Catastrophic.” Also, we may not have a precise definition of each set; for example, “Major” may be defined as “between about 500 and about 5000 deaths.” Given fuzzy sets defined with degrees of membership over the reals, the techniques of Section A.2.7 can be used to evaluate belief/plausibility for any fuzzy set of concern, and the techniques of Section A.3.5 can be used to evaluate belief/plausibility for arithmetic operations on fuzzy sets defined over the reals.

However, as discussed in Section 1, for our application we have variables whose fuzzy sets are purely linguistic, such as the variable Y being “Damage to National Security” modeled with, for example, the fuzzy sets “Not Much,” “Of Concern,” and “Yes.” These fuzzy sets are purely linguistic and do not have degrees of membership defined over the reals.

In the linguistic model, X and Y do not have to be expressed using numbers, and the fuzzy sets for X and Yare purely linguistic, not defined by degrees of membership over numerical values of X and Y. That is, the fidelity of the model is at the fuzzy set level. A random vector Z = X x Y is also described by purely linguistic fuzzy sets, and a function for Z is defined by an approximate reasoning rule base.

Evidence is over the fuzzy sets in X and Y. A focal element for X is of the form $[A_i, m(A_i)]$ where A_i is an element of the fuzzy power set of X,[37] $m(A_i)$ is the evidence assigned to A_i, and i ranges over all the focal elements for X. Similarly, a focal element for Y is of the form $[B_j, m(B_j)]$ where B_j is subset of the set of all fuzzy sets of Y, $m(B_j)$ is the evidence assigned to B_j, and j is over all the focal elements for Y. Assuming noninteraction, $m(A_i \times B_j) = m_X(A_i) \cdot m_Y(B_j)$.

From the rule base, a given fuzzy set for Z, call it C_m, is of the form $C_m = \bigcup_{i,j \text{ per rules}} \{< F_i, G_j >\}$ where F_i and G_j are fuzzy sets of X and Y, respectively, and <> denotes a tuple.

Both the evidence and the rules are at the fuzzy set level; all the A_i, B_j, and C_m are comprised of fuzzy sets from the fuzzy sets of X and Y. Since we are reasoning at the fuzzy set level without definitions for degrees of membership for the fuzzy sets, the mathematics of Section A.2.7 cannot be used in the evaluation of rules. The standard Equation A-4 for belief/plausibility is used; that is, the fuzzy sets are fuzzy in the assignment of evidence, but are treated as crisp in the evaluation of the rule base.

A.4 Belief for Focal Elements as Intervals of Real Numbers

The previous sections of this appendix focused on evaluation of uncertainty for a variable that has a discrete number of values. A similar approach can be used for a variable whose value is any real number over an interval, if degrees of evidence are assigned to a finite number of intervals within the domain of the variable. Oberkampf and Helton discuss this approach.

If the intervals to which degrees of evidence are assigned are point values (degenerate intervals) then both belief and plausibility reduce to probability.

A.5 Summary of Techniques

The material previously provided discussed the modeling of uncertainty using a set of values and an uncertainty distribution over that set. Two types of uncertainty were discussed: ambiguity and vagueness. A general measure for ambiguity was presented: belief/plausibility. For ambiguity involving strife with no nonspecificity, belief/plausibility both become probability. For a consonant body of evidence, belief/plausibility become necessity/possibility, respectively.

[37] For example if the fuzzy sets for X are “bad” and “good”, the fuzzy power set for X is { {null}, {“bad”}, {“good”}, {“bad”, “good”}}, and one possible A_i is {“bad”, “good”}.

Convolution of random variables using the following measures was discussed: probability, possibility, and belief. Linguistic convolution using an approximate reasoning rule base was also discussed.

The concept of fuzzy sets for vagueness was discussed. Extension ("fuzzification") of probability, possibility, and belief to also include vagueness were summarized.

A.6 Java Tools: *BeliefConvolution* and LinguisticBelief

Two Java codes were written to effect convolution of numeric or linguistic variables using the belief/plausibility measure. The codes were written in Java 1.5 using the netbeans 4.1 Integrated Development Environment (IDE). Features of Java 1.5, such as generic classes with parameterized types, were used in the coding so the codes will not compile or execute in versions of Java earlier than 1.5.

BeliefConvolution performs convolution of random variables with evidence assigned to intervals of real numbers. LinguisticBelief performs convolution of linguistic variables with evidence assigned to fuzzy sets.

Both codes assume that the variables are non-interacting.

A.6.1 *BeliefConvolution*

BeliefConvolution implements the mathematics of belief discussed earlier for algebraic combinations of random variables with evidence assigned to intervals of real numbers. Two aspects of *BeliefConvolution* are documented here: aggregation of evidence and the ability to evaluate belief for fuzzy sets given evidence on crisp sets.

A.6.1.1 Aggregation of Evidence

The aggregation approach used in the code is similar to the aggregation technique for discrete probability distributions discussed by Kaplan. [Kaplan] The code allows for aggregation of degrees of evidence for any variable, using linear or logarithmic binning. Aggregation is a process by which the degrees of evidence are reduced by mapping the variable into bins and assigning a point estimate, the midpoint of the bin, to any value of the variable in that bin. The need for aggregation is to reduce the numbers of degrees of evidence from the convolution of a large number of variables. For example, suppose that each variable has 3 degrees of evidence. A convolution of 10 such variables results in a variable with 3^{10} or about 60,000 degrees of evidence and this is a manageable number. But a convolution of 20 such variables results in a variable with 3^{20} or about 3.4×10^9 degrees of evidence and this is a not manageable number.

Unfortunately, convolution using belief/plausibility requires combining the degrees of evidence even if the variables are noninteractive. For two variables X and Y with A a subset (event or interval) of X and B a subset (event or interval) of Y, Bel(A x B) <= min[Bel(A), Bel(B)] where Bel is belief, so we cannot accurately calculate the belief for subsets of X x Y using belief for subsets of X combined with belief for subsets of Y. However, if X and Y are noninteractive we

can combine degrees of evidence, m, as $m(A \times B) = m_X(A) \cdot m_Y(B)$ where m_X and m_Y are marginal degrees of evidence, and using the $m(A \times B)$ for all A and B that are focal elements the $Bel(A \times B)$ can be calculated from the focal elements of $X \times Y$.[38]

Consider a variable X that is the set of all reals in [min, max]. The binning process involves defining a number of bins that partition X and considering every number within a bin to map to a single number, the value of the midpoint of the bin.

For linear binning into n bins the width of a bin is (max-min)/n and the n bins are taken as:

> [min, bin 1 max], (bin 1 max, bin 2 max], (bin 2 max, bin 3 max], …
> (bin n-1 max, max].

For log binning over the range 0 to 10^{28} the following bins are used:

> $[0, 5 \times 10^{-6}], (5 \times 10^{-6}, 1 \times 10^{-5}], (1 \times 10^{-5}, 5 \times 10^{-5}], \ldots$
> $(5 \times 10^{27}, 1 \times 10^{28}]$.

Each degree of evidence over X is a value assigned to an interval [low, high] in X. The mapping of the degrees of evidence to the binned values of X is as follows. Low maps to the point value of the bin i where low is within (bin i low, bin i high]. High maps to the point value of the bin j where high is within (bin j low, bin j high]. So the mapped degree of evidence applies to the interval [bin i point value, bin j point value]. The mapping is not one-to-one since more than one unique [low, high] has the same [bin i point value, bin j point value]. In fact, the degrees of evidence reduce, or aggregate, precisely because the mapping is not one-to-one. Each aggregated interval is assigned a degree of evidence equal to the sum of the degrees of evidence for each original interval that mapped into that aggregated interval. Figure A-11 graphically illustrates the aggregation process.

[38] For probability, if A and B are independent $P(A \times B) = P(A) \cdot P(B)$ where P is probability. For possibility, even if A and B are dependent $Pos(A + B) = \max[Pos(A), Pos(B)]$ where Pos is possibility and the '+' in $Pos(A + B)$ is the cartesian co-product; if A and B are noninterative in the possibilistic sense, it is also true that $Pos(A \times B) = \min[Pos(A), Pos(B)]$. The references provide more detailed information, specifically [Dubois and Prade, Possibility Theory].

