ABSTRACT

An Evaluation of Attack Tree Analysis Using a Structured Query Language-Based Simulation

by

Michael S. Pallos

M.B.A., Nova Southeastern University, 1999
B.S., Nova Southeastern University, 1996
A.S., Saint Petersburg Junior College, 1988

Dissertation Submitted in Partial Fulfillment
of the Requirement for the Degree of
Doctor of Philosophy
Applied Management and Decision Sciences

Walden University
May 2005

ABSTRACT

This research evaluated the effectiveness of attack tree analysis incorporated into an information system computer security risk assessment methodology. By evaluating the effectiveness of using attack tree analysis to assist with costing decisions, probability analysis, and the viability of using a structured query language (SQL) computer program simulation model developed as part of this research. Attack tree analysis is a risk assessment methodology used to identify vulnerabilities and penetration points of a system based on the goals of the attacker.

A pre- and postassessment instrument was developed to ascertain the effectiveness of using attack tree analysis. The purposeful sample was comprised of fifty-six computer security experts and leading academic authorities of attack tree analysis. The hybrid methodology incorporated quantitative data analysis using the chi-square test of homogeneity and the test for the equality of proportions; qualitative data analysis included the use of grouping of data creating bar graphs, discussions, conclusions, and other narrative components.

The quantitative findings suggested a strong support base for the use of attack tree analysis to assist with costing analysis, probability modeling used for human resource allocation, and a structured query language simulation model, ranging from 71.4% to 92.9%, whereas only 21.4% to 28.6% of participants considered implementing attack tree analysis to assist with the above mentioned managerial challenges. The qualitative data suggested the transition from theory to implementation may not be achievable.

The value of attack trees as a tool to enhance security is not limited to information systems. Many facets of society that use complex systems, such as public policy and homeland security efforts, may benefit from this research. The findings suggest that attack tree analysis has the potential for positive social change based on a more secure global infrastructure.

An Evaluation of Attack Tree Analysis Using a Structured Query Language-Based Simulation

by

Michael S. Pallos

M.B.A., Nova Southeastern University, 1999
B.S., Nova Southeastern University, 1996
A.S., Saint Petersburg Junior College, 1988

Dissertation Submitted in Partial Fulfillment
of the Requirement for the Degree of
Doctor of Philosophy
Applied Management and Decision Sciences

Walden University
May 2005

DEDICATION

To my partner, best friend, and the love of my life, Laura Anne Pallos; thank you for sharing the experiences and journey of your life with me – with us; and for taking such good care of my heart. “I am my beloved’s and my beloved is mine” (Song of Solemn, 6:3); now and forever.

My precious daughter, Danielle, and remarkable son, Mike, the two of you make life worth living and all of the challenges merely trivial obstacles to be overcome. It has been and always will be worth it. I love being your dad.

ACKNOWLEDGEMENTS

I would like to express my gratitude and appreciate to my Ph.D. advisor, Dr. Pamela Wilson, for her guidance, teaching, and encouragement over the years during this journey.

I am profoundly grateful to my dissertation committee, Dr. Ruth Maurer, for her exceptional guidance, statistical analysis facilitation, and continued motivational support – you allowed me to focus on the light at the end of the tunnel as a ray of hope and not an oncoming train! Dr. William D. Steeves, Jr., for his exceptional guidance, non-technical thought provoking feedback, and meticulous detail. Dr. Raghu B. Korrapati, for his exceptional guidance, detailed feedback, attention to details, and continued motivation for publication.

I am grateful to the tireless editors responsible for transforming my writings, my lovely wife, Laura Pallos, and linguistically talented Argero P. Robertson.

I would like to thank my family and friends for their inspiration, frequent encouragement, and motivational persuasion in moving me towards the completion of this portion of my journey.

I am grateful to my colleagues and managers who over the last few years, during this educational journey, supported my efforts in more ways than one. Thank you for your contributions in allowing me to achieve this goal – Christos Papadopoulos and Roger Butterworth.

I am grateful to Bruce Schneier for his publications which gave birth to Attack Tree Analysis and time investment in our phone conversation and emails.

Finally, but not lastly, thank you to all of the research participants who invested the time to complete both surveys, work with the SQL simulation model, and further the body of knowledge on attack tree analysis. Thank you for your time, effort, and feedback.

TABLE OF CONTENTS

LIST OF TABLES

Page

LIST OF FIGURES

Page

CHAPTER 1: INTRODUCTION

Introduction

The Internet was created in the 1960s as a medium to facilitate universities in order to freely share information (Leiner et al., 2003). The sharing of ideas, exchange of documents, and collaboration are core values upon which the Internet was built. Its simplicity of use and open communication with it invited others to join in this sharing environment. As the Internet grew, corporations began participating. The military, which once had an independent network, began to use the Internet as well. As of 2005, the Internet had grown to 317,646,084 host computers utilizing a computer infrastructure shared by corporations, governments, military agencies, and private citizens (Internet Domain Survey, 2005).

The famous bank robber Willie Sutton was once asked why he robbed banks, to which he promptly responded, "Because that's where the money is." In today's marketplace $1,219,713,000 worth of business is transacted via the Internet on an annual basis. The amount of business dollars transacted via the Internet is projected to hold to single digit growth through the year 2006 (Rosall, 2002).

Businesses such as insurance and financial institutions once had the luxury of maintaining control over their business and client data by dictating how that data could be used and accessed (Rosall, 2002). The ubiquitous presence of the Internet and the customer demands and regulatory requirements for doing business via the Internet have forced most businesses and entire industries to change their paradigm in order to remain competitive. This paradigm has created a lucrative target for hackers and other information "thieves," while at the same time giving these individuals an entirely new set of access points into corporate databases. Information managers are challenged with incorporating risk assessment and threat analysis models as a baseline to assist with

identifying potential penetration points. Unfortunately, most risk assessment models (Andrews & Moss, 2002) such as fault tree analysis (Elliott, 1998; Ericson, 1999; Helmer et al., 2000) and failure mode and effect analysis (Elliott, 1998; Huang, Shi, & Mak, 1999) were created to identify failure points within a system, or to perform postmortem analysis of catastrophic events.

What information systems managers appear to be lacking is a methodology which takes a holistic perspective of a system's penetration points, including, but not limited to, access points external to the system (Schneier, 2000). An example of a system attacker who uses more than the Internet as a means of gaining sensitive corporate data is a hacker who rummages through an organization's dumpster searching for documents containing vital information. One risk assessment model that considers the holistic perspective of system penetration points is attack trees.

Schneier (1999, 2000) first introduced the concept of attack trees in a paper co-authored with the National Security Agency (Salter, Saydjari, Schneier, & Wallner, 1998), and expanded on the notion in an article in *Dr. Dobbs Journal* the following year (Schneier, 1999). Attack trees provide a process for identifying penetration points throughout all components of a system.

Background to the Problem

The concept for this study began with five observations on the part of this researcher. First, the need for computer security was growing larger than any other sector in the information technology industry (Witty, R., Dubiel, J., Girard, J., Graff, J., Hallawell, A., Hildreth, B., et al., 2001). The Gartner Group research firm expects computer security to experience a 40% growth rate through the year 2006 (Witty et al., 2001). A demand in the computer security market space that may be addressed by attack

trees (Ellison & Moore, 2001; Salter, Saydjari, Schneier, & Wallner, 1998; Schneier, 1999, 2000).

Second, when reviewing the literature on computer security in the information technology area, Schneier (2000) introduced the notion of attack trees. Attack trees allow an analyst to build a hierarchical representation of systems vulnerabilities, not merely focusing on the technical aspects. An attack tree is built with the attacker's goal as the root node. For example, if an attacker wishes to rob the contents of a safe, the goal will be *Open Safe*. Attack trees offer a unique perspective to computer security; that is, they assume the attacker's viewpoint when considering an attack. According to Schneier (1999, 2000) the use of attack trees to perform analysis on a system often reveals penetration points that the systems designers had not considered.

Third, attack trees offered a practitioner's perspective lacking academic substantiation. Ellison and Moore (2001, 2003), Salter, Saydjari, Schneier, and Wallner (1998), and Schneier (1999, 2000) suggested the use of attack tree to assist with the quantification of an attack by assigning Boolean and continuous values to a node or leaf in the attack tree. There appeared to be substantial opportunity to validate Schneier's implications.

Fourth, the morning of September 11, 2001, witnessed horrific events in America, as terrorists destroyed the Twin Towers of New York City. The architects of the Twin Towers had not fully examined the structural possibilities of an airliner, full of fuel, crashing into the buildings. According to Robertson (2003), chief structural engineer of the World Trade Center, the towers could withstand a direct hit from an airliner, specifically a Boeing 707 moving at 600 miles an hour. Mr. Robertson was correct; the towers did withstand a direct hit from an airliner. It was only after the fuel ignited and resulting heat was able to melt the infrastructure of the tower that they fell.

Vulnerabilities exist in systems that elude the normal thought process of safety. When examining potential system vulnerabilities, one must begin to think from the attacker's perspective. One such system that is ripe for attack is the Internet.

America's reliance on the Internet for commercial, governmental, military, and personal use, as reflected in Figure 1, is growing at a rate of approximately 50,000,000 Internet host computers a year (Internet Software Consortium, 2005). Risk assessment methodologies must encompass the breadth required to identify system vulnerabilities that terrorists may exploit.

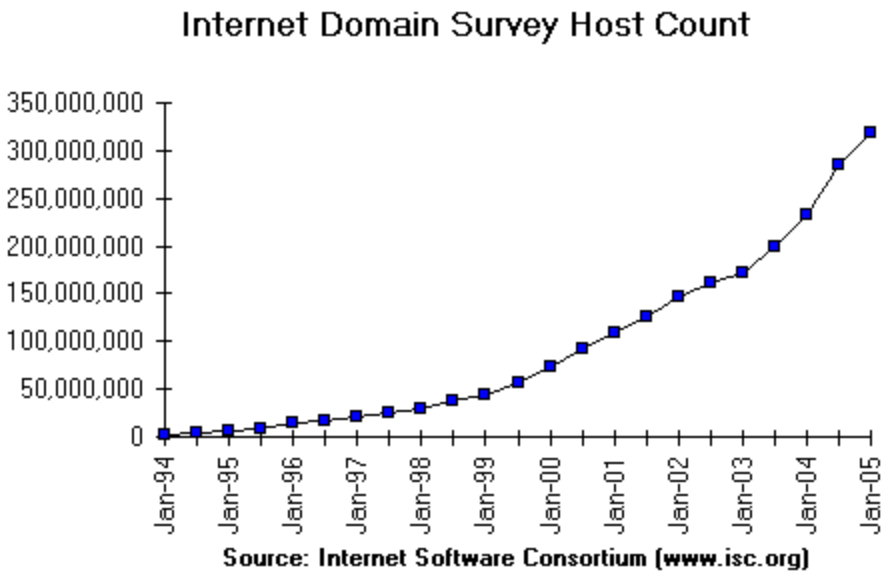

Figure 1. Internet host computers growth rate (Internet Domain Survey, 2005).

Fifth, this researcher, on a business plane trip, was seated next to a senior security consultant who held a top-secret clearance (personal communication, September 15, 2003). During discussions of the current information technology and computer security landscape, the conversation led to systems vulnerabilities and penetration points for

attackers. The security consultant revealed that he had once performed a security assessment on a highly secure site. Three layers of physical security were required to gain access to the computer center, such as what one would expect to encounter upon entering Fort Knox. After passing through the three stop points requiring identification and housing secure doors, a guard finally granted access to the computer room. During an initial assessment, the consultant noticed direct dial-in phone lines connected to the main computer, enabling technical support to gain access in order to fix any problems. With all the high physical and digital levels of security an attacker with a phone line would have been able to gain access to this system. The computer security expert suggested that many other systems have penetration access points which exist outside of the scope of the systems designers that lack a methodology for identification. The point was also raised that an assessment methodology that incorporated an attacker's view of the holistic system may have helped identify additional vulnerable access points.

The software industry has experienced an extensive reduction since the collapse of the dot-com era. Prior to the collapse of 2000, the Internet was growing at an annual rate of 239% (Rosall, 2002). Information system manager budgets were essentially flat in 2003 with an anticipated increase of only 4% in 2004 (Gomolski, 2003). Managers who face a reduced budget are not experiencing a reduction in responsibility level. Schneier (2000) suggested that attack tree analysis could assist costing analysis relative to securing assets. Therefore, this researcher began wondering if attack tree analysis could assist computer security information systems managers with costing decisions based on attack tree analysis. Specifically, it appears as though the information ascertained from attack tree analysis combined with costing analysis identified where funds should be allocated by a computer security information system managers to receive the greatest return on his/her investment.

As the initial literature review revealed, fault tree analysis (Andrews & Moss, 2002; Aven, 1992; Harrington & Anderson, 1999, pp. 131-132; Sawma, 2002; Welch et al., 2003) and event tree analysis (Andrews & Dunnett, 1997; Andrews & Moss, 2002; Aven, 1992; Herzog & Shahmehri, 2001) are the most cited methodologies used in the risk assessment of information technology. Fault Tree analysis quantifies the decision process implementing Boolean algebra and probability (Andrews & Moss, 2002; Elliott, 1998; Ericson, 1999). Since attack tree analysis appear to possess similar characteristics as Fault Trees, it appears to be possible to quantify attack trees with Boolean algebra and probability.

This research formulates many questions about the potential of attack tree analysis. Is the use of attack tree analysis capable of identifying system vulnerabilities that are currently unknown to management? Can the cost values be assigned to nodes in the attack tree providing management with information that assists them to make more informed decisions and allocate corporate financial and manpower resources more wisely? Do attack trees work, and can the attack tree process be automated with a computer program? To the best of this researcher's knowledge the answers to these questions do not exist in a quantifiable academic arena, only in practitioner speculation. If attack trees do meet the positive claims made by the peer reviewed articles and publications (Ellison & Moore, 2001, 2003; Salter, Saydjari, Schneier, & Wallner, 1998; Schneier, 1999, 2000), this study hoped to validate and produce a useful model direction to assist information systems managers with security decisions and risk assessments of their information system's infrastructure.

Attack tree analysis describes the security or vulnerability of a system based upon the goals of the attacker (Schneier, 1999, 2000). The problem is that attack tree analysis is in its infancy and lacks an in-depth academic study and rigorous testing (Daley,

Larson, & Dawkins, 2002; Ellison & Moore, 2001; Salter, Saydjari, Schneier, & Wallner, 1998). This study contributes to the body of knowledge that seeks to substantiate the process of attack tree creation and develop a mathematical quantification procedure aiding governments and industries in mitigating computer security threats.

Problem Statement

The information systems discipline continues to expand, as does reliance on advancing technologies. With this growth in information systems comes a growth in system exposure, risk, and vulnerabilities. According to Howard (1997) and Cert (2004) there are thousands of reported Internet break-ins each year.

The computer security industry does not currently possess a single computer security risk assessment model. The available options appear to be a mixture of diverse models combined with hybrid models, mostly leveraging models from the nuclear power industry (Herzog & Shahmehri, 2001). According to White (1995) most risk assessment models fail to take a holistic view arising out of whole systems. One computer security model which appears to take a holistic view of a complete system is Attack Tree Analysis. Attack trees, as introduced by Schneier (1999), offer such a model; however, validation of attack tree analysis, using costing and probability, is absent in the academic literature (Daley, Larson, & Dawkins, 2002; Ellison & Moore, 2001, 2003; Salter, Saydjari, Schneier, & Wallner, 1998). This study researched the ability to perform calculations based on costing and probability claims made by Schneier (2000, p. 323) regarding the uses of attack trees in risk assessment and security analysis in an attempt to partially fill this gap.