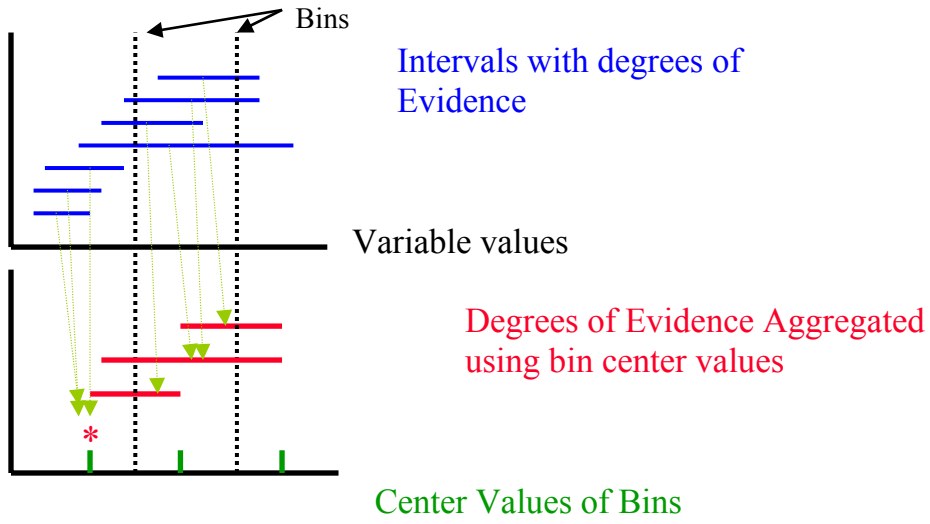

Figure A-11. Aggregation Process

For example let the variable XX be [4, 100] with the following degrees of evidence:

0.2 for [4, 11]
0.4 for [4, 45]
0.3 for [8, 20]
0.1 for [27, 51]

Using linear binning with 4 bins the bins are:

[4, 28] with midpoint 16
(28, 52] with midpoint 40
(52, 76] with midpoint 64
(76, 100] with midpoint 88

The mapping of the degrees of evidence is as follows:

[4, 11] → [16, 16] with evidence 0.2
[4, 45] → [16, 40] with evidence 0.4
[8, 20] → [16, 16] with evidence 0.3
[27, 51] → [16, 40] with evidence 0.1

The final aggregated degrees of evidence are:

[16, 16] with evidence 0.5
[16, 40] with evidence 0.5

Aggregation is useful when the number of degrees of evidence for a variable in the chain of algebraic operations becomes so large that further convolution would increase the number of degrees of evidence to an unmanageable number. The number of bins must not be set too large or aggregation causes too much loss of fidelity. For our security application, the variable min/max values are such that log aggregation may be better than linear aggregation. For example, a consequence variable may range from 0 to 10^7 deaths so consequences within about a factor of 10 are essentially identical for this scale.

The *BeliefConvolution* Java code allows for either linear or log aggregation at any step in the convolution.

Figure A-12 shows linear aggregation of evidence in *BeliefConvolution* for the variable XX just discussed.

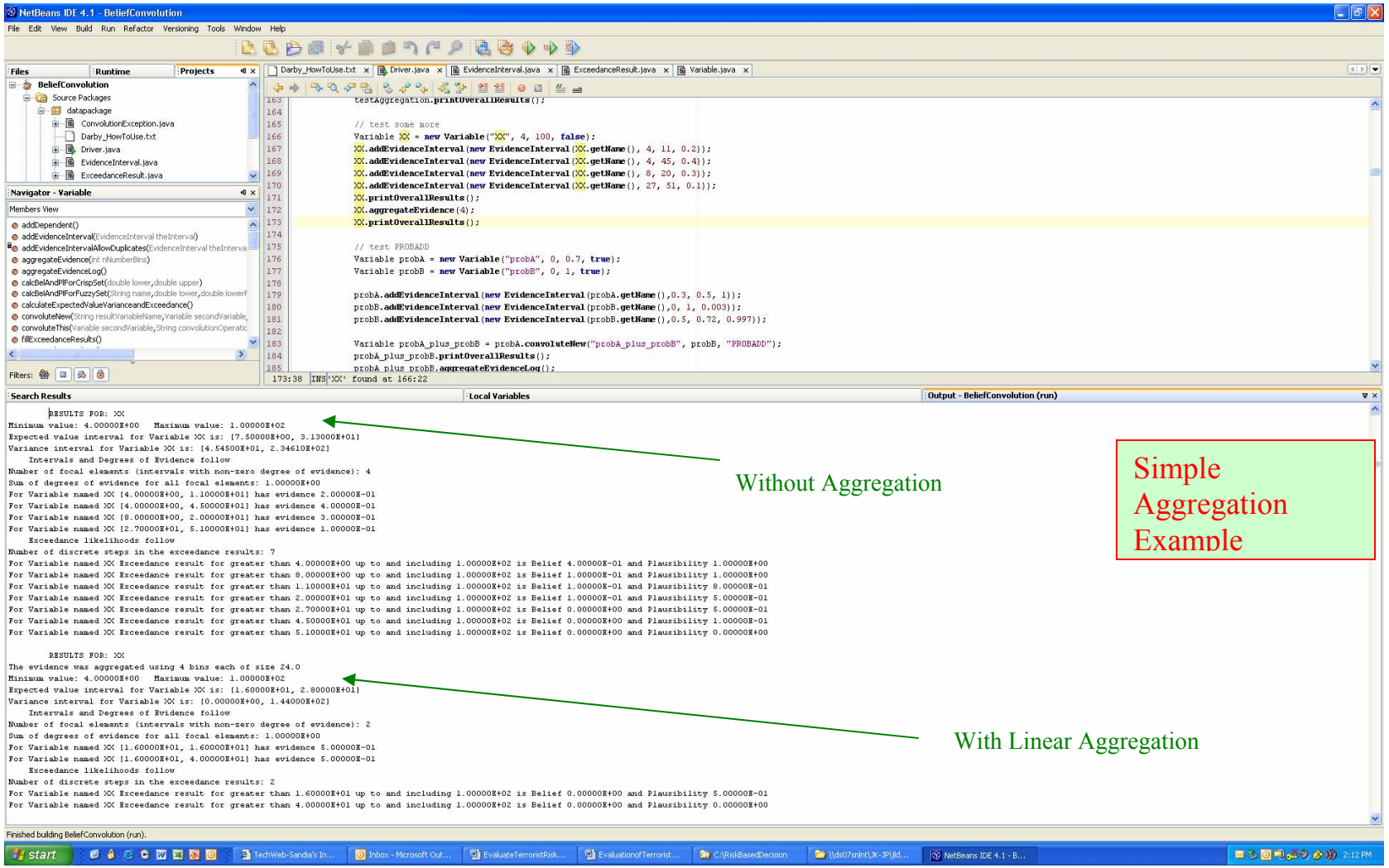

Figure A-12. Aggregation of Evidence in *BeliefConvolution* Java Code

A.6.1.2 Belief/Plausibility for a Fuzzy Set

The ability to calculate belief/plausibility for fuzzy sets allows results to be summarized as linguistic variables (e.g., Minor, Moderate, Major, Catastrophic) and each linguistic variable is a fuzzy set over the appropriate numeric variable. For cases with crisp focal elements, *BeliefConvolution* calculates the belief/plausibility for a fuzzy set using Equation A-15.

BeliefConvolution requires that the shape of the fuzzy sets be trapezoids (or triangles, or rectangles which are crisp sets). The tuple <lower, lowerCrisp, upperCrisp, and upper> specifies a fuzzy set. For example, the fuzzy set <4, 15, 27, 48> is shown in Figure A-13.

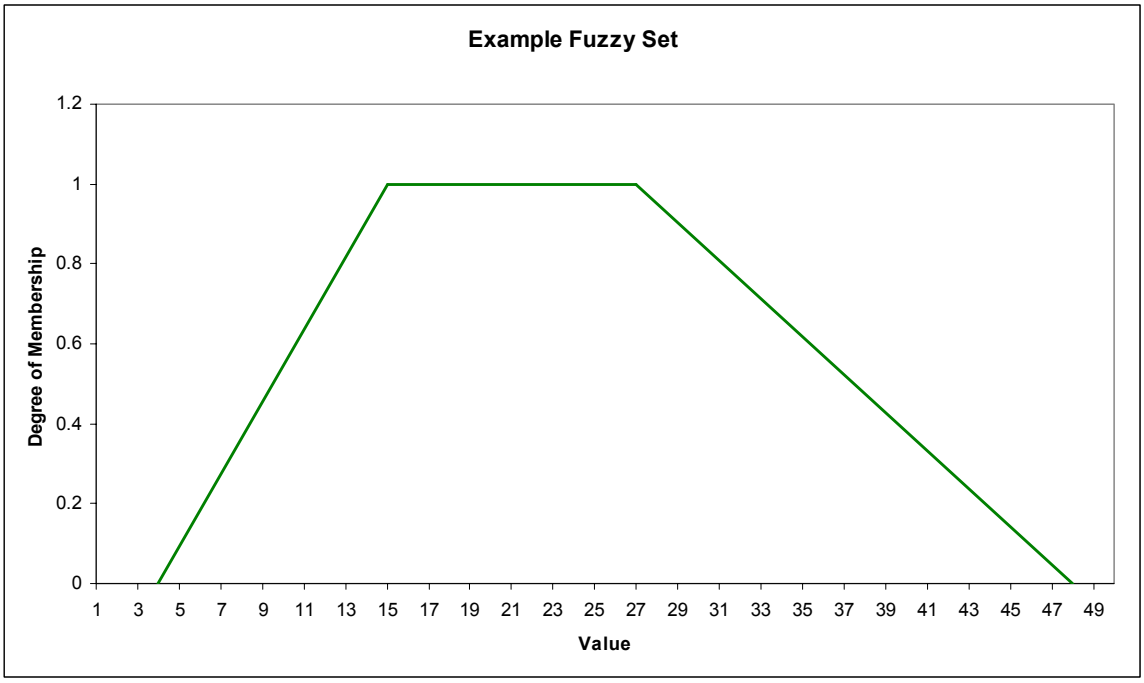

Figure A-13. Example Trapezoidal Fuzzy Set

Degenerate cases are <15, 15, 27, 27> (the crisp set shown in Figure A-14), and the triangular shaped fuzzy sets <4, 15, 15, 27> and <4, 4, 4, 27> shown in Figures A-15 and A-16.

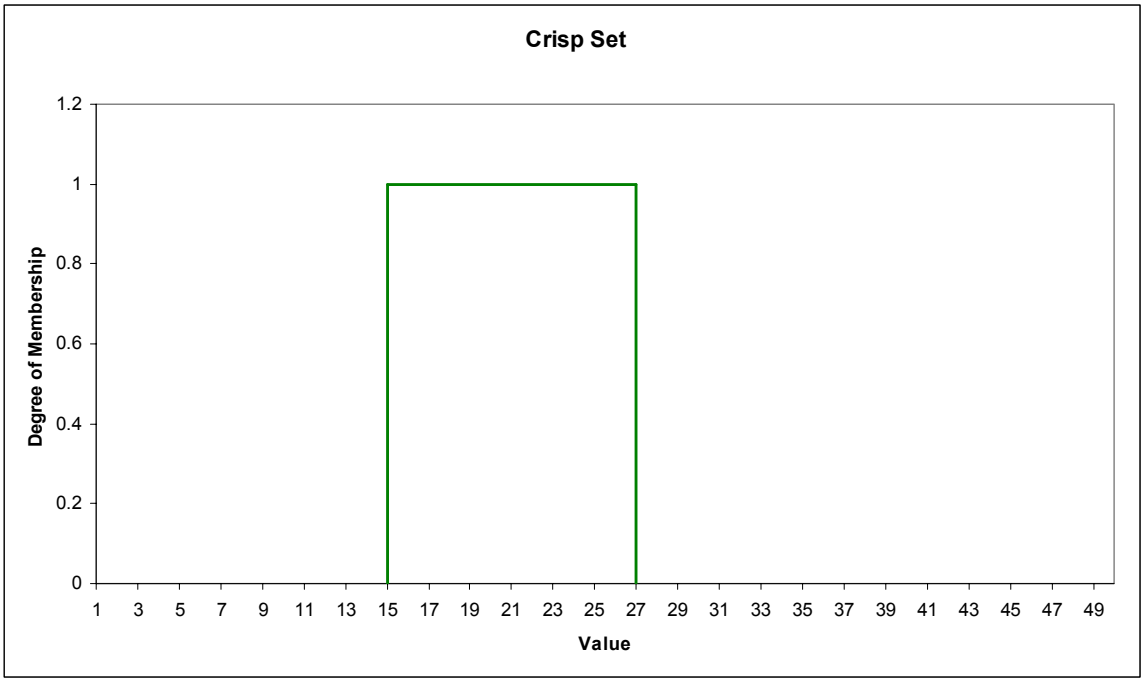

Figure A-14. Example Crisp Set

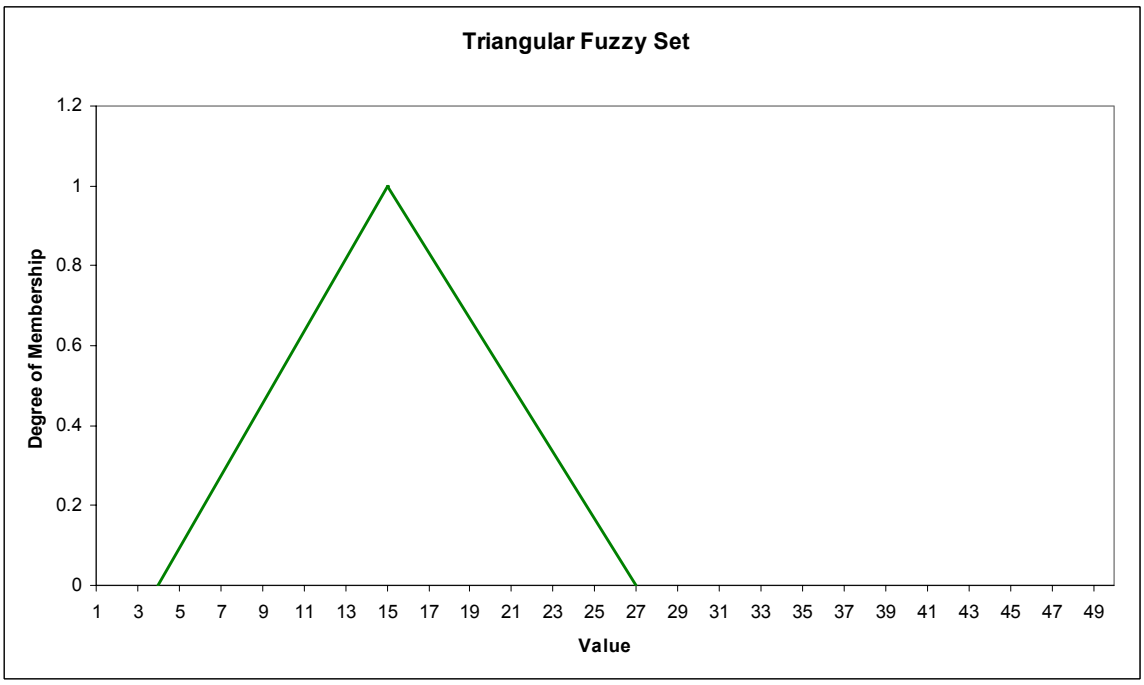

Figure A-15. Example Triangular Fuzzy Set

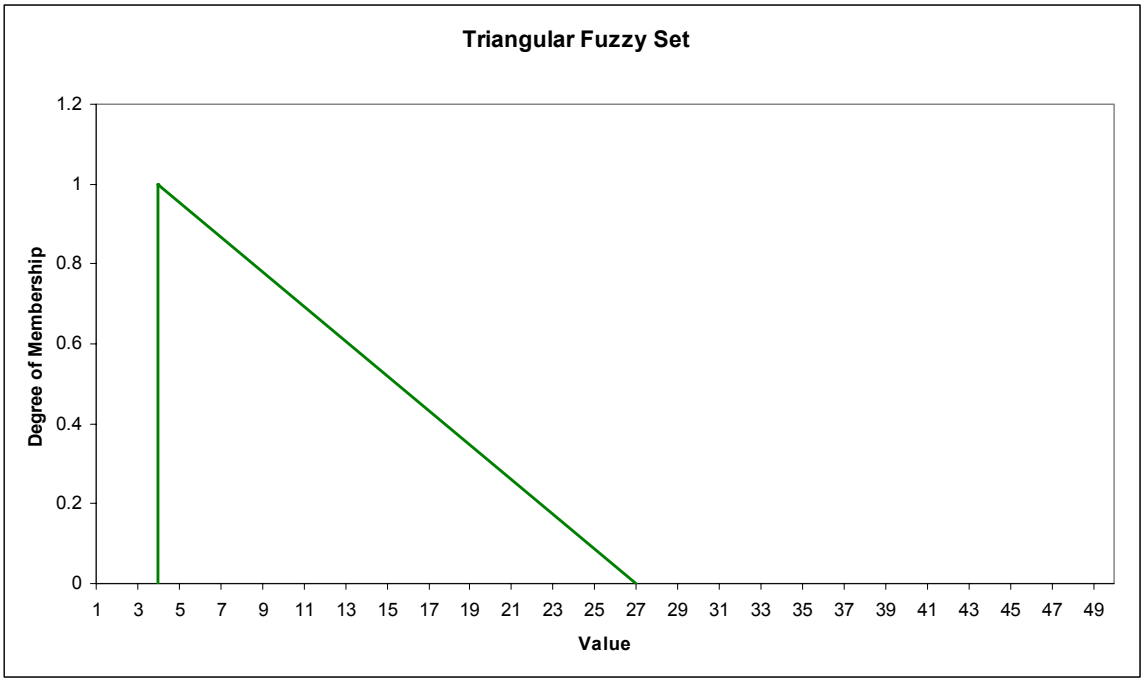

Figure A-16. Example Triangular Fuzzy Set

The fuzzy set is divided into three portions: lower fuzzy portion, crisp portion, and upper crisp portion. For the fuzzy sets shown above the portions are:

For <4, 15, 27, 48>
Lower fuzzy portion is [4, 15)
Crisp portion is [15, 27]
Upper fuzzy portion is (27, 48]

For <15, 15, 27, 27>
Lower fuzzy portion is not present
Crisp portion is [15, 27]
Upper fuzzy portion is not present

For <4, 15, 15, 27>
Lower fuzzy portion is [4, 15)
Crisp portion is not present
Upper fuzzy portion [15, 27]

For <4, 4, 4, 27>
Lower fuzzy portion is not present
Crisp portion is not present
Upper fuzzy portion is [4, 27]

Not that each fuzzy portion has an area ½ of the area if that fuzzy portion were crisp; see Figure A-17 for example.