The literature review did not produce a link between application and theory with attack trees (Ellison & Moore, 2001, 2003; Salter, Saydjari, Schneier, & Wallner, 1998;

Schneier, 1999, 2000). This research attempted to address the perceived application and theory chasm. It's focus included risk assessment costing analysis, quantifying system vulnerabilities using mathematical formulas, the automation of attack tree costing, and vulnerability assessment built algorithms contained in a software application. Risk assessment costing analysis provided a computer program for information managers as they decide where to invest their budget in order to achieve the greatest benefit from that expenditure. The use of mathematical formulas provided a similar tool for these same managers to identify critical components of their system and efficiently allocate countermeasures in the form of time, money, and human resources. Finally, the use of the software application provided an automated means to implement these methodologies in order to make the process accurate and efficient.

Research Questions

As a way of approaching validation of attack tree analysis, using costing and probability, the following questions guided this research.

1. How effectively might the inclusion of attack tree analysis be incorporated into a computer cost analysis model capable of assisting information systems managers with budgetary decisions?

2. How effectively might the inclusion of attack tree analysis be incorporated into a computer probability model capable of assisting information systems managers with human resource allocation?

3. How effectively might the inclusion of a structured query language (SQL) database program be implemented to simplify the use of a cost analysis model and a probability model to assist information systems managers with costing and human resource allocation decisions?

Purpose of the Study

The purpose of this study was to research the effectiveness of attack trees, using costing and probability, incorporated into an information system computer security risk assessment methodology. This research evaluated the effectiveness of using a computer program incorporating attack tree analysis to assist with costing decisions and probability analysis. To assist information systems managers with the above-mentioned process, a deliverable from this study included the creation of a computer program that assisted with the costing and probability decisions information systems managers made with the use of attack trees.

Conceptual Framework for the Study

Conceptual support for this study originated in a collaborative paper authored by a representative of corporate America and the National Security Agency (Salter et al., 1998) introducing the notion of a movement towards a secure system engineering methodology. The creator of attack trees expanded upon this notion as attack trees were introduced to the masses in a trade journal (Schneier, 1999), then to a larger audience in Schneier's (2000) book on computer security.

Explanation and academic use of attack trees can be found in master’s theses (Moberg, 2001; Selliah, 2001; Varner, 2001), peer reviewed journals (Ellison & Moore, 2001,2003; (Vidalis & Jones, 2003), technical reports (Ellison & Moore, 2003; Fumy et al., 2003; Vidalis & Jones, 2003), industry proceedings on computer security (Daley et al., 2002; Ericson, 1999; Tidwell et al., 2001), governmental agencies (Bieber, 2000; Salter et al., 1998), and software developed to assist with tree creation (Amenaza Technologies Limited, 2001).

Conceptually, Salter, Saydjari, Schneier, and Wallner (1998), Schneier (1999, 2000) set the foundation for the use of attack trees assisting information system managers with costing, penetration, and probability analysis. Costing decisions as to where one should invest financial resources to achieve the greatest protection from a systems attack may be achieved by using attack tree analysis. The authors have also introduced the notion that attack trees will also assist information systems managers with making decisions re: weaknesses in the current system based on exposed penetration. Finally, Schneier (2000) introduced the concept that probability analysis can be applied to attack trees providing information system managers with additional information about the system.

Assumptions

The strength of the one-group pretest-posttest research design is dependent upon the degree to which the effects of the program can be accurately measured. This limited the conclusions the researcher could draw about the effectiveness of the model. Other potential weaknesses inherent to the one-group pretest-posttest design are maturation and history. According to Singleton and Straits (1999), the longer the period between the pretest and the posttest the higher the likelihood that either of these may confound the results. “Additional threats to internal validity-testing, instrumentation, and sometimes statistical regression may present rival explanations to the hypothesis in” (p. 216) the one-group pretest-posttest. Even though the one-group pretest/posttest design has relatively low power in terms of determining causality, it can provide useful information for designing more rigorous follow-on studies (Burns & Grove, 1993; Franklin & Thrasher, 1976; Singleton & Straits, 1999).

The major assumption of this research is that the computer program developed for it accurately reflected the effectiveness of using attack trees to assist with costing decisions, probability analysis, and the viability of using a structured query language (SQL) computer program to supply that assistance. Additional assumptions included the participant's ability to effectively use the tool. Detailed assumptions included the following:

1. The computer program was capable of accurately reflecting the effectiveness of attack tree analysis as related to costing decisions and probability analysis.
2. Analysis of the researcher's experience, relevant literature review, and the pretest/posttest instruments accurately reflect the abilities of attack analysis as related to costing decisions and probability analysis.
3. The target population sample in this study was familiar with information technology, attack trees, and have an understanding of computer security.
4. The target population sample in this study was capable of assessing the SQL simulation model computer program.
5. The creation of the attack tree used for this research was limited to the researcher's current depth of knowledge in computer security and understanding of web server attacks and penetration points.

Scope and Delimitations

Contained within the bounds of this research effort was the creation of a computer program that was used to assist with using attack tree analysis to support with costing and probability models. The intent was to develop a tool that managers could use to assist with cost benefit analysis, probability assessment, and Boolean algebraic mathematics. The limitations of this tool have been partially realized and are described in the

limitations section of chapter 5. The software program's primary focus was on producing valid results and not on the presentation of the information itself.

Attack trees have been introduced into the information systems industry within the last 5 years (Schneier, 1999, 2000) as a means of assisting information systems with risk assessment and threat analysis. Review of the literature has produced no doctoral level studies using attack trees in quantitative and qualitative forms. There have been masters' theses that have incorporated the attack tree methodology (Moberg, 2001; Sawma, 2002; Selliah, 2001).

The completed evaluation research may have partially bridged the gap between applied and theoretical notions by testing an attack tree model (Singleton & Straits, 1999). The attack tree model program included the utilization of probability and Boolean algebra to assist with attack tree analysis. The proposed model also introduced a means of performing cost analysis providing information systems with the necessary data to help decide where to invest funds most beneficial to the security return.

The methods included the creation of an attack tree model with the creation of a computer program to automate the algorithms required when performing the analysis. In addition to the creation of a computer program, pre- and post-instruments were developed to assess the computer program's value and by default, the extended attack tree model. The computer program was submitted to a group of 56 computer security experts and information systems managers via a purposive sample. The group received a cover letter explaining the value of the study, a computer program, and a link to the survey instruments. The data was collected via a survey using the created instruments and produced quantitative and qualitative data. Clarification was required, or to follow up on participants who have failed to complete the survey, participants were called via the telephone and the phone interview rescheduled.

Limitations

The limitations of this research included the researcher's lack of extensive knowledge of the sample target's environment, that is, the technical landscape in which the sample target operates its practitioner and academic duties, as well as the sample population's experience with implementing attack tree analysis. If the attack tree program was populated with an attack tree that mapped directly to the sample target's own technical landscape, the value and degree of knowledge as to the literal attack tree itself could have contained a deeper understanding, allowing the sample targets to focus more on the costing and probability aspects as opposed to the structure of the attack tree.

Using the computer program to assist with risk assessment had the same limitation of all computer programs; that is, the information produced relies heavily on the data input into the system. The term *GIGO* (garbage-in garbage-out) is relevant.

External validity may be limited due to the methodology model incorporated. A purposeful sample targeted the leading attack tree experts based on the literature review. Incorporation of the expert's knowledge as a reflection of the actual results one may obtain from a random sample of the entire Internet domain may vary.

Research Design

The design of this study was an evaluation research of attack tree analysis using a SQL based simulation model by means of a computer program. The SQL based simulation model researched attack tree analysis assisting with costing analysis and probability analysis as a means of providing information systems management with information to assist in costing and human resource allocation decisions.

The data-gathering technique included a purposive sample of 56 computer security experts and leading academic authorities on attack trees whose responsibilities include computer security, information systems, publication, and graduate and doctoral level education (Singleton & Straits, 1999). The data was gathered using pre- and post-instruments that were developed as part of this research. Walden University professors who have extensive content knowledge and attack tree subject matter experts were used to validate the instruments. Data presentation includes a mixed model approach including qualitative and quantitative analysis.

Significance of the study

This study adds to the body of knowledge existing for the risk assessment of computer security systems while potentially providing an academic quantification of attack trees viability and usefulness with costing and probability analysis to information systems managers, government agencies, military organizations, and private citizens who have home computers connected to the Internet. As requested by Salter, Saydjari, Schneier, and Wallner (1998, p. 2), this study partially bridged the gap and facilitated "dialog among academia, industry, and government toward securing the global information infrastructure."

The process of developing attack trees was automated by a computer program that housed the mathematical properties contained within computer algorithms incorporating probability, Boolean algebra, and cost benefit analysis that aided information systems managers and security consultants in system analysis (Schneier, 1999, 2000). This program and process may have aided in the ability to run countermeasure scenarios and "what-ifs" also adding to the security of information systems.

Many aspects of society that incorporate information systems may benefit from the results of this study. This positive social change includes a more secure society obtained in a cost effective manner identifying the best use of human resource allocation. The positive social change may be founded in the proposed algorithms assisting with the costing and probability protocols. Social entities that may benefit include governmental organizations that may be able to reduce the risk of terrorism by identifying vulnerabilities and penetration points previously unrecognized. Additional social entities that may benefit include corporations such as electrical companies and the airline industry, which may be able to identify where to invest funding in order to achieve the highest benefit in countering terrorism and threats. As requested by Salter, Saydjari, Schneier, and Wallner (1998, p. 2), this study partially bridged the gap and facilitated "dialog among academia, industry, and government toward securing the global information infrastructure." This research may help to guide society into a more secure information technology infrastructure by evaluating a risk assessment model capable of identifying and reducing vulnerabilities in systems, processes, and policies.

Finally, the results of this study and the processes produced may aid all interested parties in the effort to reduce the risks of exposure. These risks often exist external to the software applications themselves, and mitigate risk in the most cost effective manner ensuring the highest probability for success.

Definitions of Terms

Asset. An asset is a resource of value including hardware, software, intellectual property, and tangible and intangible resources (Microsoft, 2003).

Attack. The process in which an asset is harmed, compromised, or exposed.

Attack Tree. Ellison and Moore (2003) define an attack tree as "a mission-critical compromise of a system and a hierarchical organization of intrusion scenarios, each of which accomplishes that compromise by different means." The creator of attack tree Schneier (2000) describes them as a "methodical way of describing threats against, and countermeasures protecting, a system." Attack trees will de covered extensively providing an in depth explanation and analysis in chapter 2.

Attackers. Attackers are human beings categorized by Howard (1997) as "corporate raiders," "crackers," "hackers," "professional criminals," "spies," "terrorists," and "vandals" who engage in a "single unauthorized access attempt, or unauthorized use attempt, regardless of success" (p. 287)

Countermeasure. A safeguard implemented to mitigate risks, reduce penetration points, and address a threat (Microsoft, 2003).

Penetration. Penetration occurs when an attacker gains unauthorized access to a system (Helmer et al. 2000).

Risk Analysis. "The identification and evaluation of the most likely permutation of assets, known and anticipated vulnerabilities, and known and anticipated types of attackers" (Durrett, 2003).

Threat. A threat is a process or method in which vulnerability may be exploited (Amenaza Technologies Limited, 2001).

Virus. A virus is a small computer program that reproduces and propagates itself by attaching to another computer program (Thunstrom & Ahs, 2003).

Vulnerability. Vulnerability is a weakness, feature, or some aspect of a system that makes a threat possible (Microsoft, 2003). Vulnerability as described by Kabay (1996) is an area or point where a system is vulnerable to an attack. Pfleeger (1997) expands the definition to include any limitation in a system that may be exploited causing

harm or loss. Finally, Blyth (2001) expands vulnerability to include a limitation in a system that could allow security to be violated.

Summary and Organization of the Study

Chapter 1 introduced the notion of attack trees and their proposed ability to assist information systems managers, corporations, governments, people, and society with computer security and risk assessment by identifying penetration points in a system that an attacker may leverage. A conceptual basis for attack tree has been produced in the foundation laid by Salter, Saydjari, Schneier, and Wallner (1998) and Schneier (1999; 2000). The body of knowledge on attack trees appears to lack a clear melding between applied and theoretical. The purpose of this study was to fill that applied and theoretical gap using a methodology that will quantify attack tree uses and validity in computer security risk assessment by identifying penetration points in a quantifiable fashion using probability, Boolean algebra, and a cost benefit model.

Chapter 2 reviews the leading risk assessment methodologies currently being used by the information systems. The leading techniques used to assess risk are identified in the literature review. The ten techniques explored by the research effort can be grouped into four categories: a) qualitative, b) tree-based, c) dynamic, and d) hybrid. Each group is reviewed in chapter 2.

Chapter 3 contains the methodology in a research design structure that is the blueprint for this dissertation. This research used an evaluation research that may have bridged the gap between applied and theoretical notions by testing an attack tree model. Chapter 3 includes the architecture, design, and pseudo-code of the computer program. The concepts contained within the pre- and post-instruments to be used for data collection are also found in chapter 3.

Chapter 4 describes the results of the dissertation research. The three research questions are evaluated through data analysis of the pre- and post-survey results. The information is explained in tables, graphs, and written description. Variables include familiarity of attack tree analysis, costing analysis, probability analysis, and use of a structured query language (SQL) simulation model built in a computer program.

Information included in chapter 5 contains a discussion of the results including interpretations and conclusions drawn from the findings. The limitations of the study are explored as well as the implications the findings may have on information systems managers. Each section of chapter 5 ends with a discussion of the recommendations that information system mangers may wish to incorporate.

CHAPTER 2: LITERATURE REVIEW

The literature review strategy incorporated a comprehensive search of computer security risk assessment models. The literature review contains the most often cited methods for identifying, assessing, and managing risk along with other related theories. One of the components examined in the risk models included their effectiveness and ability to assist information managers with budgetary decisions and resource allocation based on costing models and probability models represented in an automated process. The risk assessment models are presented in alphabetical order.

The notion of probability is fundamental to the assessment of risk. The risk assessment models introduced here identify several processes used to identify risks and are utilized today in leading industries.

Analytic Hierarchy Process

Saaty (1990) created the analytic hierarchy process (AHP) in 1980 to assist with the complex decision-making process managers often face. These complexities of assessing decision tradeoff and balance points, such as hard costs with soft intangible benefits, include a process. For example, the computer hardware in a corporate call center is a hard cost. An example of a soft cost derived from the corporate call center is the value customers receive when the call is effectively processed.

The AHP process addresses the above-mentioned complexity by utilizing a combination of quantitative and qualitative measures assisting with decision-making in the risk assessment arena. One of the themes of AHP is not only to arrive at a completed assessment, but also to capture the decision process experience responsible for arriving at the assessment's conclusion. To achieve this objective the AHP includes (a) the building of a hierarchy model representing the objectives, criteria, and alternatives; (b) pairwise

comparison of problem attributes; (c) synthesis scale, and (d) sensitivity analysis (Saunders, 2000).

Hierarchies

The problem domain is captured using a graphical representation built with hierarchies. This visual representation allows all of the components to be captured in a single flow. In Figure 2, Saunders (2000) provided a horizontal hierarchy for prioritizing the budget items in a single, simplified aspect of information security.