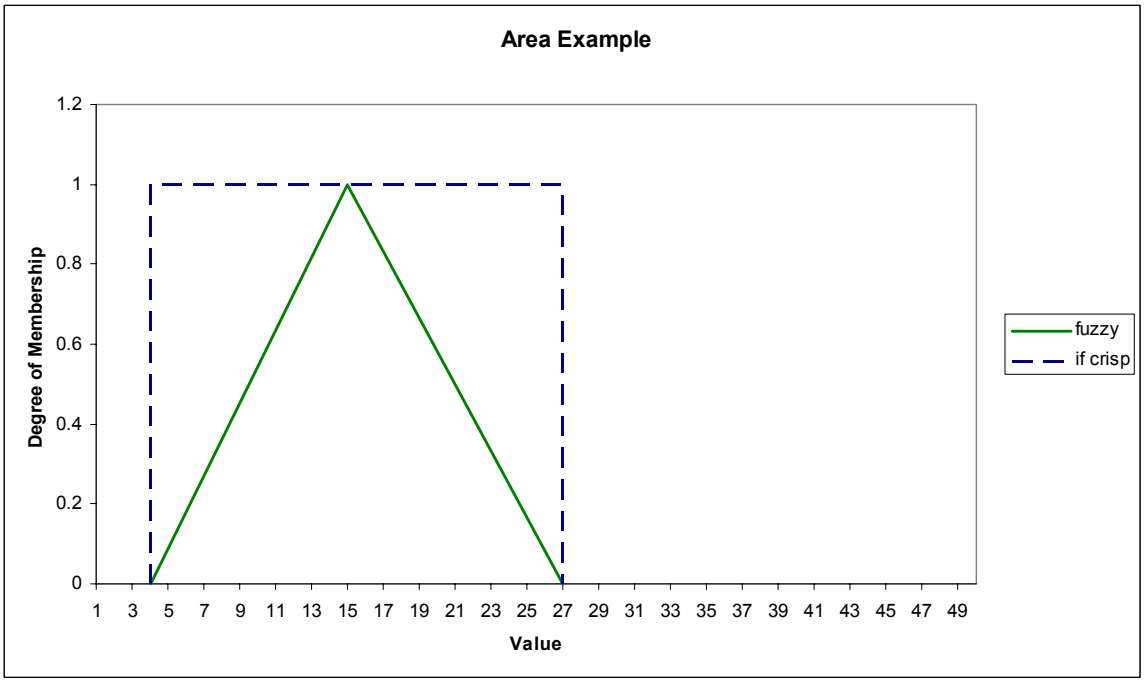

Figure A-17. Area Example

As an example of the calculation of belief for a fuzzy set, consider the fuzzy set <4, 15, 27, 48> shown in Figure A-18.

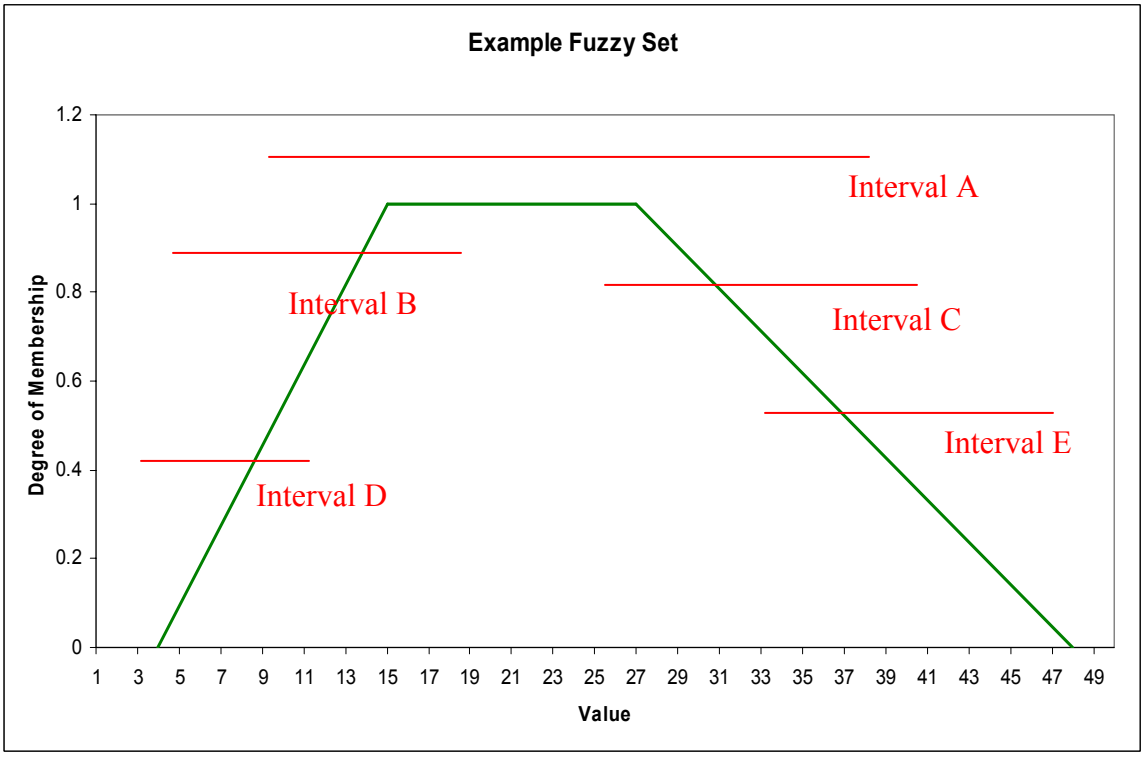

Figure A-18. Crisp Evidence with a Fuzzy Set

Assume the degrees of evidence over the crisp intervals indicated in Figure A-18 are as follows:

Interval A [7, 36] has degree of evidence 0.15
Interval B [4, 18] has degree of evidence 0.23
Interval C [25, 40] has degree of evidence 0.07
Interval D [3, 11] has degree of evidence 0.49
Interval E [33, 48] has degree of evidence 0.06

Using *BeliefConvolution*, the belief and plausibility for this fuzzy set can be calculated:

```
// report fuzzy set belief example input
Variable reportFuzzySet = new Variable("ReportFuzzySet", 1, 50, false);
reportFuzzySet.addEvidenceInterval(new EvidenceInterval(reportFuzzySet.getName(), 7, 36, 0.15));
reportFuzzySet.addEvidenceInterval(new EvidenceInterval(reportFuzzySet.getName(), 4, 18, 0.23));
reportFuzzySet.addEvidenceInterval(new EvidenceInterval(reportFuzzySet.getName(), 25, 40, 0.07));
reportFuzzySet.addEvidenceInterval(new EvidenceInterval(reportFuzzySet.getName(), 3, 11, 0.49));
reportFuzzySet.addEvidenceInterval(new EvidenceInterval(reportFuzzySet.getName(), 33, 48, 0.06));

reportFuzzySet.printOverallResults();
reportFuzzySet.printBelPlForFuzzySet("Example Fuzzy Set", 4, 15, 27, 48);
reportFuzzySet.printBelPlForCrispSet(9.5, 37.5);
```

RESULTS FOR: ReportFuzzySet
Minimum value: 1.00000E+00 Maximum value: 5.00000E+01
Expected value interval for Variable ReportFuzzySet is: [7.17000E+00, 2.06100E+01]
  Intervals and Degrees of Evidence follow
Number of focal elements (intervals with non-zero degree of evidence): 5
Sum of degrees of evidence for all focal elements: 1.00000E+00
For Variable named ReportFuzzySet [3.00000E+00, 1.10000E+01] has evidence 4.90000E-01
For Variable named ReportFuzzySet [4.00000E+00, 1.80000E+01] has evidence 2.30000E-01
For Variable named ReportFuzzySet [7.00000E+00, 3.60000E+01] has evidence 1.50000E-01
For Variable named ReportFuzzySet [2.50000E+01, 4.00000E+01] has evidence 7.00000E-02
For Variable named ReportFuzzySet [3.30000E+01, 4.80000E+01] has evidence 6.00000E-02
  Exceedance likelihoods follow
Number of discrete steps in the exceedance results: 10
For Variable named ReportFuzzySet Exceedance result for greater than 3.00000E+00 up to and including 5.00000E+01 is Belief 5.10000E-01 and Plausibility 1.00000E+00
For Variable named ReportFuzzySet Exceedance result for greater than 4.00000E+00 up to and including 5.00000E+01 is Belief 2.80000E-01 and Plausibility 1.00000E+00
For Variable named ReportFuzzySet Exceedance result for greater than 7.00000E+00 up to and including 5.00000E+01 is Belief 1.30000E-01 and Plausibility 1.00000E+00
For Variable named ReportFuzzySet Exceedance result for greater than 1.10000E+01 up to and including 5.00000E+01 is Belief 1.30000E-01 and Plausibility 5.10000E-01
For Variable named ReportFuzzySet Exceedance result for greater than 1.80000E+01 up to and including 5.00000E+01 is Belief 1.30000E-01 and Plausibility 2.80000E-01
For Variable named ReportFuzzySet Exceedance result for greater than 2.50000E+01 up to and including 5.00000E+01 is Belief 6.00000E-02 and Plausibility 2.80000E-01
For Variable named ReportFuzzySet Exceedance result for greater than 3.30000E+01 up to and including 5.00000E+01 is Belief 0.00000E+00 and Plausibility 2.80000E-01
For Variable named ReportFuzzySet Exceedance result for greater than 3.60000E+01 up to and including 5.00000E+01 is Belief 0.00000E+00 and Plausibility 1.30000E-01
For Variable named ReportFuzzySet Exceedance result for greater than 4.00000E+01 up to and including 5.00000E+01 is Belief 0.00000E+00 and Plausibility 6.00000E-02
For Variable named ReportFuzzySet Exceedance result for greater than 4.80000E+01 up to and including 5.00000E+01 is Belief 0.00000E+00 and Plausibility 0.00000E+00

For Variable named ReportFuzzySet the fuzzy set Example Fuzzy Set with {lower, lowerCrisp, upperCrisp, upper} of {4.00000E+00, 1.50000E+01, 2.70000E+01, 4.80000E+01} has Belief 6.75758E-02 and Plausibility 8.04675E-01

For the example problem, the fuzzy set has Belief/Plausibility of 0.068/0.80.