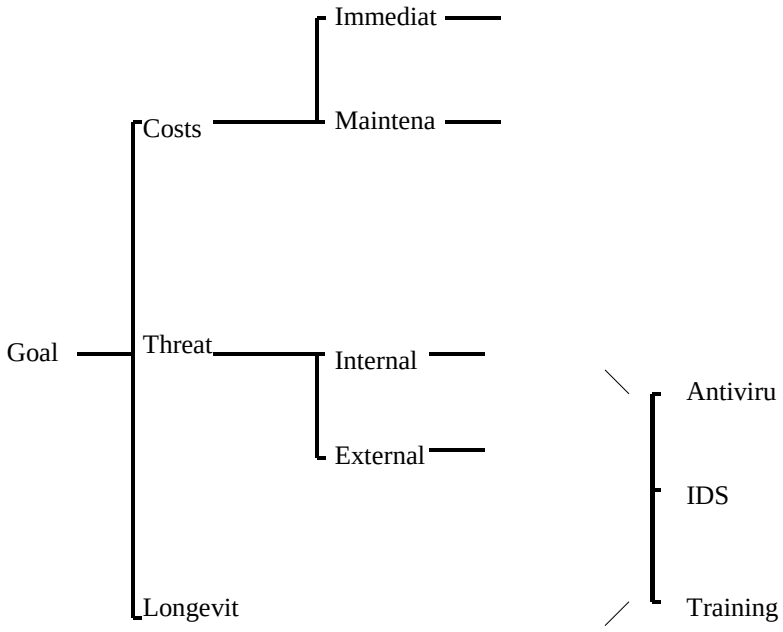

Figure 2. Prioritize security initiatives (Saunders, 2000, p. 3).

Each hierarchy in the AHP process includes a definition table. This table allows abbreviated names to be used in the graphic hierarchy representation allowing for a

cleaner looking diagram. The definition table includes a multiword definition of the abbreviations. The table for the hierarchy found in Figure 2 is reflected below in Table 1 (Saunders, 2000).

Table 1.

Prioritize Security Initiatives

Abbreviation	Definition
Goal	
$Cost	
Antiviru	Upgrade level of antivirus software protection
External	Likelihood of external attacks occurring
IDS	Install Intrusion Detection System
Immediat	Immediate purchase and acquisition costs
Internal	Likelihood of internal attacks occurring
Longevit	Longevity of solutions
Maintena	Longer term maintenance costs
Threats	Likelihood of attacks/malfeasance occurring
Training	Allocate more budget dollars to user and network admin security training

Paired Comparison

AHP allows for two objects in the hierarchical tree to be compared to each other, essentially allowing a weighted value to be assigned (Saunders, 2000). The weights for the two objects total 100%. During the paired comparison analysis the analyst identifies the importance of the first variable relative to the second variable. Using a scale of 1 to 9, the user assigns the value to the first variable, the remaining percentage to the second variable. Paired comparison allows the analyst to take a large hierarchical tree and only focus on one aspect, creating a more manageable task.

Synthesis

The synthesis component of AHP allows one to incorporate the paired comparison process for generating a numeric value to be assigned to each hierarchical object. One is not required to arrive at purely quantifiable values. The use of qualitative methods is incorporated using a common numeric ratio scale mechanism also created by Saaty (1990). The ratio scale method uses paired comparison values as input criteria for the ratio-scale matrix. The matrix derives weights corresponding to each criterion (Durfee et al., 2000). The mathematics utilized in the ratio scaling method is a combination of supermatrices, eigenvalues, and eigenvectors, which all collect qualitative information from the user and produce a number value (Saaty, 1990).

Sensitivity Analysis

Sensitivity analysis provides the "what-if" mechanism to the AHP model allowing the analyst to adjust weights assigned to objects in the hierarchy. The effects of changing a weight are then rippled down through the hierarchy and impacts may be observed. AHP has existed for over 20 years and has been applied to a diverse grouping of risk

management applications such as air traffic control, energy, national security, and information technology (Saunders, 2000).

AHP utilization of ratio scale measurements is one of the key strengths to the methodology. For example, a security risk of 5 would be half as threatening as a security risk of 10 (Knott, 2002). AHP requires a level of mathematical complexity, which prevented the methodology from gaining market and academic acceptance; however, the creation of AHP software applications, such as Expert Choice (2003), have simplified the process allowing the software users to enter data modifying the scales and weights by sliding a graphical bar back and forth. This simplification and automation of the process has lead to greater industry acceptance.

Attack Tree Analysis

Attack tree analysis was created by Schneier (1999; 2000). Attack trees describe the security or vulnerability of a system based upon the goals of the attacker. For example, if an information systems manager were responsible for an order fulfillment system that contained customer information, including credit card detail, an attacker may have the goal of stealing the customer's credit card data. This goal "steal credit card data" is the starting point or root node, also known as the goal, of the attack tree. The attack tree is then extended, building branches down the tree to identify the different sub-goals and penetration points available to the attacker. The branching process continues as the means of penetration are decomposed to the lowest level of intrusion, known as the leaves.

A single computer system will most likely contain several attack trees since an attack tree represents the satisfaction a single goal. Information systems attackers may evaluate a system from different perspectives requiring the creation of numbers of attack

trees to cover the vulnerabilities of a single system. The consolidation of numerous attack trees is known as an attack forest (Moore, Ellison, & Linger, 2001).

The attack tree approach allows analysts to rethink system vulnerabilities from the attacker's perspective (Schneier, 2000). This perspective expands the software or system vulnerabilities model, as defined by Krsul (1998), to include elements other than the software application, such as the example stated previously of an attacker rummaging through a dumpster searching for sensitive documents containing password information or the enterprise architecture.

Original Structure

Attack trees are represented graphically and textually. The graphical representation is usually built with the root node, or goal, on the top. The tree then descends branches and sub-goals until the leaves are finally reached at the bottom level (Schneier, 1999, 2000). Figure 3 contains the conceptual model of an attack tree represented in the graphical format.

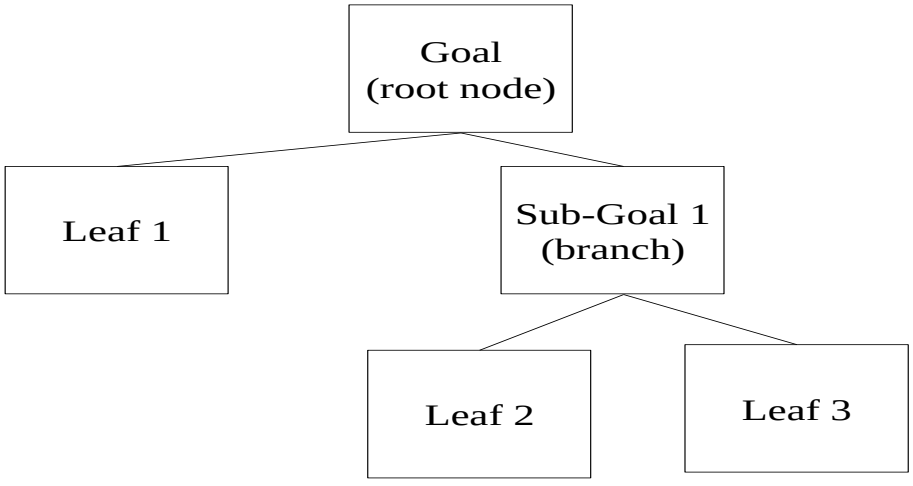

Figure 3. Conceptual view of a graphical attack tree.

The textual representation of an attack tree follows a numeric outline structure. The root node or goal is represented at the one (1) level with no indentation. Each sub-goal is then numbered accordingly and indented one unit per level of decomposition. Figure 4 represents the textual view using the same example content found in Figure 3.

1. Goal (root node)
    1.1 Leaf 1
    1.2 Sub-Goal
        1.2.1 Leaf 2
        1.2.2 Leaf 3

Figure 4. Textual view of an attack tree.

To leverage Schneier's (2000) example of an attack tree, the attack tree reflected in Figure 5 represents an attack against a safe in which the attacker wishes to open the safe obtaining the contents.

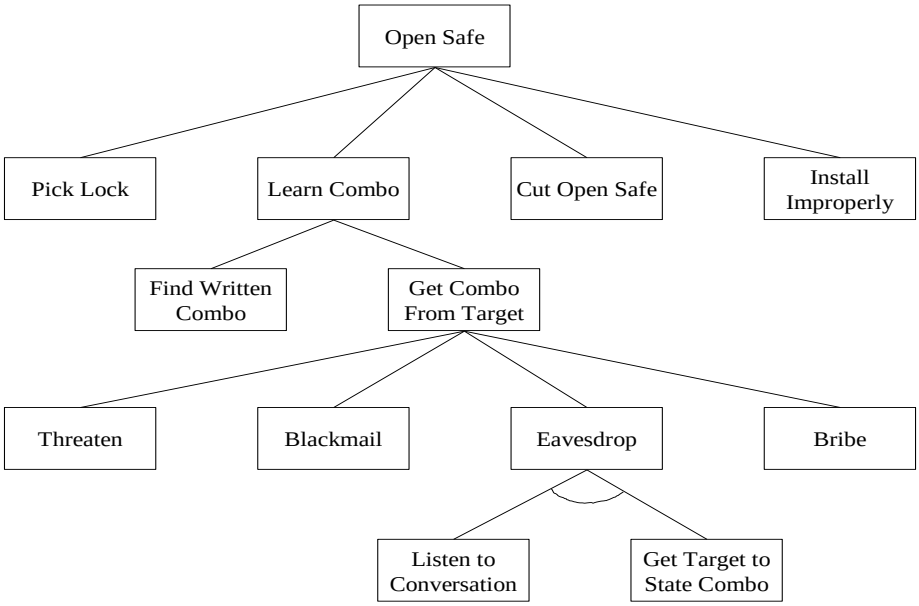

Figure 5. Attack tree to open a safe (Schneier, 2000).

In the attack tree paradigm, the tree is built with the main goal as the top node. Decomposition of each additional node becomes a sub-goal of the primary goal. As one traverses down the tree the bottom leaf represents the diverse suite of opportunities available for the attacker to choose from in order to achieve the primary goal. For example, at the first level of nodes, if the attacker were to *Pick Lock* then she would be able to open the safe; however, if her accessibility was derived from bribery, she would continue down the tree until the *Bribe* leaf were reached (Schneier, 2000).

In the creation of attack trees one will face scenarios in which multiple objectives must be ascertained simultaneously in order to achieve the desired goal or sub-goal. To accommodate this requirement, attack trees incorporate the use of AND nodes and OR nodes. All nodes are implicit OR nodes. AND nodes represent nodes in which both tasks must be accomplished to achieve the goal or subgoal. In Figure 6, a pair of AND nodes are located at the bottom right of the diagram in nodes *Listen to Conversation* and *Get Target to State Combo*. Both nodes are required to successfully accomplish the subgoal of *Eavesdrop*. The explicit AND nodes are represented by a semi-circular line that connects the two node lines (Ellison & Moore, 2001, 2003; Schneier, 1999, 2000).

Once an attack tree has been created, values can be assigned to each node. An example provided by Schneier (2000), shows that to obtain a cost model associated with the risk, cost values are assigned to each node, as reflected in Figure 6.

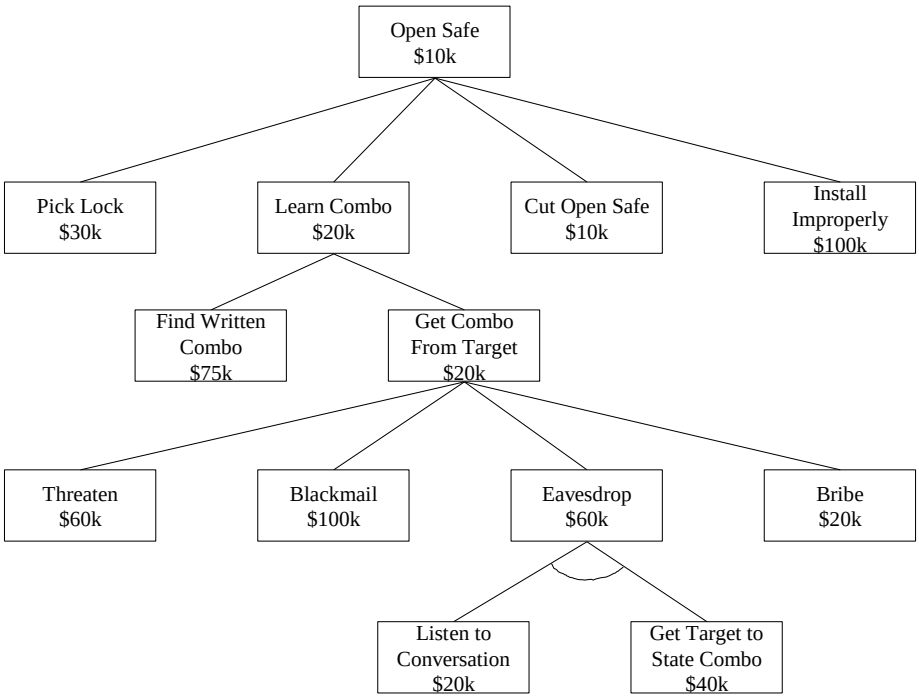

Figure 6. Open safe attack tree with costing details (Schneier, 2000).

Schneier (2000) suggests that once values have been entered, analysis can be run against the attack tree. Management can identify the most cost effective measure to be implemented provided with a given fiscal budget. Security cost justification can be produced allowing decision makers to identify the cost associated with each area of vulnerability.

In addition to continuous values, such as dollar amounts, attack tree nodes can also be assigned Boolean values to represent additional perspectives on the attack tree. These perspectives expand based upon the corporation or threat vulnerability. For instance, legal versus illegal, requiring special equipment or no equipment needed, skilled versus non-skilled, possible versus impossible, et cetera (Schneier, 2000, p. 320).

Once an attack tree has been created, there is an opportunity for reuse. For example, if an attack tree has been created for *Steal Database* at any point that an additional database is introduced into the environment, the previous attack tree can be leveraged (Schneier, 2000).

The nongraphical outline representation is recommended for complex systems (Schneier, 2000) an example of an outline for *Open Safe* is:

Goal: Open Safe

1. Pick lock
2. Learn combo
   - 2.1. Find written combo
   - 2.2. Get combo from target
     - 2.2.1. Threaten
     - 2.2.2. Blackmail
     - 2.2.3. Eavesdrop
       - 2.2.3.1.Listen to conversation (AND)
       - 2.2.3.2.Get target to state combo
     - 2.2.4. Bribe
3. Cut open safe
4. Install improperly

Figure 7. Textual view of the *Open Safe* attack tree.

Attack trees introduce a unique perspective on computer system security. The security analysis views the system from a computer hacker's perspective. This provides the advantage of a more holistic view of the software application in the production environment. This unique perspective often introduces levels of exposure that are initially not considered by the application architects (Schneier, 2000).

Five Step Methodology Overview

Schneier (2000) introduced the notion of attack tree as a chapter in his publication *Secrets & Lies*. The book was originally due for publication in 1998; however, Schneier (2000) choose not to have the book printed until two years later in 2000. Throughout 1998, Schneier participated in a team of four authors of the paper, *Towards A Secure System Engineering Methodology*, (Salter, Saydjari, Schneier, & Wallner, 1998). This paper offered a five step methodology for the creation of attack trees. The dates of these documents are interesting since attack trees were introduced to the general masses though Schneier's (2000) book, two years after the proposed methodology. Therefore, a foundation was set in place two years prior to the creation of the concept.

The methodology, based on an attack tree model provides a five step process for characterizing attacks and choosing countermeasures. Salter, Saydjari, Schneier, and Wallner (1998) provided a broad step overview:

1. Create attack trees for the systems.
2. Apply weights to the leaves.
3. Prune the tree so that only exploitable leaves remain.
4. Generate corresponding countermeasures.
5. Optimize countermeasure options.

Step 1

The first consists of the creation of the attack tree. This process includes identifying the attack goal, or root node, followed by the creation of the tree down to the lowest levels, or leaves. The creation of the attack trees is consistent with attack trees as described in the previous *attack tree* section of this document (Salter, Saydjari, Schneier, & Wallner, 1998).

Step 2

Once the attack tree has been built, qualitative values for risk, access, and cost are assigned to each node. As reflected in Table 2, values of High, Medium, and Low are assigned to each leaf (Salter, Saydjari, Schneier, & Wallner, 1998).