The fuzzy set can be approximated by the crisp set <9.5, 9.5, 37.5, 37.5> and for this crisp set the code calculates:

For Variable named ReportFuzzySet the crisp set [9.50000E+00, 3.75000E+01] has Belief 0.00000E+00 and Plausibility 1.00000E+00

This crisp set has Belief/Plausibility of 0/1.0.

A.6.2 LinguisticBelief

LinguisticBelief implements the mathematics of belief discussed earlier for rule based combinations of linguistic variables with evidence assigned to fuzzy sets. For a variable formed from a rule base, no more than three input variables are allowed.

Figure A-19 shows a screen capture of the LinguisticBelief code in the netbeans IDE.

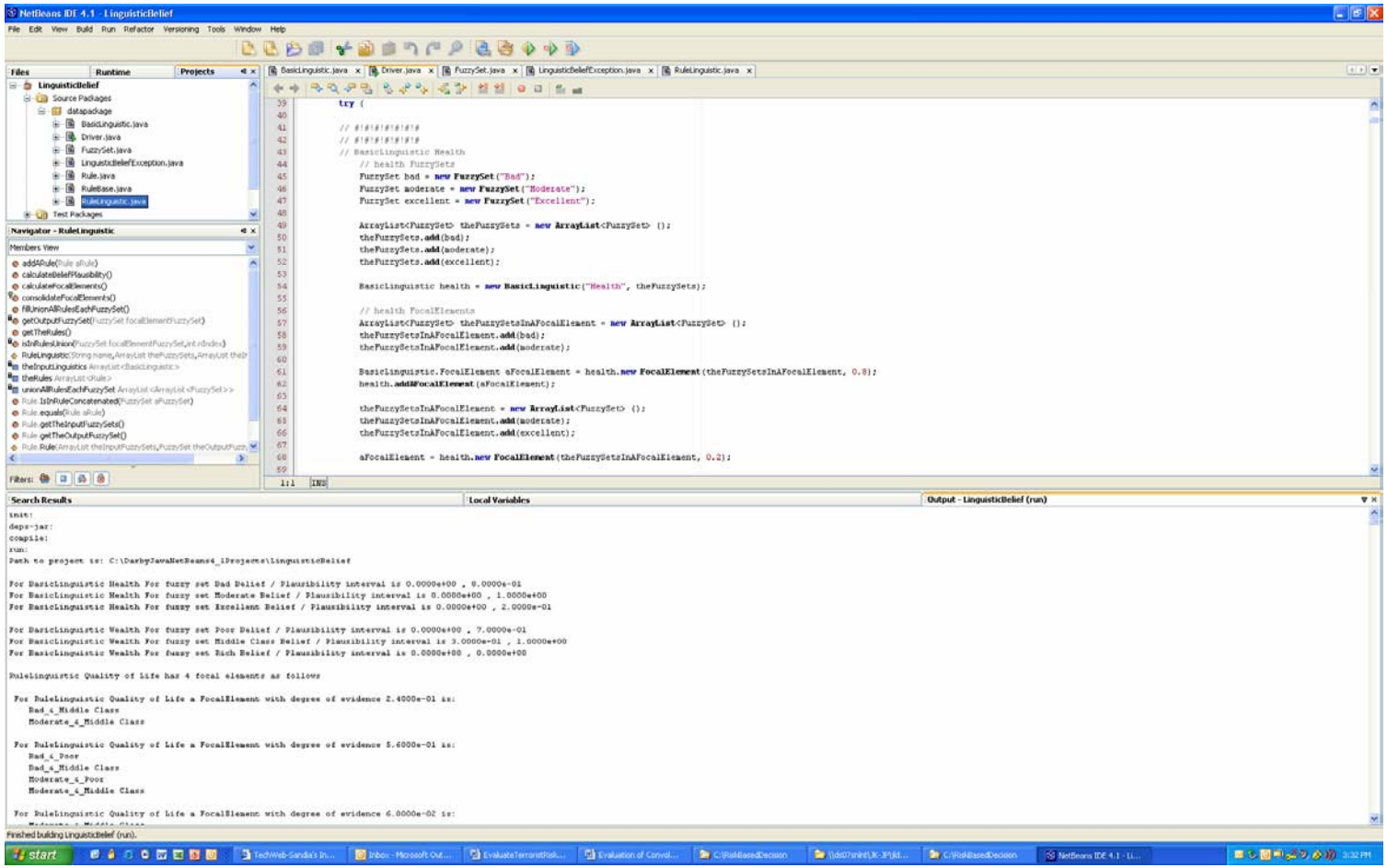

Figure A-19. LinguisticBelief in the netbeans IDE

Based on the mapping of input fuzzy sets to output fuzzy sets per the rule base, there is an automatic "aggregation" of focal elements performed in the LinguisticBelief code. (As discussed in Section A.1, for convolution of numeric variables using *BeliefConvolution*, aggregation requires user-specified binning.)

This aggregation is best discussed by an example. First the example is presented, then the process of aggregation is discussed using this example.

Consider a simple purely linguistic example where we wish to reason on Quality of Life based on Health and Wealth. For "Health" we will use the fuzzy sets "Bad," "Moderate," Excellent." Uncertainty is reflected by the assignment of degrees of evidence to appropriate combinations of these fuzzy sets. For example, based on the information available for a specific individual named "John" we may assign the following evidence for the "Health" of "John":

0.8 to {"Bad," "Moderate"}, and
0.2 to {"Moderate," "Excellent"}

Assume we model "Wealth" with the fuzzy sets "Poor," "Middle Class," and "Rich." Based on the evidence available we assign evidence for the "Wealth" of "John" as:

0.3 to {"Middle Class"}, and
0.7 to {"Poor," "Middle Class"}

We wish to reason on the linguistic "Quality Of Life" based on combining "Health" and "Wealth" using the rule base for "Quality Of Life" is provided in Table A-6.

Table A-6. Rule Base for Quality of Life

Health ***Quality of Life*** **Wealth**	**Bad**	**Moderate**	**Excellent**
Poor	***Not So Good***	***Not So Good***	***Good***
Middle Class	***Not So Good***	***Not So Good***	***Good***
Rich	***Not So Good***	***Good***	***Good***

The rule base implies that "Quality Of Life" "Not So Good" is formed from:

{<"Bad", "Poor">, <"Bad", "Middle Class">, <"Bad", "Rich">, <"Moderate", "Poor">, <"Moderate," "Middle Class">},

and that "Quality Of Life" "Good" is formed from:

{<"Moderate", "Rich"">, <"Excellent", "Poor">, <"Excellent", "Middle Class">, <"Excellent," "Rich">}.

Using the evidence provided for "Health" and "Wealth" for "John" and the rule base for "Quality Of Life", and assuming that "Health" and "Wealth" are non-interactive, we obtain the following focal elements (here focal elements are evidence for combinations of fuzzy sets) for "John":

{<"Bad," "Middle Class">, <"Moderate," "Middle Class">} with evidence 0.24

{<"Bad," "Poor">, <"Moderate," "Poor">, <"Bad," "Middle Class">, <"Moderate," "Middle Class">} with evidence 0.56

{<"Moderate," "Middle Class">, <"Excellent," "Middle Class">} with evidence 0.06

{<"Moderate," "Poor">, <"Excellent," "Poor">, <"Moderate," "Middle Class">, <"Excellent," "Middle Class">} with evidence 0.14.

The combinatorics involved in evaluating a rule base are straightforward but tedious; the author wrote a Java computer code called LinguisticBelief to perform convolution of linguistic variables with fuzzy sets using belief/plausibility.

For the single simple rule base for "Quality Of Life" for "John" a manual evaluation is instructive.

The belief/plausibility for Not So Good is calculated using Equation A-4. The following two focal elements of Health x Wealth are a subset of Not So Good:

{<Bad, Middle Class>, <Moderate, Middle Class>}, and
{< Bad, Poor>, <Moderate, Poor >, < Bad, Middle Class >, <Moderate, Middle Class >},

so Bel(Not So Good) = 0.24 + 0.56 = 0.80.