Table 2

Leaf Weight Assignments

	Weights		
Leaf	Risk	Access	Cost
Cut Safe Open	H	M	L

Step 3

The next step in the process includes the pruning of the attack tree based on input from step 2. The attack tree analysis allows one to discount certain paths in the tree based on the analyst's understanding of the attacker's known possibilities. For example, in the event the attack is considered to be waged by an attacker with a small budget of twenty thousand dollars, all nodes in the attack tree with a high cost, being in excess of one hundred thousand dollars, can be removed or pruned from the trees. Pruning the attack tree provides one with an opportunity to focus on the attack tree branches which pose the highest probability of attack (Salter, Saydjari, Schneier, & Wallner, 1998).

Step 4

In step four, countermeasures are applied to the remaining nodes. The tactics created or implemented to combat an attack against a threat is known as a countermeasure. For example, an effective countermeasure for the leaf node of *Obtain Combination* is to not write down the safe's combination. Hopefully, each leaf in the attack tree will have at least one countermeasure applied to the leaf. In certain cases, leaves may experience a no-known countermeasure (Salter, Saydjari, Schneier, & Wallner, 1998).

Step 5

The final step includes the ranking of the attack tree countermeasures utilizing five attributes as defined by Salter, Saydjari, Schneier, and Wallner (1998).

1. The cost to purchase and run countermeasure.
2. The ease of use for the countermeasure.
3. The countermeasure's compatibility with in-place technology, and the ability to interoperate with other communities of interest
4. The countermeasure's overhead on system resources
5. The countermeasure's time to market or availability.

Salter et al. (1998) believes that the ability to implement an algorithm or a computer based program to assist with step 5 in the ranking of the countermeasures is not feasible due to the coarse granularity of the weights.

Stratified Node Topology

Daley, Larson, and Dawkins (2002) introduced an extension to the existing attack tree paradigm by proposing the stratified node topology (SNT). The SNT partitions attack

tree nodes into three distinct layers offering an organizational taxonomy and a reduction in ambiguity. These three layers are (a) event-level, (b) state-level, and (c) top-level. Figure 7 displays the *Open Safe* attack tree implementing the SNT.

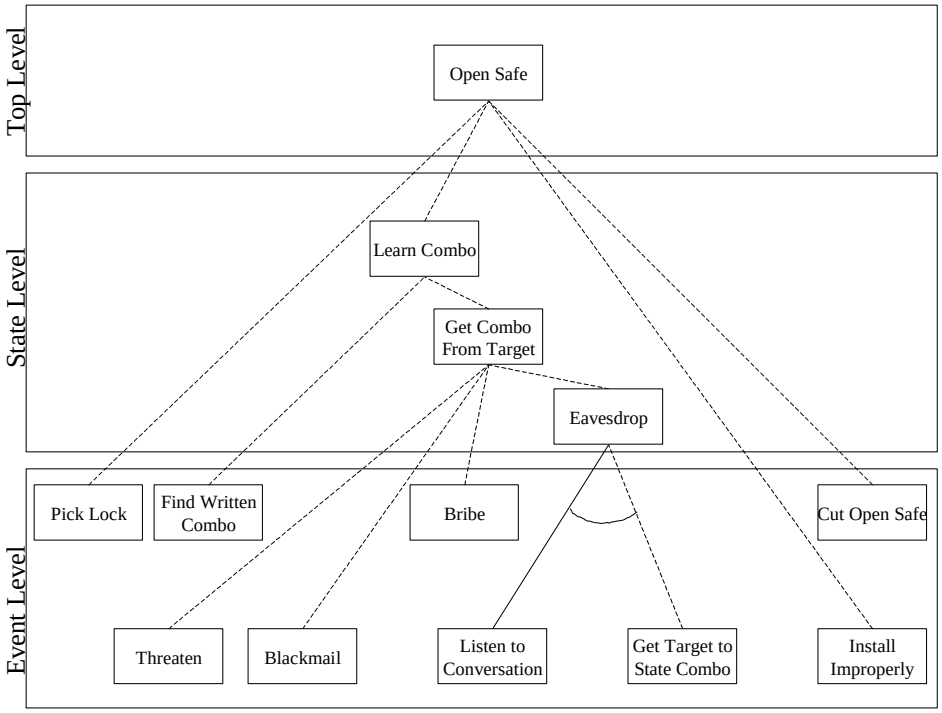

Figure 8. Stratified node topology.

Event-Level

The event-level layer of the SNT represents the action taken by the attacker to achieve her goal. In the open safe example, event-level nodes are the events that trigger a successful attack achieving the attack goal (Daley, Larson, & Dawkins, 2002).

State-Level

State-level nodes represent intermediate steps the attacker must take in order to achieve the goal. They provide the connection between the attack goal and the attack

implementation required to achieve a successful attack. For example, in *learn the combo* is a step one must achieve to open the safe. In and of itself, learning the combination to a safe is useless until one uses the knowledge to break into the safe (Daley, Larson, & Dawkins, 2002).

Top-Level

The event-level layer of the SNT directly correlates to the goal as identified by the root nodes of the attack tree paradigm. In our example, the Top-level node is *Open Safe*.

Attack Node Correlation

According to Daley, Larson, and Dawkins (2002) the stratified node topology offers an abstraction and categorization to the attack tree paradigm allowing one to place attack tree nodes in a layered taxonomy offering additional structure to the existing model. The SNT introduces the notion of attack node correlation by identifying the relations between nodes as implicit or explicit.

Implicit node links occur when the reaching of one node in the tree automatically triggers a secondary node. This then creates an implied relationship. As in the *Open Safe* attack tree example, if one is to *Eavesdrop* on a telephone line, then the listening to a conversation automatically begins. The relationship between the nodes is implied (Daley, Larson, & Dawkins, 2002).

Explicit links occur when the node relationship contains a decision point for the attacker. For instance, using the *Open Safe* attack tree example, if one is to learn the safe combination via the *Learn Combo* node, the attacker must decide which path down the tree to traverse causing an explicit node relationship.

Context Sensitive Nodes

According to Daley, Larson, and Dawkins (2002) context sensitive nodes provide a mechanism in which the attack tree is mapped to physical environment applying context. The attack tree nodes are assigned the value of the computer hardware and software. For example, in the event that two people had knowledge of the safe's combination, the attack tree is extended to include two *Get Combo* nodes, one for the first person containing the combination *Person1*, and a second for the second person containing the combination, *Person2,* as reflected in Figure 8.

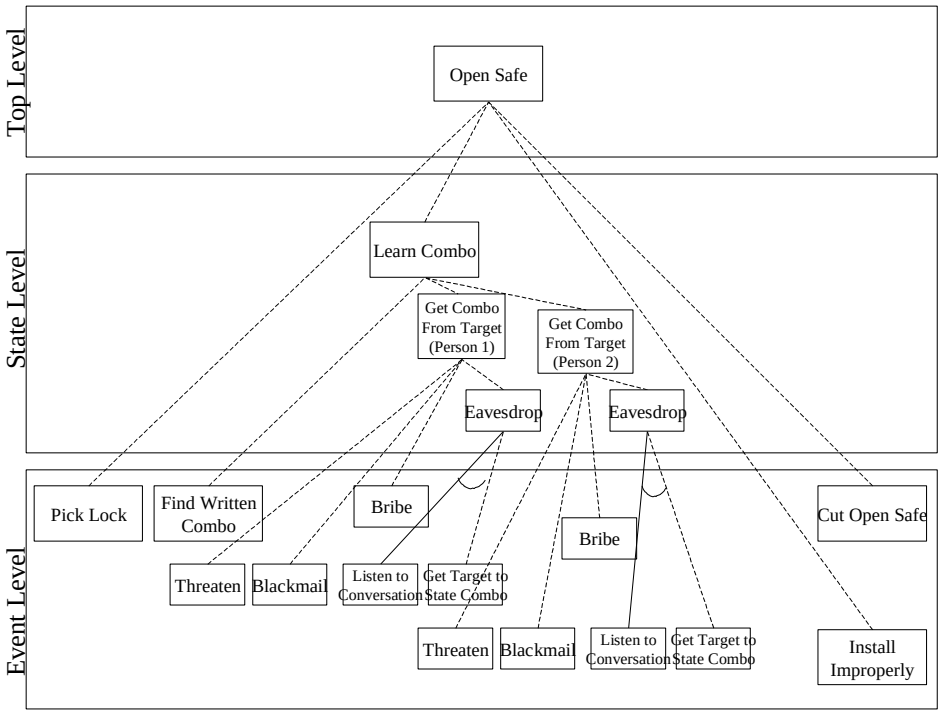

Figure 9. SNT with attack node correlation and context sensitive nodes.

The stratified node topology provides a method for classifying multi-stage network attacks. The SNT approach maps physical devices with attack tree nodes

expanding on attack tree reuse capabilities describing a method for correlating attacks against an enterprise (Daley, Larson, & Dawkins, 2002).

Cause-and-Effect Analysis

The cause-and-effect model is built upon the notion that specific causes use specific mechanisms producing specific effects (Cohen et al., 1999). Thus, since a specific relationship exists among the particular causes, mechanisms, and effects, protective mechanisms can be implemented to reduce the exposed risk at any of the three levels. The cause-and-effect model is reflected in Figure 9.

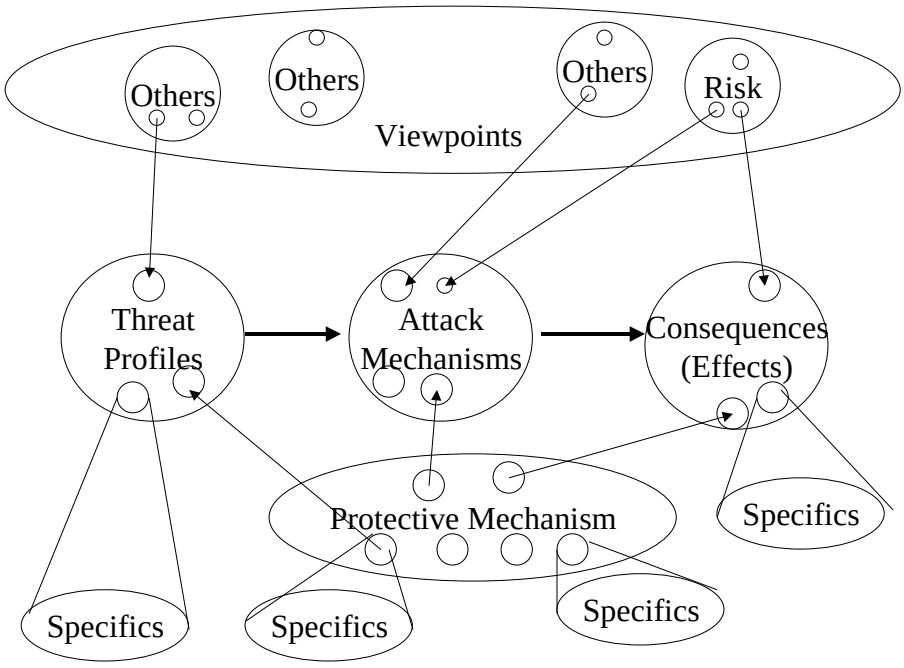

Figure 10. Cause-and-effect model (Cohen et al., 1999).

To align the taxonomy used within the cause and effect model as applied to computer security and risk assessment, a cause is also known as a threat, the mechanism is also known as the attack, and the effect is also known as the defenses or countermeasures. The schema used to collect the details of the specific Causes, Mechanisms, and Effects are of a simple template consisting of an actor, mechanism, and consequence. For example, Cohen et al. (1999) described Threat 10, in which the actors are hackers as:

> **Threat 10**: *hackers*
> People who enjoy using computers and exploring the information infrastructure and systems connected to it.
>
> Complexity: While not generally malicious, these people tend to gather and exploit tools that open holes to other attackers. They also sometimes make mistakes or become afraid and feel they have to cover their tracks, thus causing incidental harm. (p. 13)

The cause-and-effect model describes different types of actors that may cause information systems failure (threats), the mechanism by which systems are caused to fail (attacks), and mechanisms, which may mitigate risk (defenses). Model output produces short concise definitions for cause, mechanism, and effects (Cohen et al., 1998).

The cause-and-effect model requires a comprehensive understanding of the system. The analyst must first identify the causes, introduce the mechanics, and then capture the anticipated effect. This mode does lend itself to pattern development and reuse. Causes and effects can be articulated into cause patterns and effect patterns allowing multiple causes producing the same effect pattern to be potentially reused. The model is also best suited for a more sequential failure process (Cohen et al., 1998).

Cause-Consequence Analysis

RISO Laboratories in Denmark developed the cause-consequence analysis (CCA) methodology in the early 1970s for use in the nuclear industries (Aven, 1992). CCA is a blend of fault tree analysis and event tree analysis. The process begins with a critical event, determining the causes of the event, and the consequences of the event. The event causes are captured using deductive logic and fault trees. The event consequences are recorded using inductive logic and event trees. The main purpose of CCA is to identify a chain of events, which result in undesirable consequences as opposed to a single event (Andrews & Moss, 2002; Herzog & Shahmehri, 2001). The method often results in a complex process, a mixture of event trees and fault trees, such as the cause-consequence analysis diagram reflected in Figure 10.

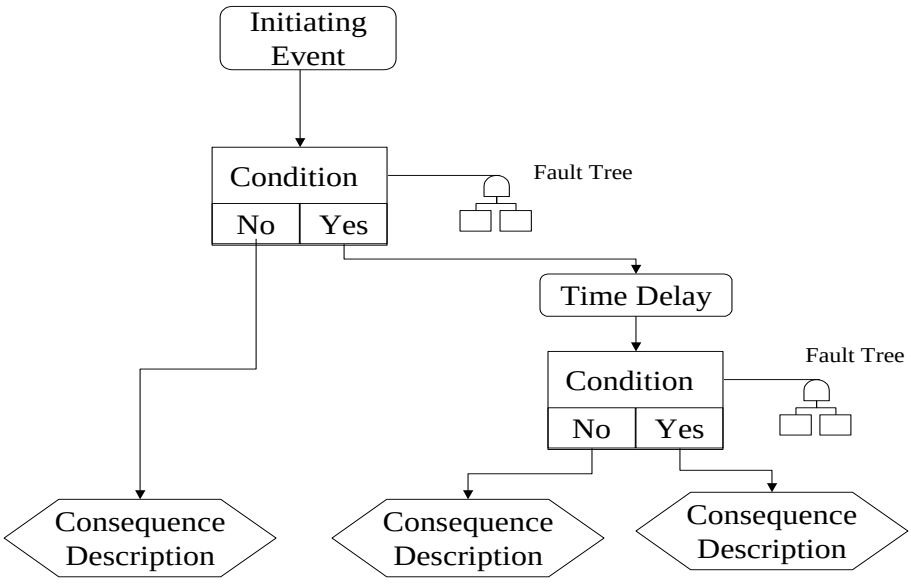

Figure 11. A typical cause-consequence analysis.

Cause-Consequence Analysis allows the analyst to use top-down and bottom-up searches to provide different perspectives. CCA produces complex diagrams not easily understood. Analysts require an extensive understanding of CCA prior to being able to achieve analytical benefits (Herzog & Shahmehri, 2001).

CORAS

CORAS is a European project focused on modeling the risk assessment process by means of model-based risk assessment. The project began in 2001 and is to be completed in 2003. The CORAS consortium consists of three commercial companies, seven research firms, and one university (Stolen et al., 2002).

CORAS has created a framework building upon current methodologies creating a four-pillared approach reflected in Figure 11.

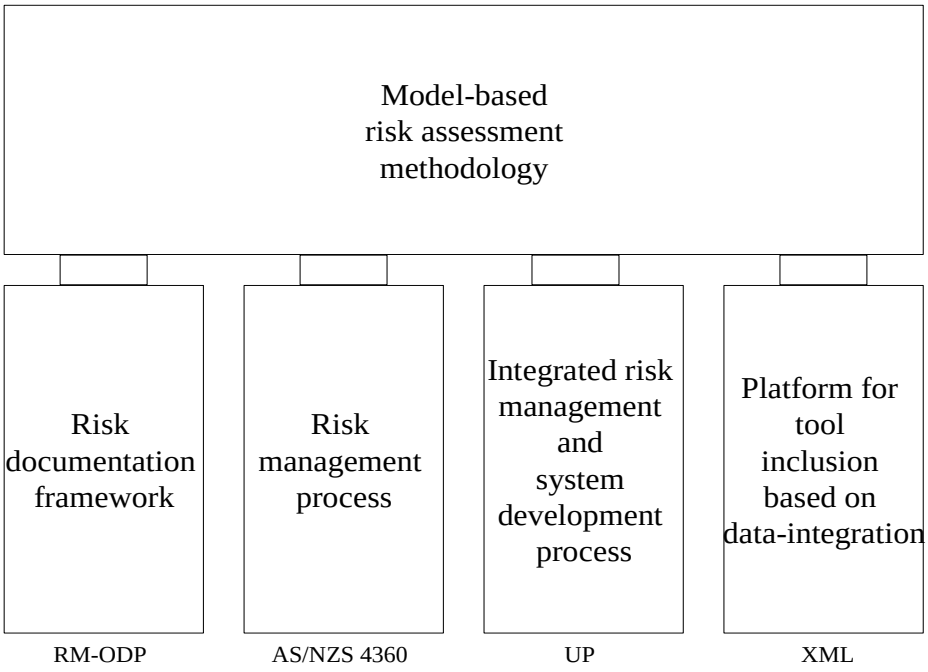

Figure 12. The CORAS framework (Stolen, Braber, & Dimitrakos, 2002).

Risk Document Framework

The CORAS risk document framework pillar is built upon the reference model for open distributed processing (RM-ODP) (AS/NZS 4360, 1999). RM-ODP provides an existing taxonomy of structured terminology, an existing classification process that utilizes conformance modeling and a distribution model.

CORAS extended the RM-OPD model taxonomy to include risk assessment and security. The model has also been extended to incorporate additional support for conformance checking (Stolen et al., 2002). Reusable components are also achieved with the risk document framework such as specification fragments, patterns, and templates used to capture generic aspects.

Risk Management Process

The CORAS risk management process is built by consolidating multiple standards including AS/NZS 4360 (1999), *Code of Practice for Information Security Management* (ISO/IEC 17799, 2000), *Guidelines for the management of IT Security (ISO/IEC, 2001)*, and the *Functional Safety of Electrical/Electronic/Programmable Electronic Safety Related Systems* (IEC, 2000). CORAS utilizes the Universal Modeling Language (UML) (Jacobson et al., 2000) toolkit to model the above-mentioned standards.

Integrated Risk Management and System Development Process

The CORAS integrated risk management and system development process is based upon the Unified Process (UP) (Jacobson et al., 1999) incorporating a 5-step approach with an iterative process paralleling the software development life cycle approach. CORAS model includes assessing security throughout the development of software

applications, allowing for iterations that address additional concerns as they are introduced; therefore the management of risk is included in all phases of development.

Platform for Tool Inclusion Based on Data Integration

CORAS contains a data repository utilizing an industry standard for data formatting known as extensible markup language (XML) (Bray et al., 2000). Usage of XML increases the probability that CORAS data will be integrated with other application tools. The CORAS repository may also ingest data from other computer automated software engineering (CASE) tools. CORAS is a model-based risk assessment tool which incorporates many of the industry's leading methodology for risk assessment using an open standard database structure of XML.

CORAS has been and is being developed by practitioners and academics. This combined approach offers a unique comprehensive perspective (CORAS, 2003). The methodology provides a framework and the use of modeling tools. There is a lack of, and difficulty, in building CORAS experience, domain and context. The methodology does not lend itself to easy adaptation and tailoring forcing a restricted implementation that constrains users (Gan & Scharf, 2003)

Event Tree Analysis

The Nuclear Regulatory Commission created event tree analysis (ETA) in the 1960s for the Reactor Safety Study, also known as the WASH 1400 study (Andrews & Dunnett 1997; Fullwood & Hall, 1988; Rasmussen, 1975). Using inductive, or forward logic, event trees consider the failure of a single component and explore the ramifications which can occur from this failure (Henley & Kumamoto, 1992; Herzog and Shahmehri, 2001; Fullwood & Hall, 1988; Sutton, 1992).

Quantification of event tree is created using a mapping of the frequency of occurrence and the probability of outcome. Event trees are recommended for complex systems where the task of creating a fault tree for the failure of an "accidental release of toxic gas" is needed. The process of identifying all the values for the failure of one event produces a more effective manageable result (Andrews & Dunnett, 1997).

Event trees represent a hierarchical left-right tree structure in which the base event is the tree base. This relationship is continued down the tree creating additional branches until all known outcomes have been captured (Andrews & Dunnett, 1997). A simple event tree capturing the leakage of gas from a system with a single value producing one of three outcomes is reflected in Figure 12.

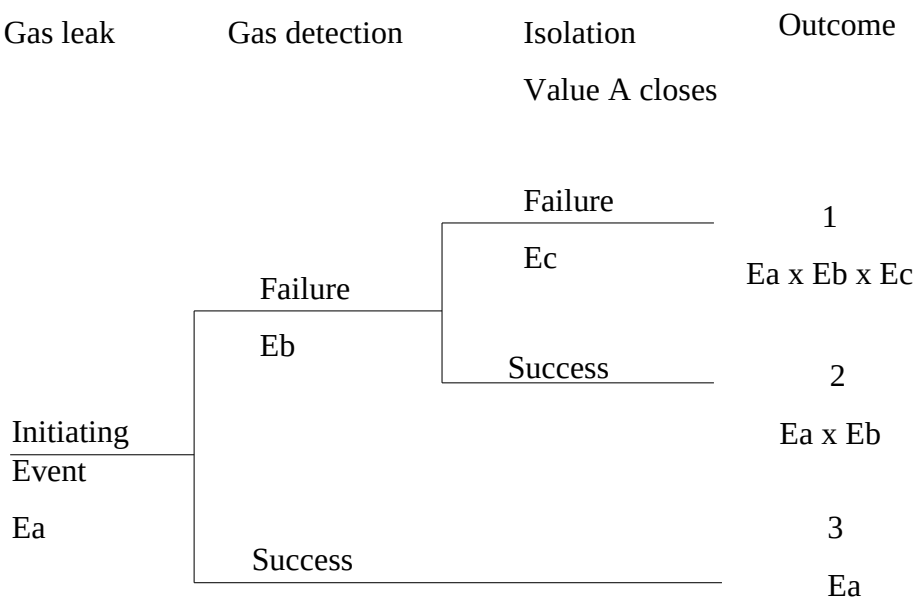

Figure 13. Event tree depicting a gas leak.

Event trees allow a component by component approach to be taken. This approach offers a systematic functional view perspective, and is not a comprehensive system analysis (Herzog & Shahmehri, 2001). Therefore, when performing event tree analysis, another method is also included to complement the use of event trees. Often Fault Tree analysis is combined with Event tree analysis creating a more complete risk assessment solution (White, 1995).

Failure Modes and Effects Analysis

Failure modes and effects analysis (FMEA) was developed in the 1950s by reliability engineers to assist with analysis of military systems. FMEA technique evaluates an entire system by reviewing each mode, or item, in the system (Andrews & Moss, 2002; Aven, 1992; Henley & Kumamoto, 1992; Fullwood & Hall, 1988). System experts, using historical information from similar items, perform analysis on how each component or subsystem might fail to perform its intended function. Each of these potential failures is assigned a numeric value in three separate categories. The first category identifies the probability that the failure will occur. The mode is also ranked with the severity that will occur if the worst possible outcome occurs from the failure. Then, the mode is ranked as to the probability that the failure will be detected and corrected by the system. For example, Table 3 represents a failure mode and effect analysis applied to a web server.

Table 3

FMEA of Web Server

	Analysis					
System	Component	Failure Mode	Effect	Criticality	Mitigation	Probability
Web Server	Software	Locked up	Web site is unreachable	No customer assess	Automate server reboot process	1×10^{-3}/hr
	Hardware	Sudden crash	Web site is unreachable	No customer assess	Page technician	1×10^{-3}/demand
	Concurrent users exceeded	Slow processing	Customer experience long waits	Potential loss of customers	Expand environment	1×10^{-10}/demand

During FMEA analysis numeric values are assigned to failure indicators of occurrence, severity, and detection for each component. These values are then multiplied, producing an overall risk factor for each component potential failure mode. The overall risk factor is then used to identify existing system design that is most likely to produce reliability, quality, or safety problems (Elliott, 1998).

Failure modes and effects critical analysis (FMECA) extends the FMEA methodology focusing on the critical element of measure (Sutton, 1992). FMECA is noted here due to the extensive overlay with FMEA. It also contains the critical element of measure; therefore, FMECA demands a more intense focus of this single element.

The FMEA process begins with a brain-storming session in which analysts focus on each component of a system identifying possible risks and a failure scenario. This requires knowledgeable systems engineers or analysts to participate in the brain-storming process. The FMEA model allows for qualitative exploration into system components, by viewing single components in isolation of the complete system.

According to White (1995, p. 36) “FMEA is a reductionist procedure which fails to identify interactive combinations of equipment failures or common cause failures.”

Fault Tree Analysis

H. A. Watson, of Bell Laboratories, funded by the United States Air Force to study the Minuteman Launch Control System or Minuteman Intercontinental Ballistic Missile Launch Control System, created fault tree analysis (FTA), also known as failure tree analysis, in 1961 (Ericson, 1999; Vincoli, 1994). Fault tree analysis, using deductive logic, is a method in which a particular system failure is expressed in terms of component failure mode and operator actions. FTA uses a visual hierarchical tree representation capturing the failure or fault as the base of the tree, expanding into branches reflecting root causes and fault path (Elliott, 1998; Ericson, 1999; Harrington & Anderson, 1999, pp. 131-132). A simple fault tree, used to identify the cause of a fire, is reflected below in Figure 13.

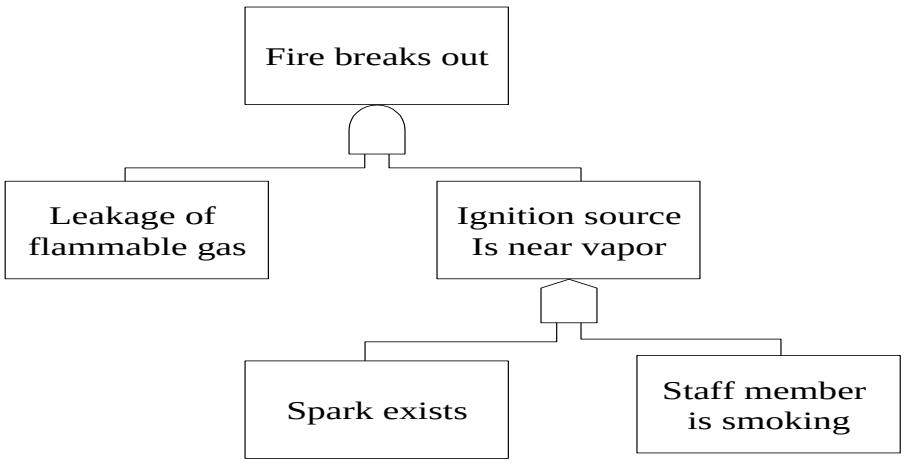

Figure 14. Fault tree depicting the event “fire breaks out.”

Fault tree analysis incorporates the use of AND, OR, and NOT gates, gates being the decomposition of one event into lower casual events. The above-mentioned gates allow Boolean algebraic analysis to be performed with FTA since the gates AND, OR, and NOT correspond to union, intersection, and complementation (Ericson, 1999).

FTA is based on reliability theory, Boolean algebra, and probability theory. A simple set of rules and symbols provide mechanism for analyzing complex systems, allowing the analyst to incorporate quantitative and/or qualitative risk analysis (Elliott, 1998; Ericson, 1999; Harrington & Anderson, 1999, pp. 131-132; Vincoli, 1994).

Fault tree analysis assumes failures shall occur as a sequential process that maps to a hierarchical tree. This static approach lacks the dynamic aspects of dynamic failures (White, 1995).

Hazards and Operability Studies

Hazard and operability studies (HAZOP) were developed by Imperial Chemical Industries Limited in the early 1970s (Sutton, 1992; Andrews & Moss, 2002). HAZOP's main concern in the chemical industry focuses on flow and causes of flow state, such as too little flow, too much flow, and partial or no flow between system components. The flow analysis is mapped to the information technology world via the flow of data (McDermid et al., 1995; Storey, 1996). The HAZOP methodology is reflected in Figure 14.

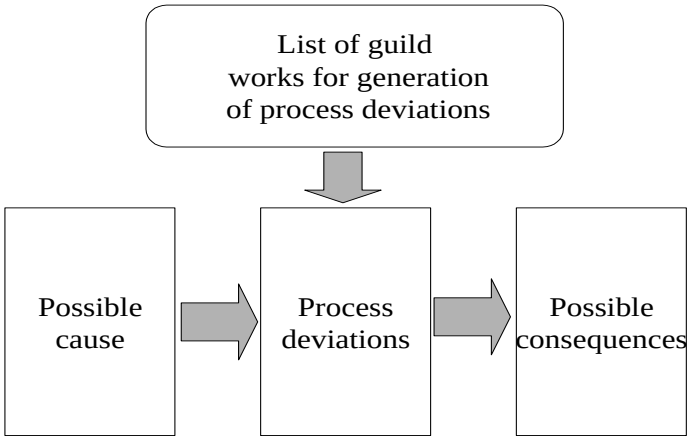

Figure 15. Hazard and operability studies methodology.

The HAZOP methodology calls for analysis to be performed by a multidisciplinary team, headed by an experienced HAZOP study leader, which uses a model of the proposed system and a combination of guidewords and process parameters to guide the study (Raafat, 2002). The team creates scenarios using guidewords to prompt flow deviation resulting in the identification of hazards and operational problems, as listed in Table 4.

Table 4
A list of HAZOP guide words

Guide work	Meaning
NO or NOT	The complete negation of these intentions (e.g., NO flow)
MORE LESS	Quantitative increase or decrease (e.g., high pressure or low pressure)
AS WELL AS	A qualitative increase (e.g., impurity)
PART OF	A qualitative decrease (e.g., only one of two components in a mixture)
REVERSE	The logical opposite of the intention (e.g., backflow)
OTHER THAN	Complete substitution (e.g., flow of wrong material)

Upon hazard and operation problem identification, the team is able to identify consequences and measures to reduce the frequency of the hazard (Smith & Harrison, 2003). The HAZOP process utilizes brainstorming techniques and is often time consuming and produces excessive output. Therefore, only those scenarios, also known as deviations, which produce a hazard or operational problem, are recorded (Raafat, 2002).

HAZOP is also a process fed with brain storming output, thereby requiring system knowledge analysts. It considers components independently of the system, lacking component interdependence and holistic system perspectives. The most serious shortcoming of HAZOP is the failure to foresee human errors (White, 1995).

Markov Model

The models discussed up to this point evaluate static states in a sequential flow. That is, an event occurs followed by either a mode failure or a component is successful. In dynamic systems, a component may fail, but then a backup component assumes functional responsibility for the failed unit. In the event the second unit failed, a state would occur based on the failure of multiple components. The second component could only fail if the first one did also. In addition, a component may exist in more than two states, such as success or failure, open or close, and may be time-dependent of each other. For analysis of type-dependent, dynamic systems, the Markov Modeling technique is used. This model produces a chain in which transition between states occurs at a discrete moment in time (Andrews & Moss, 1992, 2002; Rajgopal & Mazumdar, 2002).