All the focal elements of Health x Wealth have non-null intersection with Not So Good, so Pl(Not So Good) = 1.0.

Similarly for Good, Bel(Good) = 0 and Pl(Good) = 0.06 + 0.14 = 0.20.

In summary, using the mathematics of belief/plausibility, we obtain the following results for "Quality Of Life" for "John":

"Not So Good" has a belief/plausibility interval of 0.8/1.0
"Good" has a belief/plausibility interval of 0/0.2.

Figure A-20 shows these results.

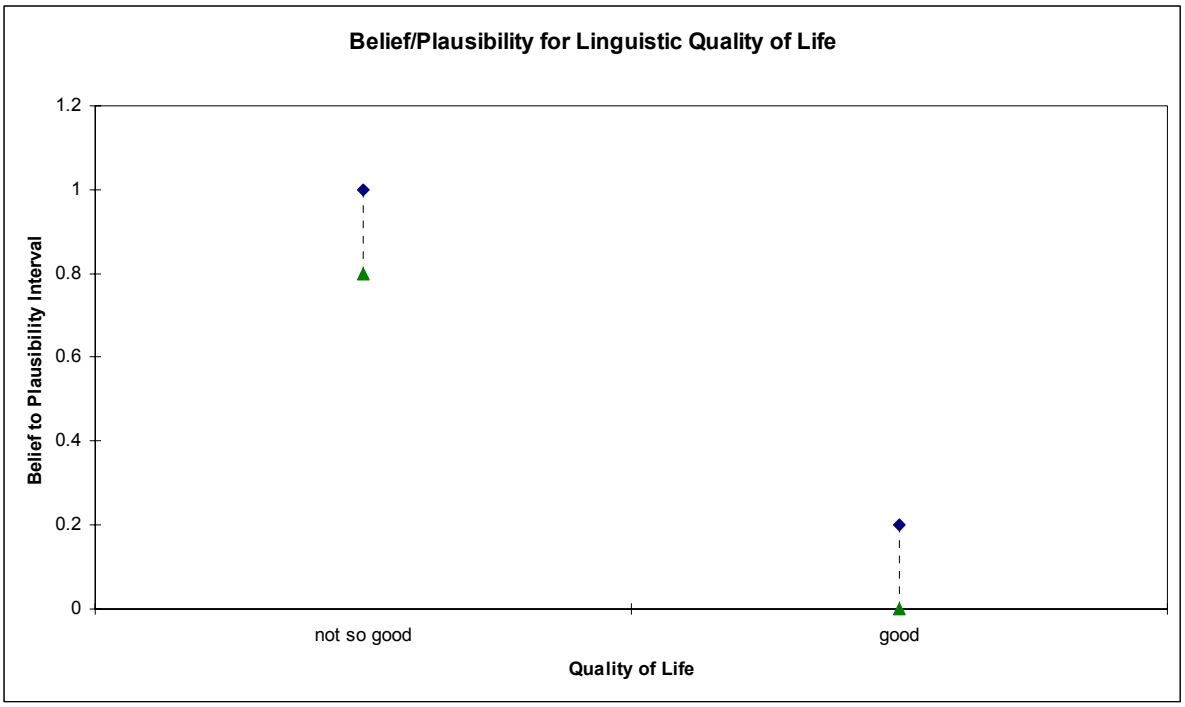

Figure A-20. Happiness for John

The rule base can be extensive. For example, we can combine "Quality Of Life" with "Outlook On Life" to evaluate "Happiness." Let the fuzzy sets for "Outlook On Life" be "Pessimist" and "Optimist" and let the fuzzy sets for "Happiness" be "Depressed," "Accepting," and "Very Happy." Form "Happiness" using the following rule base:

Table A-7. Rule Base for Happiness

Outlook On Life ***Happiness*** **Quality Of Life**	**Pessimist**	**Optimist**
Not So Good	***Depressed***	***Accepting***
Good	***Accepting***	***Very Happy***

Assume the following evidence for "Outlook On Life" for "John":

{"Pessimist"} with evidence 0.02, and
{"Pessimist," "Optimist"} with evidence 0.98.

Using the LinguisticBelief code, the results for for "Happiness" for "John" are:

Very Happy has a belief/plausibility interval of 0 / 0.196
Accepting has a belief/plausibility interval of 0 / 0.984
Depressed has a belief/plausibility interval is: 0.016 / 1.0.

Figure A-21 illustrates the results graphically.

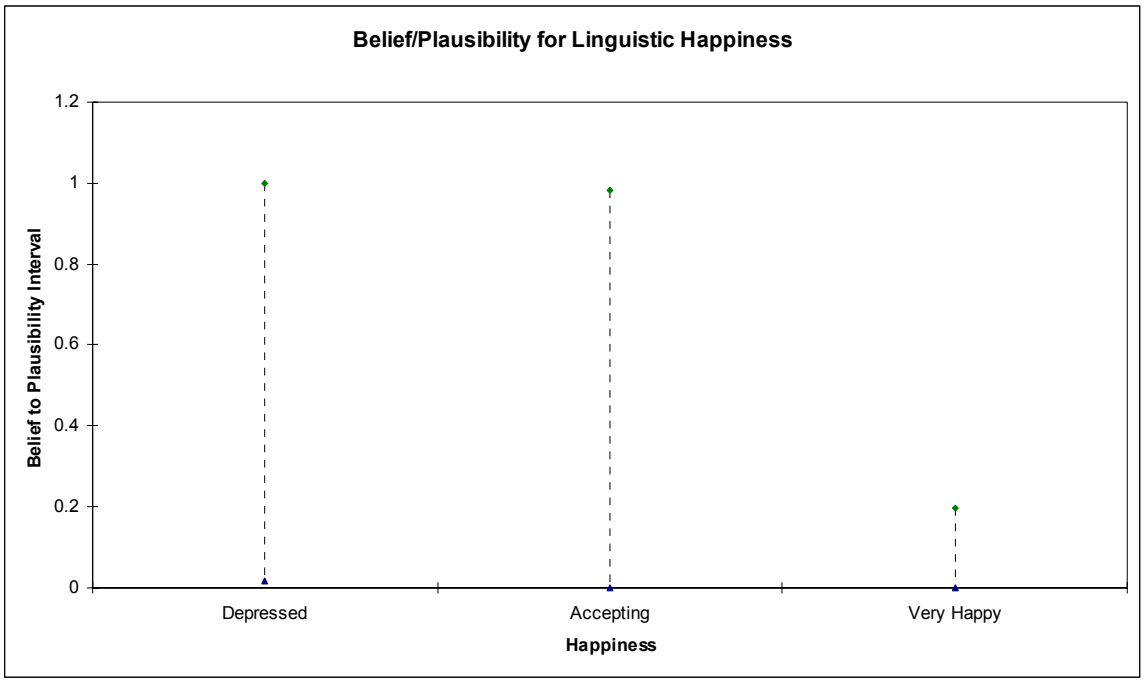

Figure A-21. Happiness for John

The utility of the linguistic technique using belief/plausibility is that it focuses the argument on the rules for how we evaluate the result of a combination of many very different variables each with considerable epistemic uncertainty, rather than focusing on a precise numerical estimate of the result given little information.

To show how aggregation is performed for the linguistic calculation, selected output from the LinguisticBelief Code for the evaluation of the prior example follows.

Quality of Life Focal Elements

For RuleLinguistic Quality of Life a FocalElement with degree of evidence 2.4000e-01 is:
Bad_&_Middle Class
Moderate_&_Middle Class

For RuleLinguistic Quality of Life a FocalElement with degree of evidence 5.6000e-01 is:
Bad_&_Poor
Bad_&_Middle Class
Moderate_&_Poor
Moderate_&_Middle Class

For RuleLinguistic Quality of Life a FocalElement with degree of evidence 6.0000e-02 is:
Moderate_&_Middle Class
Excellent_&_Middle Class

For RuleLinguistic Quality of Life a FocalElement with degree of evidence 1.4000e-01 is:
Moderate_&_Poor
Moderate_&_Middle Class
Excellent_&_Poor
Excellent_&_Middle Class

Quality of Life Rule Base

For RuleLinguistic Quality of Life the fuzzy sets from union of rules with output FuzzySet Not So Good are:
Bad_&_Poor
Moderate_&_Poor
Bad_&_Middle Class
Moderate_&_Middle Class
Bad_&_Rich

For RuleLinguistic Quality of Life the fuzzy sets from union of rules with output FuzzySet Good are:
Excellent_&_Poor
Excellent_&_Middle Class
Moderate_&_Rich
Excellent_&_Rich

Quality of Life Belief / Plausibility Intervals

For BasicLinguistic Quality of Life For fuzzy set Not So Good Belief / Plausibility interval is: 8.00000e-01 / 1.00000e+00
For BasicLinguistic Quality of Life For fuzzy set Good Belief / Plausibility interval is: 0.00000e+00 / 2.00000e-01

Outlook on Life Belief / Plausibility Intervals

For BasicLinguistic Outlook On Life For fuzzy set Pessimist Belief / Plausibility interval is 2.0000e-02 , 1.0000e+00
For BasicLinguistic Outlook On Life For fuzzy set Optimist Belief / Plausibility interval is 0.0000e+00 , 9.8000e-01

To evaluate "Happiness" using the rule base, its focal elements need to be expressed in terms of the fuzzy sets of its input variables. This is accomplished by aggregation.