In order to incorporate the Markov model, the system must be identifiable, and lack memory. It must lack memory in the sense that the historical state of the component or all events leading up to the current state are not considered or relevant, only the current

state of the component is so considered. The component must also lack independence, that is, the component is not considered in isolation. Once the above-mentioned conditions have been met, meeting the Markov model criteria, the analyst may begin to draw the state transition diagram (Andrews & Moss, 1992; 2002).

A simple example of Markov Analysis, reflected in Figure 15, considers a component in a working state of time t = 0. The component may transition from the working state of 1, to a failed state of 2, occurring at a constant rate of λ. The component repair process is represented by a constant rate of *v*.

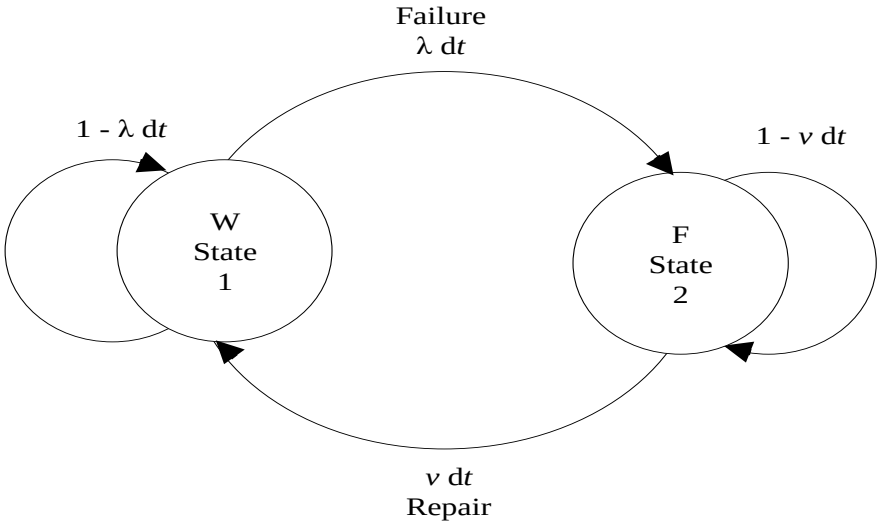

Figure 16. Markov model depicting a repairable component state diagram (Aven, 1992).

Markov model effectively addresses dynamic component-to-component scenarios addressing one of the shortcomings of the risk assessment techniques reviewed earlier in this paper; however, Markov requires the explicit identification of all known states and the transition between these states. The task of identifying the entire state set of a system prior to scenario development is difficult (Aven, 1992; Andrews & Moss, 1992, 2002).

Literature Review Summary

Many techniques used to assess risk have been identified by this literature review. The ten techniques explored by this paper can be grouped into four categories: (a) qualitative, (b) tree-based, (c) dynamic, and (d) hybrid.

The qualitative methodologies include HAZOP and FMEA which focus specifically on component failures such as hardware or software. The qualitative methodologies do not account for human failures and offer no insights into software risk assessment from an application vulnerability, or corporate asset perspective. These methodologies are used extensively in nuclear and chemical industries (Huang, Shi, & Mak, 1999; Keong, 1997; Lawrence, 1995; Smith & Harrison, 2003; Sutton, 1992).

Fault tree analysis (Aven, 1992, 2002; Elliott, 1998; Ericson, 1999), event tree analysis (Herzog & Shahmehri, 2001; Fullwood & Hall, 1988; Sutton, 1992), attack tree analysis (Schneier, 1999, 2000), cause-consequence analysis (Cohen et al., 1998) are methodologies categorized as tree methodologies. Tree methodologies offer static logical modeling of systems and processes providing a vehicle for quantification as values, weights, and other numeric values are assigned to tree branches, nodes, and leaves. Tree analysis views events or processes in a sequential systematic flow requiring decision points at each fork in the model and creates a non-dynamic view of possibilities for failure.

The Markov model's use of dynamic analysis addresses the shortcomings of the qualitative and tree-based approaches; however, extensive knowledge of all system states and state transitions are required prior to building scenarios for analysis. Extensive data gathering and model construction is required prior to performing analysis (Andrews & Moss, 1992, 2002).

Hybrid methodologies include the utilization of many of the diverse methodologies into a single framework. For example, the CORAS framework incorporates HAZOP, FTA, FMECA, Markov, and CCTA (Dimitrakos et al., 2000) incorporating the creation of a tool, completed in 2003, to assist with analysis. The Analytic Hierarchy Process introduced in the early 1980s was too difficult for analysts to use until a software tool was created to simplify the process. Hybrid methodologies offer additional advantages leveraging diverse methodologies that complement each other; however, comprehensive experience is often lacking (White, 1995).

Most computer security risk assessment models utilized are borrowed from other industries (Andrews & Moss, 1992, 2002; Aven, 1992; Durrett, 2003; Elliott, 1998; Fullwood & Hall, 1988; Gan & Scharf, 2003; Herzog and Shahmehri, 2001; Rasmussen, 1975; Smith & Harrison, 2003; Sutton, 1992; Vincoli, 1994). According to the literature review the nuclear power industry offers the most guidance for computer security and risk assessment. In reviewing the literature, the models most used in assessing computer software risks include HAZOP, FMEA, ETA, FTA, CCA, and the introduction of attack trees (AT).

All risk assessment models, except attack tree analysis, view the systems, process, and components from a failure perspective. Risk assessments require more than a focus on the system; a focus on penetration points from people or entities wishing to disrupt, damage, invade, or steal information from the system is also necessary. Attack trees offer an additional perspective allowing system or process vulnerabilities to be viewed from an intruder’s perspective.

CHAPTER 3: RESEARCH DESIGN

The concept of the attack tree has been introduced into the information systems industry within the last 5 years (Schneier, 2000) as a means of assisting information systems with risk assessment and threat analysis. The literature review revealed that attack trees appear to lack the academic support in quantitative and qualitative forms (Ellison & Moore, 2001, 2003; Salter, Saydjari, Schneier, & Wallner, 1998; Schneier, 1999, 2000). The problem is that attack trees have been introduced into the practitioner arena yet lack academic evaluation and substantiation. This problem has created the current gap between applied and theoretical knowledge of the methodology. This research partially bridged the gap between applied and theoretical notions by testing an attack tree model evaluating costing and probability analysis using a structured query language simulation model.

The completed research was an evaluation research that extends and tests an attack tree model (Singleton & Straits, 1999). The one-group pretest-posttest design was proposed. The attack tree model extension included the introduction of probability analysis to assist with attack tree analysis. The proposed model also introduced a means of performing cost analysis, providing information systems management with the information needed to assist in the decision as to where funds are best invested to maximize the security return, and to provide the most protection to information assets.

The methods included the creation of an attack tree model with the inception and implementation of a computer program, developed as part of this research effort, to automate the algorithms required when performing the analysis. This model was implemented by creating an attack tree using the computer program to perform analysis on a generic Internet host domain, also known as a corporate web site, such as Walden University's public home page, http://www.waldenu.edu. This approach offered a

foundation Internet attack tree that can be used by organizations since the Internet requires certain consistency among all host domain servers (home page), such as domain name, domain address, and web servers.

The computer program created for this dissertation evaluated the effectiveness of the attack tree extensions and costing model providing information systems personnel with additional information to be used in the cost-benefit model. The computer program incorporated the use of structured query language, macros, and programming extensions as needed to assist with the evaluation.

When using the computer program, one was able to make a decision as to the effectiveness of the computer program, attack trees, and the extended attack tree paradigm as introduced by this research. This approach also offered a conduit between basic research and applied research as practitioners and academics evaluated the proposed model in the field.

A second approach considered was a case study in which a single organization would have been examined incorporating a mixed model of quantitative and qualitative research. Implementing a case study would allow the research effort to focus on the information security department of a single organization, and not a purposeful sample of computer security and information systems managers (Singleton & Straits, 1999).

The case study approach would entail the researcher's need to take a week of vacation in order to spend time with the case study organization. During this week's time, a pilot project of the security risk assessment model proposed as part of dissertation would have been implemented. The project would then have begun with a kickoff meeting in which a pretest survey were conducted utilizing the instrument, then the next few days would have been spent using the model created to assist risk exposure assessment within the organization. On the last day of the week, after the pilot model had

been incorporated into the organization, a posttest survey would have been supplied to the participants following the one-group pretest-posttest design.

The benefit to the studied organization would have been the results of the week of analysis of the corporations' systems, thereby providing free consultation. There is, however, a risk that the model proposed would not be of value. The deliverable from the project would have been a report to the organization's security staff re: identified penetration points of the systems studied during the week. The deliverable would have also included cost estimates for securing those penetration points and the value of the asset being protected in order to support executive decision making processes and cost justifications (Singleton & Straits, 1999).

The case study methodology was not selected due to the lack of external validity. Based on the ability of evaluation research to reach a greater mixture of organizations representing diverse corporate structures, the implications of the data have a greater impact and are thereby believed to be of greater use to organizations, governmental agencies, and universities. The case study did, however, possess a stronger potential of control over internal validity.

Target Sample

The attack tree paradigm, implemented in a computer program, was submitted to a target population of 56 computer security experts and leading academic authorities on attack trees. Since attack trees are relatively new, the target population was identified by selecting all of the authorities from attack tree publications in peer-reviewed journals (Daley, Larson, & Dawkins, 2002; Moore, Ellison, & Linger, 2001; Salter, Saydjari, Schneier, & Wallner, 1998; Tidwell, Larson, Fitch, & Hale, 2001), theses (Moberg, 2001;

Sawma, 2002; Thunstrom & Ahs, 2003), published books (Bauer, 2002; Schneier, 2000), and individuals with an industry reputation (Cohen, 2003; Porter, personal communications, 2003) as leading computer security experts based on their experience and credentials. Since the target population is such a manageable size (n=56), the entire population was approached to participate in this research. The sample population evaluated the methods proposed during this dissertation.

Sampling Procedure

A nonprobability purposive sample was incorporated based on the need to identify important sources in the population (Singleton & Straits, 1999, p. 158) due to the relative newness of attack trees. The researcher compiled a list of professors, authors, and computer security experts of whom the purposeful sample was assembled. The list of publications and affiliations is found in Table 5. The names of the individuals are considered protected data and are therefore not reflected in the population table. A detailed list of the target population, including names and contact information has not been included in this document to protect the confidentiality of the participants.

Table 5

Population

	Description	
Name	Affiliation	Publication
Protected	University of Tulsa	A Structural Framework for Modeling Multi-Stage Network Attacks
Protected	University of Tulsa	A Structural Framework for Modeling Multi-Stage Network Attacks
Protected	University of Tulsa	A Structural Framework for Modeling Multi-Stage Network Attacks
Protected	Creator of Attack Trees	Secrets & Lies, Digital Security in a Networked World
Protected	University of Maryland	Secrets & Lies, Digital Security in a Networked World (referenced)
Protected	Carnegie Mellon University	Attack Modeling for Information Security and Survivability
Protected	Carnegie Mellon University	Attack Modeling for Information Security and Survivability
Protected	Carnegie Mellon University	Attack Modeling for Information Security and Survivability
Protected	National Security Agency	Toward A Secure System Engineering Methodology
Protected	DARPA	Toward A Secure System Engineering Methodology
Protected	National Security Agency	Toward A Secure System Engineering Methodology
Protected	Avaya	Personal Communications
Protected	University of Karlsruhe	Development Process Of Secure Systems
Protected	University of Ottawa	E-Commerce Security, A New Methodology for Deriving Effective Countermeasures Design Models
Protected	Chalmers University of Technology	Security analysis of an information system using an attack tree-based methodology
Protected	George Mason University	Attack Tree Modeling using Rational Rose
Protected	University of Washington	Looking at Vulnerabilities

(table continues)

Protected	US Military Academy	Modeling Internet Attacks
Protected	University of Tulsa	Modeling Internet Attacks
Protected	US Military Academy	Modeling Internet Attacks
Protected	University of Tulsa	Modeling Internet Attacks
Protected	Jet Propulsion Laboratory	ICSE 2003 Workshop on Software Engineering for High Assurance Systems: Synergies between Process, Product, and Profiling
Protected	California Institute of Technology	ICSE 2003 Workshop on Software Engineering for High Assurance Systems: Synergies between Process, Product, and Profiling
Protected	Center for High Assurance Computer Systems	ICSE 2003 Workshop on Software Engineering for High Assurance Systems: Synergies between Process, Product, and Profiling
Protected	Carnegie Mellon University	ICSE 2003 Workshop on Software Engineering for High Assurance Systems: Synergies between Process, Product, and Profiling
Protected	Chalmers University of Technology	Security analysis of a system connected to a future Network Based Defense
Protected	Chalmers University of Technology	Security analysis of a system connected to a future Network Based Defense
Protected	West Virginia University	Mobile Agent Based Attack Resistant Architecture for Distributed Intrusion Detection System
Protected	University of Wisconsin	Survivability Analysis of Networked Systems
Protected	Texas Technical University	Threat Modeling and Risk Management
Protected	O'Reilly Books	Building Secure Servers with Linux
Protected	Burton Group	Risk Management: Concepts and Frameworks

Sample

The population contained the entire target population of 56, since Singleton and Straits (1999) state "thirty cases generally are regarded as minimally adequate for statistical data analysis" (p. 168).

Instruments

Surveys

Surveys were used as the data collection instrument for this research. Rea and Parker (1997) suggested several advantages to the survey process. The major advantages of surveys are the cost associated with collecting the data and the time required for the data collection process. Surveys often require less time to complete than other data collection types, such as face-to-face or phone interviews. Surveys are also often less expensive than other types of data gathering as the survey may be administrated via postal mail, electronic-mail, or the telephone, without an in-person presence.

Mertens (1998) warned, "After an exhaustive search of the literature, you may determine that no existing instrument will measure exactly the construct in which you are interested. Thus, you will find it necessary to develop your own data collection instrument" (p. 313). The above-mentioned caution was experienced by this research effort to the best of the researcher's knowledge, the literature review revealed the absence of a construct that measured any aspect of attack trees. Mertens (1998) suggested the following model when creating a unique research instrument (a) define the objective of the instrument, (b) identify the intended respondents, (c) review existing measures, (d) develop an item pool, (e) prepare and pilot test the prototype, and f) conduct an item analysis and revise the instrument as necessary.

Sheatsley (1983) suggested that survey design is more an art than a science. He then concluded that survey questions should be short, ideally fewer than 25 words each, and that the entire survey be brief. Rea and Parker (1997) added that the responder should be able to complete a survey in less than 15 minutes.

Participant Pre- and Post-Assessment Survey

Pre- and post-assessment survey instruments were designed to measure the effectiveness of the variables considered (Singleton & Straits, 1999), that is, use of attack trees, cost-benefit analysis, probability, and structured query language computer simulation model. Since suitable instruments could not be identified for measuring those variables, the researcher developed survey instruments based on information derived from relevant literature. A computer science expert with knowledge of attack trees and an experienced instrument designer reviewed both survey instruments. The pre-assessment survey instrument was pilot-tested on a group of 5 computer security experts and information systems managers.

The pre- and post-assessment survey consisted of a total of 20 close-ended yes/no questions in four functional areas (attack trees, cost-benefit analysis, probability, and using a computer program built leveraging the SQL programming environment). Putt and Springer (1989) suggest that close-ended questions provide procedural advantages since these questions require the respondents to check or circle their preferences. Questions with a “yes” or “no” choice are easier and quicker for the respondents to answer as they do not require a written answer. Additional benefits include an increased response rate, reduced costs to the researcher, reduced time in data analysis, and a reduced time requirement for the respondents to complete the survey (Putt & Springer, 1989, pp. 209-211). The categorization of the pre-assessment survey questions are reflected below.

Table 6
Pre-assessment survey questions categorization

Question Number	Functional Area			
	Familiarity Attack Tree	Costing	Probability	SQL Program
1 – 8	Yes/No			
9 – 12		Yes/No		
13 – 16			Yes/No	
17 – 20				Yes/No

According to Alreck and Settle (1985) responses scored on a dichotomous scale such as "yes" and "no" questions are administered more easily and completed in shorter time frames by the respondents. Data analysis requires less effort for the researcher. The disadvantages are erroneous data that may be collected if the respondents cannot decide how to answer the question and the amount of information that can be gathered is limited (pp. 198-202) . This researcher believes these limitations have been addressed by the crafting of survey questions that are clear and complete. The survey questions accurately ask focused information and solicit responses directly related to the question in which only one response can be singled out. According to Alreck and Settle (1985), only when one response can be singled out, may "yes" or "no" questions be used.

In addition to the 20 closed questions, the post-assessment survey also contains fifteen Likert type responses using the Likert scaling with ranges from "disagree" to "agree" on a 1 to 5 continuum in which 1=disagree and 5=agree. Singleton and Straits (1999) suggested that Likert scales are a common way of measuring attitudes.

Finally, the post-assessment survey contains an additional six open-ended questions. Singleton and Straits (1999) suggested that open-ended questions offer a respondent the greatest advantage in the freedom granted when answering the question. However, the freedom afforded with open-ended questions introduces a level of complexity for the researcher since he is responsible for coding the questions. Coding such material is a more "time-consuming and costly process that invariably results in some degree of error" (Singleton & Straits, 1999, p. 281). Open-ended questions also require the researcher to be skilled in "recognizing ambiguities of response and in probing and drawing respondents out…to make sure they give codable answers" (Sudman & Bradburn, 1982, p. 151). The categorization of the pre-assessment survey questions are reflected in Table 7.

Table 7
Post-assessment survey questions categorization.

Question Number	Functional Area			
	Familiarity Attack Tree	Costing	Probability	SQL Program
1 - 8	Yes/No			
9 - 12		Yes/No		
13 - 17		1 to 5 scale		
18 - 21			Yes/No	
22 – 26			1 to 5 scale	
27 - 30				Yes/No
31 – 35				1 to 5 scale
36		Open-ended		
37			Open-ended	
38				Open-ended
39 - 41	Open-ended questions covering all functional areas.			

Several of the survey questions were reverse-scored to preclude response sets. According to Kiplinger (1973) a response set is “a general tendency to agree or disagree with questionnaire items, regardless of their content” (p. 43). These tendencies are for some personalities to strongly agree or disagree with all questions. To counteract a response set, the survey includes six reverse-scored, also known as stated in the negative. Bradbury (1983) suggests “trying to develop positive and negative statements with which to measure attitudinal dimensions; by using balanced items, survey researchers tried to minimize the impact of such response sets” (p. 316).

The pre- assessment survey is listed in Appendix A. The post-assessment survey instrument is listed in Appendix B.

Survey Instrument Validity and Reliability

Pre- and post-assessment survey instrument validity was accomplished by submission to review by an attack tree subject matter expert and an experienced instrument developer. The reliability of the survey instruments was confirmed by incorporating a small pilot program submitting the instrument to five security experts.

Data Collection Procedures

The data collection procedures began with each subject receiving electronic mail that included a cover letter describing the research and requesting their assistance in the study. Also included in the cover letter was a universal resource locator (URL) address, also known as an Internet address, in which the respondents were able to download the computer program and the survey instrument. To initiate the download process the respondents would complete the pre-survey and enter a compliance of consent. Respondents who did not download the survey within one week's time received follow-up email requesting participation. Respondents who did not download the survey and computer program within 2 weeks received a phone call. A total of two phone calls where made within a 6 week time period. Respondents also received a toll free, 800 number to return the researcher's phone call at their convenience.

All cover letters were electronically mailed on the same date, to monitor the return of the completed survey within a 2-week timeframe. Surveys were captured using the survey web site SurveyMonkey.com (2005). Data was returned to the researcher by

means of an electronic download. The respondents who did not return the survey within the 2-week time period, after the initial download, received follow-up electronic mail requesting the completed survey. Respondents who did not return the completed survey within 2 weeks received a phone call. A total of two phone calls were made within a 6-week time period. They also received a toll free, 800 number to return the researcher's phone call at their convenience. In the event respondents returned a partial survey, follow-up phone calls were made in an attempt to gain complete information. During the phone call interviews, respondents where asked if recording of the conversation were permissible. If so, the phone conversations were then recorded (Singleton & Straits, 1999).

According to Babbie (1990) survey literature suggests a wide range of acceptable response rates. Babbie (1990) and Mertens (1998) suggest that a response rate of 50% is generally considered adequate for analysis and reporting. A response rate of 60% is considered good, and a response rate of 70% is very good.

Data Analysis

All data from the pre- and post-instruments was entered into an automated electronic survey system available on the Internet (SurveyMonkey, 2004). The participants' responses were captured via a computer system. Once captured, the instrument data was electronically submitted from the survey Internet site to the researcher in a compressed zip file (Winzip, 2004). The compressed zip file contained multiple spreadsheet files in the open standard comma-separate-values (csv) format. These data files have been saved on a write-once only CD for security and transportability. The data was imported into and compiled using Microsoft Excel (2004)

spreadsheets or Statistical Packages for the Social Sciences (SPSS, 2004), a commercially available software used for data management, manipulations, and analysis.

The pre-survey consists of 20 yes/no questions, resulting in the collection of nominal data (Putt and Springer, 1989). This data is then divided into four functional categories, which are (a) attack tree familiarity, (b) cost benefit, (c) probability, and (d) structured query language program. The post-survey contains 41 questions; 20 of those questions are identical to the pre-survey group in the above-mentioned four categories. The data analysis for this segment was accomplished by first identifying the functional area profiles. After that, it was determined if the profiles had changed as a result of the program process. The significance of the survey results was measured using the *Chi-Squared Test of Homogeneity* (Aczel, 2002). Each of the 20 questions compared the pre- and post-survey responses evaluating if the respondent changed using the *Test for Equality of Proportions* (Aczel, 2002, p. 351).

As well as performing the z test on each of the 20 groupings of questions, a profile was also created for each of the four categories, (a) attack tree familiarity, (b) cost benefit, (c) probability, and (d) structured query language program. The profiles are displayed in a column or bar graph, one bar for each subcategory.

In addition to the 20 yes/no questions, the post-survey also contains an additional 20 Likert type queries, and 6 open-ended concluding questions. The 20 Likert questions consisted of five questions in each of the four subcategories with a respondent scale of 1 through 5. The responses in each category were combined to create interval data. Three of the questions are direct and two of the questions are negatively reversed. Aggregate analysis occurred in each subcategory with the negative responses being flipped, such as a five changed into a one, then the subcategory questions were totaled. Since there are

five questions in each subcategory with a respondent scale ranging from 1 to 5, the total range was 5 through 20. Bar graphs are used to display the distribution of the sums. Additional statistical analysis is reflected the mean and median of all responses communicated in a descriptive measure.

The final six open-ended questions on the post-survey are communicated in a narrative format. When relevant, the researcher did extract and assimilate similarities in the verbiage triggered by key words (Singleton & Straits, 1999).

The data was presented in a descriptive format using tables, graphs, and percentage calculations. The level of significance used for all statistical analysis is 0.05.

Academic Attack Tree Quantification

Schneier (1999; 2000) and Salter, Saydjari, Schneier, and Wallner (1998) have introduced attack trees proposing implementation models, steps, and ideas as to how one can obtain benefits from using such a model; however, there has been no quantification of an attack tree value or substantiation as to neither their effectiveness nor their viability. A study is needed to validate or disprove attack tree value.

Schneier (2000) also introduced an array of exceptional implementation ideas such as assigning values, continuous or finite numbers, and the idea of tree pruning without providing a clear methodology as to how one would implement such processes. What appears to be lacking from attack trees is a methodology providing a finer level of granularity capable of assisting information system managers with a clear repeatable process, quantified numbers, probability analysis, tree pruning process, and a cost benefit analysis tool. The next section of this paper submits a process addressing the previously mentioned deficiency.

Proposed Algorithm Protocols

Probability Protocol

Schneier (2000) suggested that attack tree nodes may be assigned the value *possible* or *impossible*. If, in fact, these values can be assigned to each node, one could raise the assignment of the value up one level to the gate or node connection point. The value assigned to represent this connector was NOT.

Based on Schneier's (2000) suggestion, NOT nodes represent sub-goals or leaves that exist but are highly improbable of being achieved. Incorporating the use of NOT nodes allows Boolean Algebraic calculations to be performed on attack trees reflected in Table 8, since gate connectors now obtain the three states required for probability and Boolean algebra of NOT, AND, and OR.

Table 8

Boolean Algebra

	Disciplines		
Operation	Probability	Mathematics	Attack Trees
Union of A and B	A or B	$A \cup B$	OR
Intersection of A and B	A and B	$A \cap B$	AND
Compliment of A and B	Not A	A'	NOT

Probability can also be used to determine an event's level of measure. Schneier (2000) also suggested that probability assignments can be made at each node. The outcome would be identification of the likelihood of an attack at key locations within an attack tree. This information may be used to identify which nodes of an attack tree should be addressed first to reduce the risk of an attack. The formula used to calculate probability is

$$P(A) = \frac{\text{The number of ways an event can occur}}{\text{The total number of possible outcomes}}$$

In this formula the probability of event A is the number of ways event A can occur divided by the total number of possible outcomes (Aczel, 2000).

Costing Protocol

Schneier (2000) suggested that cost information could be ascertained with attack trees by assigning continuous values to nodes. These continuous values, in reference to costing, would be a dollar figure. Assigning the cost of implementing an attack per a specific node allows one to assign costs to the attack tree totaling the potential cost of an attack against the tree. A second costing figure, or another perspective, may be the cost associated with defending from the attack, that is, the cost of the countermeasure.

Program Design

The computer program was developed using a structured query language (SQL) database. SQL is a standard language used to access data in a database. IBM first

developed SQL in the mid 1970s (Webopedia, 2003). The American National Standards Institute (ANSI) approved SQL as a standard in 1986 and in 1991 updated the SQL standard to SAG SQL. There are many corporations who have built SQL databases including IBM, Microsoft, Oracle, and Sybase. The application built here will utilize one of Microsoft's versions of SQL known as Access (Microsoft, 2003).

A database is built using a database name as the master file and table names as units which house the data. For example, a spreadsheet can be considered a table containing rows and columns of data. This application will contain one database, *attack tree master*, and three tables, *attack tree, Node,* and *AND* to house the data.

Implementation Process

This research introduces a costing model leveraging the attack tree paradigm using a top down and bottom up combination. The top down approach requires each node of the attack tree to be assigned a penetration cost. For example, assume a tool capable of viewing data packets as they were transported across a computer network, a process referred to as "packet sniffing." The cost of the packet sniffer is approximately $5,000 dollars. The penetration cost, P_1 for the "packet sniffing" node of an attack tree would be $5,000. To complete the attack tree each node, also referred to as penetration point, is assigned a value to the P_{in} attribute. Upon completion of the attack tree, summations calculations can be obtained identifying the penetration cost of a path through the attack tree $T_p = \sum(P_1 + \ldots + P_{n)}$. The process described here is the top-down view referring to beginning the process at the root node of the tree (T_p) and working itself down to the leaves.

The second viewpoint is the bottom up perspective in which the asset being evaluated is assigned a financial value. For example, if the goal of the attacker is to obtain the corporate employee payroll database, a value must be assigned by management to the database, perhaps a value of two hundred thousand dollars. In this example, asset A or A_1, is assigned a value. This asset resides at the base of the attack tree. Analysis can now be performed from the bottom of the tree upwards. Information systems managers are now able to ask if $A_1 > \sum(P_1 + P_2 + \ldots + P_n)$. This formula can assist with determining if the cost of protecting the asset is greater than the value of the asset itself.

Attack trees promote the creation of all feasible nodes somewhat mimicking the results of a brain storming session. If nodes exist in an attack tree that are highly improbable of coming to fruition, the P_{ox} value will dilute the cost assessment and therefore should be removed from the $A_1 > \sum(P_1 + P_2 + \ldots + P_n)$ inequality representing a more realistic measurement of the probable penetration points P_p to the asset, A_1. The formula is then tweaked with the non-probable penetration points removed and their respective costing $A_1 > \sum((P_{p1} + P_{p2} + \ldots + P_{in}) - (NP_{p1} + NP_{p2} + \ldots + NP_{pn}))$. The process of removing non-probable penetration points is known as “pruning” the tree. To effectively prune that attack tree, the use of Boolean algebra is introduced here identifying each branch with one of three values, AND, OR, or NOT.

The values are assigned to the node connectors with one of three visual representations. As introduced by Schneier (1999; 2000) the AND connectors are linked together with a semicircle connection, as reflected in Figure 17.

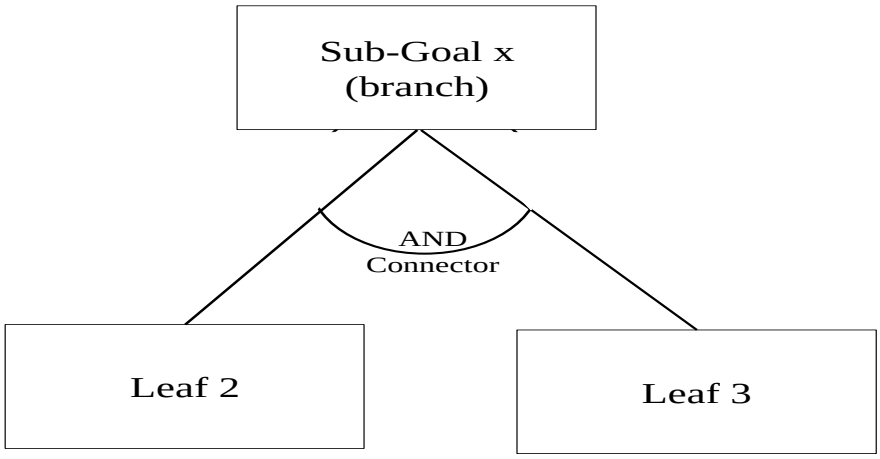

Figure 17. Attack tree using an AND connector.

Attack tree connections built using AND connections require both leaves to be satisfied by the attacker in order to achieve the goal node. For example, if an attacker were attempting to obtain the combination to a safe by reading one's email, Leaf 1 would include *Obtaining Safe Owners Email* while Leaf 2 would include *Written Safe Combination*, both leaves must be satisfied to achieve access to the upper node.

Often attackers are able to satisfy one of many choices in order to obtain access to the upper node. For example, if the attack tree goal is to gain access to one's house, two of the leaves may be (a) *Use Key*, or (b) *Climb Through Open Window*. Of the two mentioned penetration points, either one will suffice representing an OR node connection. Figure 18 displays an OR connection in which either Leaf 1 or Leaf 2 may be fulfilled satisfying the requirements to obtain access to the above subgoal.

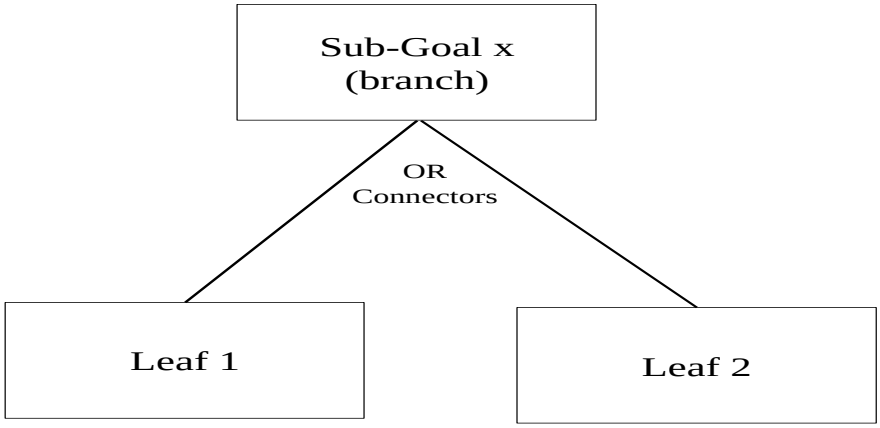

Figure 18. Attack tree using an OR connector.