For example, before aggregation the focal elements for "Happiness" are:

(1) For RuleLinguistic Happiness a FocalElement with degree of evidence: 4.8000e-03 is
Bad_&_Middle Class_&_Pessimist
Moderate_&_Middle Class_&_Pessimist

(2) For RuleLinguistic Happiness a FocalElement with degree of evidence: 2.3520e-01 is
Bad_&_Middle Class_&_Pessimist
Bad_&_Middle Class_&_Optimist
Moderate_&_Middle Class_&_Pessimist
Moderate_&_Middle Class_&_Optimist

(3) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.1200e-02 is:
Bad_&_Poor_&_Pessimist
Bad_&_Middle Class_&_Pessimist
Moderate_&_Poor_&_Pessimist
Moderate_&_Middle Class_&_Pessimist

(4) For RuleLinguistic Happiness a FocalElement with degree of evidence 5.4880e-01 is:
Bad_&_Poor_&_Pessimist
Bad_&_Poor_&_Optimist
Bad_&_Middle Class_&_Pessimist
Bad_&_Middle Class_&_Optimist
Moderate_&_Poor_&_Pessimist
Moderate_&_Poor_&_Optimist
Moderate_&_Middle Class_&_Pessimist

Moderate_&_Middle Class_&_Optimist

(5) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.2000e-03 is:
Moderate_&_Middle Class_&_Pessimist
Excellent_&_Middle Class_&_Pessimist

(6) For RuleLinguistic Happiness a FocalElement with degree of evidence 5.8800e-02 is:
Moderate_&_Middle Class_&_Pessimist
Moderate_&_Middle Class_&_Optimist
Excellent_&_Middle Class_&_Pessimist
Excellent_&_Middle Class_&_Optimist

(7) For RuleLinguistic Happiness a FocalElement with degree of evidence 2.8000e-03 is:
Moderate_&_Poor_&_Pessimist
Moderate_&_Middle Class_&_Pessimist
Excellent_&_Poor_&_Pessimist
Excellent_&_Middle Class_&_Pessimist

(8) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.3720e-01 is:
Moderate_&_Poor_&_Pessimist
Moderate_&_Poor_&_Optimist
Moderate_&_Middle Class_&_Pessimist
Moderate_&_Middle Class_&_Optimist
Excellent_&_Poor_&_Pessimist
Excellent_&_Poor_&_Optimist
Excellent_&_Middle Class_&_Pessimist
Excellent_&_Middle Class_&_Optimist

The rule base for "Happiness" is:

For RuleLinguistic Happiness the fuzzy sets from union of rules with output FuzzySet Very Happy are:
Good_&_Optimist

For RuleLinguistic Happiness the fuzzy sets from union of rules with output FuzzySet Accepting are:
Not So Good_&_Optimist
Good_&_Pessimist

For RuleLinguistic Happiness the fuzzy sets from union of rules with output FuzzySet Depressed are:
Not So Good_&_Pessimist

The focal elements for "Happiness" contain "Quality Of Life" in terms of the fuzzy sets of its input variables "Health" and "Wealth", but the rule base for "Happiness" is expressed in terms of the fuzzy sets for "Quality Of Life." The focal elements for "Happiness" must be aggregated to be expressed in terms of the fuzzy sets of "Quality Of Life" instead of being expressed in terms of the fuzzy sets for the input variables for "Quality Of Life."

Aggregation is performed in two steps. First, for any given focal element, its constituent input fuzzy sets are mapped to the appropriate output fuzzy set per the rule base. For "Happiness" this results in the following focal elements:

(1) For RuleLinguistic Happiness a FocalElement with degree of evidence 4.8000e-03 is:
Not So Good_&_Pessimist

(2) For RuleLinguistic Happiness a FocalElement with degree of evidence 2.3520e-01 is:
Not So Good_&_Pessimist
Not So Good_&_Optimist

(3) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.1200e-02 is:
Not So Good_&_Pessimist

(4) For RuleLinguistic Happiness a FocalElement with degree of evidence: 5.4880e-01 is
Not So Good_&_Pessimist
Not So Good_&_Optimist

(5) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.2000e-03 is:
Not So Good_&_Pessimist
Good_&_Pessimist

(6) For RuleLinguistic Happiness a FocalElement with degree of evidence 5.8800e-02 is:
Not So Good_&_Pessimist
Not So Good_&_Optimist
Good_&_Pessimist
Good_&_Optimist

(7) For RuleLinguistic Happiness a FocalElement with degree of evidence 2.8000e-03 is:
Not So Good_&_Pessimist
Good_&_Pessimist

(8) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.3720e-01 is:
Not So Good_&_Pessimist
Not So Good_&_Optimist
Good_&_Pessimist
Good_&_Optimist

Second, focal elements resulting from step one containing identical fuzzy sets are combined by adding the degrees of evidence, resulting in the following focal elements for "Happiness".

(A) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.6000e-02 is:
Not So Good_&_Pessimist

(B) For RuleLinguistic Happiness a FocalElement with degree of evidence 7.8400e-01 is:
Not So Good_&_Pessimist
Not So Good_&_Optimist

(C) For RuleLinguistic Happiness a FocalElement with degree of evidence 4.0000e-03 is:
Not So Good_&_Pessimist
Good_&_Pessimist

(D) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.9600e-01 is:
Not So Good_&_Pessimist
Not So Good_&_Optimist
Good_&_Pessimist
Good_&_Optimist

This two-step aggregation operation reduced the number of focal elements for "Happiness" from eight to four.

References

Amenaza Technologies, SecurITree software, http://www.amenaza.com/

ANSI T1A1.2 Working Group on Network Survivability Performance, "A Technical Report on Network Survivability", Report No. 24A, Alliance for Telecommunications Industry Solutions, 1997.

ASME, 2004. Risk Analysis and Management for Critical Asset Protection: General Guidance. Draft. American Society of Mechanical Engineers, Washington, DC. July 30, 2004.

ATIS 1997a, Alliance for Telecommunications Industry Solutions Committee T1, "Technical Report on Enhanced Analysis of FCC-Reportable Service Outage Data," T1A1.2/97-001, Alliance for Telecommunications Industry Solutions, Red Bank, NJ, 1997.

ATIS 1997b, Alliance for Telecommunications Industry Solutions Committee T1, "An Outage Index for the Wireline Industry Segment," T1A1.2/97-001, Alliance for Telecommunications Industry Solutions, Red Bank, NJ, 1997.

Bennett, H.A. *The EASI Approach to Physical Security Evaluation*, SAND76-0500, Sandia National Laboratories, Albuquerque NM, 1977.

Biringer, Betty E., Richard D. Brown, Michael Ford, and William K. Paulus, *Survey of Security System Design and Analysis Tools*, SAND2000-0717, Sandia National Laboratories, Albuquerque NM, May 1999.

Campbell, P.L., and J.E. Stamp, *A Classification Scheme for Risk Assessment Methods*, Sandia National Laboratories Report SAND2004-4233, Albuquerque NM, August 2004.

Campbell, Philip L. and Bryan J. Smith, *Automating Quickstart ViaThe Sandia Quickstart Baseline Tool (SQBT)*, Sandia National Laboratories Albuquerque NM, January 2006.

Campbell, Philip L., "Proposed Sandia Customizations to the Sandia Quickstart Baseline Tool (SQBT) for SCADA Customers", Sandia National Laboratories Albuquerque NM, February 21, 2006.

COBIT Quickstart ©2003 IT Governance Institute (ITGI). All rights reserved.

Cramond, W.R., et al., *Probabilistic Risk Assessment Course Documentation*, SAND85-1495, NUREG/CR-4350, 7 volumes. Prepared by Sandia National Laboratories for the U.S. Nuclear Regulatory Commission, Washington, DC, 1985.

Darby, John, Evaluation of Risk from Acts of Terrorism: The Adversary/Defender Model using Belief and Fuzzy Sets, SANDOC 2006-5777, Sandia National Laboratories, Albuquerque NM, 2006.

Depoy, J., et al., "Risk Assessment for Physical and Cyber Attacks on Critical Infrastructures," IEEE MILCOM 2005, contributed by Sandia National Laboratories, October 2005.

Devooght, J., and Smidts, C., "Probabilistic Reactor Dynamics – I: The Theory of Continuous Event Trees," *Nuclear Science and Engineering* 111:229-240, 1992a.

Devooght, J., and Smidts, C., "Probabilistic Reactor Dynamics – III: A Framework for Time-Dependent Interaction Between Operator and Reactor During a Transient Involving Human Error," *Nuclear Science and Engineering* 112:100-113, 1992b.

Dougherty, Edward R., Probability and Statistics for the Engineering, Computing, and Physical Sciences, Prentice Hall, Englewood Cliffs, NJ, 1990.

Dubois, Didier and Henri Prade, *Possibility Theory, An Approach to Computerized Processing of Uncertainty*, Plenum Press English Translation, 1988.

Duggan, D., *Generic Threat Profiles*, SAND Report 2005-5411, Sandia National Laboratories, Albuquerque NM, July 2005

Eisenhawer, Steve, Terry Bott and D.V. Rao, 2003. Assessing the Risk of Nuclear Terrorism Using Logic Evolved Decision Analysis. American Nuclear Society Annual Meeting, San Diego, CA. June 1-4, 2003. LA-UR-03-3467.

Engi, D., "GAIA: Global Approaches to Infrastructure Assurance, a Work in Progress," SAND2000-2543, Sandia National Laboratories, Albuquerque NM, October 2000.

Engi, D., and C. P. Harlan, "Brief Adversary Threat Loss Estimator (BATLE) User's Guide," NUREG/CR-1432, SAND80-0952, Sandia National Laboratories, May 1981.

Engi, Dennis, and Jessica Glicken-Turnley, *The Vital Issues Process : Strategic Planning For A Changing World*, SAND95-0845, Sandia National Laboratories, Albuquerque, New Mexico, 1995.

Engi, Dennis, *The Vital Issues Process: Managing Critical Infrastructures In The Global Arena*, SAND97-1451, Sandia National Laboratories, Albuquerque, New Mexico, 1997.

"EPANET 2.0, Water Supply and Water Resource (WSWRD) Risk Management Research NRMRL OSD US EPA," www.epa.gov/ORD/NRMRL/wswrd/epanet.html

Ferson, Scott, Roger B. Nelson, Janos Hajagos, Daniel J. Berleant, Jianzhong Zhang, W. Troy Tucker, Lev R. Ginzburg, and William L. Overkampf, *Dependence in Probabilistic Modeling, Dempster-Shafer Theory, and Probability Bounds Analysis*, Sandia National Laboratories, Albuquerque NM, SAND2004-3072, October, 2004.

Garcia, M. L., The Design and Evaluation of Physical Protection Systems, Butterworth-Heinemann, 2001.

Gordon, K. A. and G. D. Wyss, *Comparison of Two Methods to Quantify Cyber and Physical Security Effectiveness*, SAND2005-7177, Sandia National Laboratories, Albuquerque, NM, November 2005.

Greenberg, H.R., and Cramer, J.J., editors, *Risk Assessment and Risk Management for the Chemical Process Industry*, Van Nostrand Reinhold, New York, 1991.

Information Design Assurance Red Team (IDARTTM) at Sandia National Laboratories. http://www.sandia.gov/idart.

IT Governance Institute® (2004). *Control Practices*. United States of America: ITGI.

Jae, M., and Apostolakis, G.E., "The Use of Influence Diagrams for Evaluating Severe Accident Management Strategies," *Nuclear Technology* 99:142-157, August 1992.

Jansma, Roxanna M., Sharon K. Fletcher, Martin D. Murphy, Judy J. Lim, Gregory D. Wyss, *Risk-Based Assessment of the Surety of Information Systems*, SAND96-2027, Sandia National Laboratories, Albuquerque, NM, July 1996.

Joint Publication 1-02 "Department of Defense Dictionary of Military and Associated Terms," August 31, 2005, http://www.dtic.mil/doctrine/jel/doddict.

Kaplan, Stanley and B. John Garrick, "On the Quantitative Definition of Risk", *Risk Analysis,* Vol. 1 No. 1, 1981 pp 11-27.

Kaplan, Stanley, "On the Method of Discrete Probability Distributions in Risk and Reliability Calculations-Application to Seismic Risk Assessment", *Risk Analysis*, Vol. 1, No. 3, 1981 pp189-196.

Klir. G., and B. Yuan, *Fuzzy Sets and Fuzzy Logic, Theory and Applications*, Prentice Hall PTR, Upper Saddle River NJ, 1995.

R. P. Lippmann and K. W. Ingols, "An Annotated Review of Past Papers on Attack Graphs", Technical Report ESC-TR-2005-054, MIT Lincoln Laboratory, Lexington, MA, 2005.

McCormick, N.J., Reliability and Risk Analysis: Methods and Nuclear Power Applications, Academic Press, New York, 1981.

Meyer, M, and J. Booker, *Eliciting and Analyzing Expert Judgment: A Practical Guide*, ISBN 0124932304, Academic Press, San Diego, 1991.

NRIC, Network Reliability and Interoperability Council Steering Committee, *NRIC Cyber Security Best Practices*. http://www.nric.org, 2006

Oberkampf, W. L., and Helton, J.C., "Evidence Theory for Engineering Applications", Chapter 10 of *Engineering Design Reliability Handbook*, Nikolaides, E., Ghiocel, D.M., and Singhal, S., eds., CRC Press, 2005.

PHPSurveyor, http://www.phpsurveyor.org/index.php

RAM-W: Risk Assessment Methodology for Water Utilities, Second Edition. Awwa Research Foundation. Denver, CO. 2001

Rees, Daniel C. and K. I. Rubin, 2003. Managing and Protecting Infrastructure Assets. Proceedings of IMECE 2003: 2003 ASME International Mechanical Engineering Congress and RD&D Expo. Washington, D.C., November 15-21, 2003.

Roberts, N.H., W.E. Vesely, D.F. Haasl, and F.F. Goldberg, *Fault Tree Handbook*, NUREG-0492, U.S. Nuclear Regulatory Commission, Washington, DC, 1981.

Saaty, Thomas L., Decision Making For Leaders: The Analytic Hierarchy Process for Decisions In A Complex World, ISBN 0962031704, Expert Choice, Inc., 1988.

Saaty, Thomas L., Multicriteria Decision Making : The Analytic Hierarchy Process; Planning, Priority Setting, Resource Allocation, ISBN 0962031712, RWS Publications, Pittsburgh, PA, 1990.

Smitds, C., and Devooght, J., "Probabilistic Dynamics: A Numerical Comparison Between a Continuous Event Tree and DYLAM-Type Event Tree," from the Proceedings of PSAM II, G.E. Apostolakis and J.S. Wu of the University of California at Los Angeles, editors, San Diego, California, March 20-25, 1994.

SNAC, Systems and Network Attack Center, National Security Agency (NSA). "The 60 Minute Network Security Guide (First Steps Towards a Secure Network Environment)." Updated July 12, 2002, http://www.nsa.gov/snac/support/sixty_minutes.pdf.

Stamp, J. and M. Berg, *Fundamental Security Practices for Control and Automation Systems in Electric Power*, SAND Report, SAND2005-7784J, Sandia National Laboratories, Albuquerque NM, 2005.

Stamp, Jason and Dominique Kilman, *Framework for SCADA Security Policies*, Sandia National Laboratories Report SAND2005-1002C, Albuquerque, New Mexico 2005.

Tidwell, V., J. A. Cooper, C. J. Silva, and S. Jurado, 2004. Threat Assessment of Water Supply Systems Using Markov Latent Effects Modeling. Sandia National Laboratories Report SAND2004-0818C, Albuquerque, New Mexico. 2004.

"VSAT: Vulnerability Self Assessment Tool", http://www.vsatusers.net/

Wyss, G.D., and Durán, F.A., *OBEST: The Object-Based Event Scenario Tree Methodology*, SAND2001-0828, Sandia National Laboratories, Albuquerque, NM, March 2001.

Wyss, G.D., Craft, R.L., and Funkhouser, D.L., *The Use of Object-Oriented Analysis Methods in Surety Analysis*, SAND99-1242, Sandia National Laboratories, Albuquerque, NM, May 1999.

Wyss, G.D., Felicia A. Durán And Vincent J. Dandini, "An Object-Oriented Approach to Risk and Reliability Analysis: Methodology and Aviation Safety Applications," *SIMULATION*, 80(1):33-43 (January 2004)

Yager, R. "Arithmetic and other operations on Dempster Shafer structures," *International Journal of Man-Machine Studies*, 25: pp 357-366, 1986.

Young, Mary Louise, Robert A. Jung, Bryan J. Smith, Richard L. Craft, Frank Joseph Jr. Schelling, Lee Shyr, Miriam Minton, Michael J. Berg, "Optimal allocation of terrorist countermeasures using risk-based systems analyses" by Sandia National Laboratories, Albuquerque, NM, SAND2004-4675, September 2004, OUO (Exemption 2)

Distribution

2		LDRD Office	
1	MS	0782	William G. Rhodes, III, 06418
6		0782	James Phelan, 06418
1		1368	Jennifer Depoy, 05615
1		0762	S. Jordan, 06407
1		0762	M. Bradley Parks, 06410
1		0672	Robert L. Hutchinson, 05616
1		0672	Peter Sholander, 05616
1		1361	John Matter, 06957
1		1361	G. Bruce Varnado, 06957
1		0757	Carla Ulibarri, 06442
1		0757	Gregory D. Wyss, 06442
1		0757	John Darby, 06442
1		0757	Andrew Walter, 06442
2	MS	9018	Central Technical Files, 08944
2		0899	Technical Library, 04536