Although a leaf may exist as a possible penetration point in an attack, there may be, based on the attack tree builder's knowledge of the information system environment, a high probability that the leaf will never be satisfied. If this is the case, the attack tree is then pruned, essentially disabling the improbable branch at the point of deniability. For example, one of the leaves in an *Open Safe* attack tree is *Bribe.* The ability to open a safe due to bribing someone with knowledge of the safe combination is a real threat. However, what if the person was "un-bribe-able"? For instance, imagine that an *Open Safe* attack tree is being built for the safe in Microsoft Corporation's executive offices. The probability that an attacker would be able to bribe Mr. Gates is extremely low or highly improbable at best. In such an instance, the leaf connector to the *Bribe* node of the attack tree would use a NOT connector. NOT connectors are new to the attack tree paradigm and are being introduced for the first time in this paper. Figure 19 represents a NOT connector.

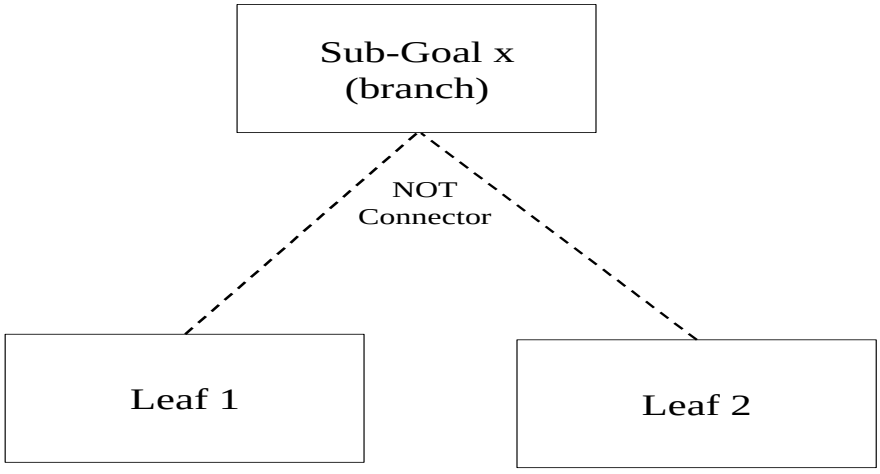

Figure 19. Attack tree using an OR connector.

The NOT connector is represented by a dashed line. Fault trees use connectors known as Gates that reside between the nodes leveraging the Fault Tree paradigm. The solution proposed here uses the lines themselves as connectors. This use allows multiple connector types to exist for one branch with many leaves. As reflected in Figure 20, a single node may have a combination of AND, NOT, and OR connectors in as many combinations as necessary.

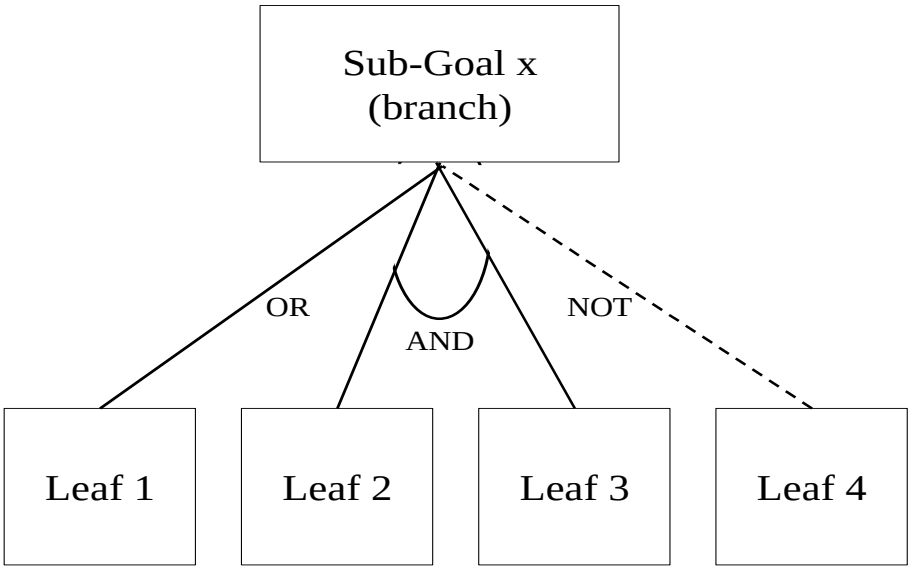

Figure 20. Attack tree using a combination of AND, OR, and NOT connectors.

Identifying Tasks

Viescas (1999) lists the application design fundamentals, built upon the Yourdon and Constantine work of the 1960s, to include steps for identifying tasks, charting task flow, identifying the data elements, application construction, and testing. For the research design, the first two tasks are listed as Figure 21, identifying tasks, and Figure 22, SQL program task flow diagram. The identification of data elements has been reflected in the database schema provided in Table 9. The database for this research was populated with data representing the attack tree *Access Web Server* denoted in Figure 23.

- Enter Attack Tree
    - Enter Tree
    - Enter Goal
    - Enter Outline Number
    - Enter OR Nodes
    - Enter AND Nodes
    - Enter NOT Nodes
    - Enter Leaves
    - Cost of asset loss
- Assign values to Leaves
    - Cost of penetration
    - Cost of countermeasure implementation
    - Probability of attack on leaf
    - Time estimate to implement countermeasure (in days)
- Assign weights to Leaves
    - Risk (High/Medium/Low)
    - Access (High/Medium/Low)
    - Cost (High/Medium/Low)
- Generate Attack Tree
    - Print text version
- Prune Attack Tree (output to printer or screen)
    - Prune by cost of penetration
        - Ability to enter a cost
        - Print a report
    - Prune by cost of asset loss
        - Ability to enter a cost
        - Print a report
    - Prune by cost of countermeasure implementation
        - Ability to enter a cost
        - Print a report
    - Prune by probability
        - Ability to enter an acceptable probability
        - Print a report

Figure 21. SQL program task identification.

Task Flow

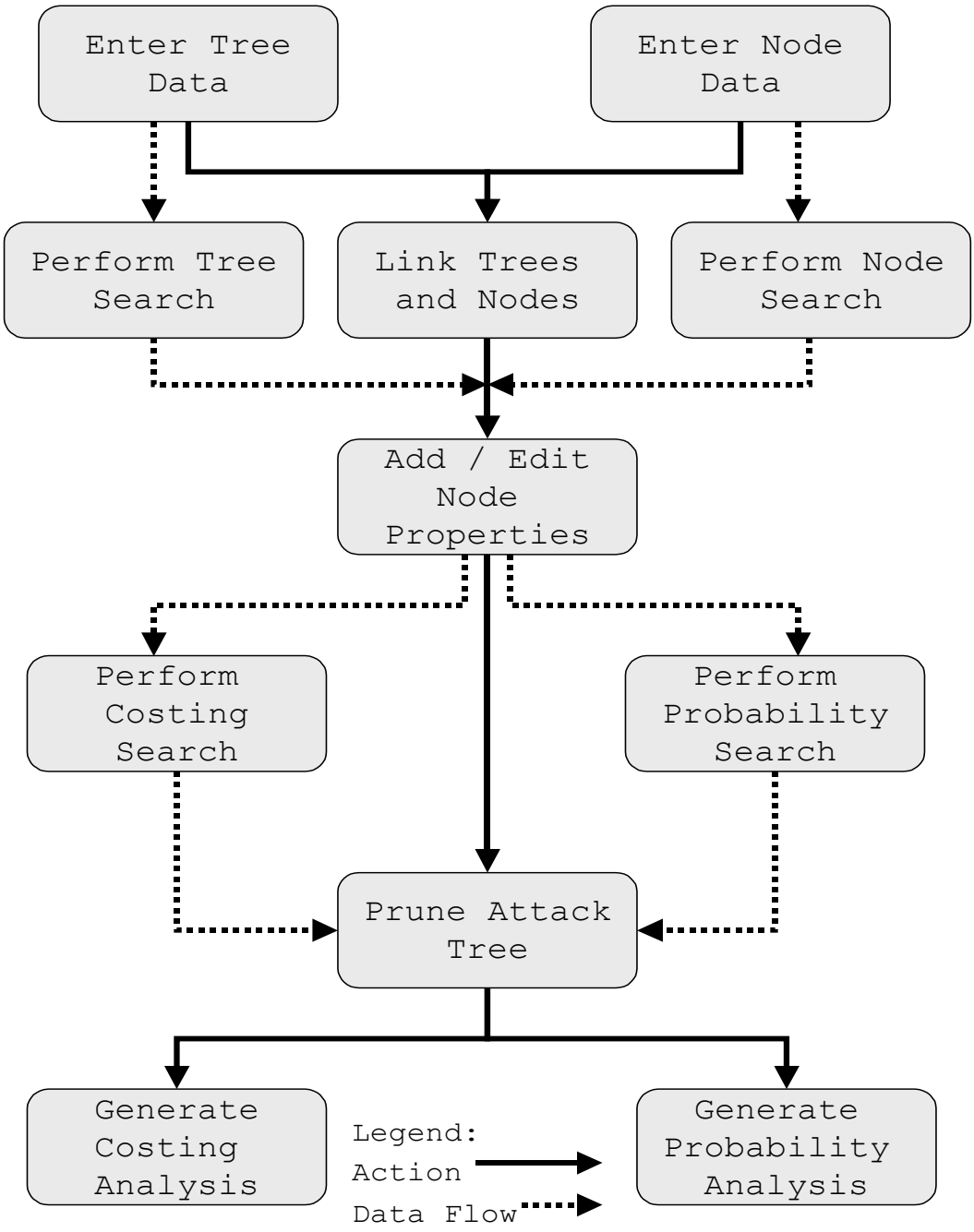

Figure 22. SQL program task flow diagram.

Attack Tree

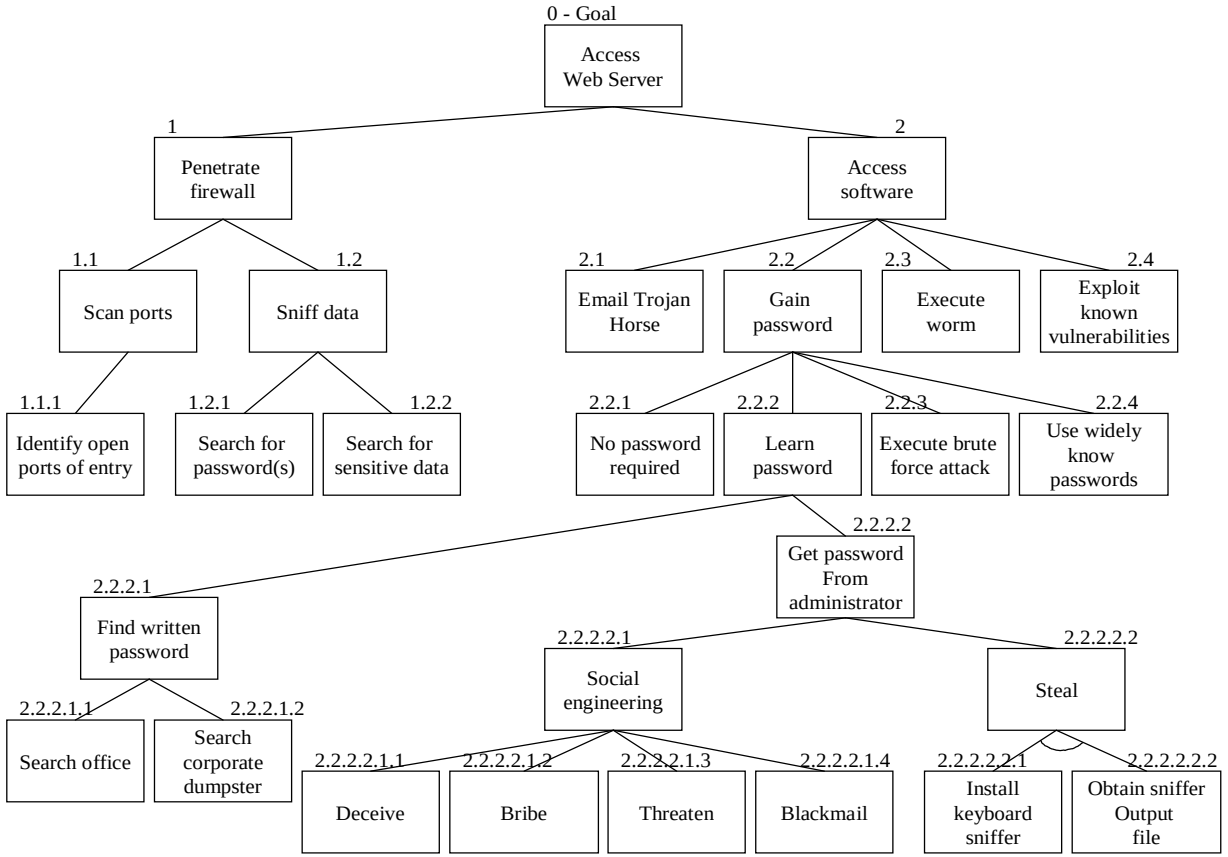

Figure 23. Graphical web server attack tree.

Database schema

Table 9

Node Table

Name of Field in Database Table	Type of Data
Node Number	Computer Assigned Number
Node Name	Variable Length Character
Attack Tree Number	Number
Upper Node Number	Number
Lower Leaf Number	Number
Level Number	Number
Type of Node (AND, OR, NOT)	One Character (A, O, or N)
AND Number	Number
Penetration Cost	Dollar Figure
Node Value	Dollar Figure
Risk (High, Medium, or Low)	One Character (H, M, or L)
Access (High, Medium, or Low)	One Character (H, M, or L)

Table 10

Leaf Table

Name of Field in Database Table	Type of Data
Leaf Number	Number
Leaf Name	Variable Length Character
Assessed Leaf Value	Dollar Figure

Table 11

Tree Table

Name of Field in Database Table	Type of Data
Attack Tree Number	Number
Attack Tree Name	Variable Length Character

Program Summary

Attack trees appear to offer information system managers a viable solution as a tool used to identify penetration points and areas of exposure that an attacker may leverage. Schneier (1999, 2002) introduced attack trees including ideas on their usage while Salter et al. (1998) introduced a methodology for attack tree implementation. Current literature, however, lacks a quantification of attack tree usage. This paper reviews the current body of attack tree knowledge and offers an implementable method quantifying attack tree analysis using probability, Boolean algebra, and cost-benefit analysis. This method of attack tree analysis has been designed into a computer application utilizing a SQL database.

Research Design Summary

Chapter 3 contains the evaluation research methodology that was used to evaluate attack tree analysis. The research design includes details of the target sample, sampling procedure, target sample, research instruments, data collected procedures, and the data analysis plan.

Chapter 3 also includes proposed algorithms to assist with a costing protocol and a probability protocol that may assist information system managers with costing and probability decisions. These algorithms have been designed into a computer program that was used as a component of this research. The design of this program is also included in chapter 3.

Chapter 4 includes data results from the research design that was proposed in chapter 3. The data includes tests on the results from the pre- and post-assessment surveys. The significance of the survey results was measured using the *Chi-Squared Test of Homogeneity* (Aczel, 2002). The *Test for Equality of Proportions* was used to analyze the data (Aczel, 2002, p. 351).

Chapter 5 includes the conclusions and recommendations of the research. Additional recommendations on future work have also been integrated in chapter 5.

CHAPTER 4: RESULTS

Introduction

In this chapter, all results of this dissertation research are described. The three research questions are evaluated through data analysis of the pre- and post-survey results. The information is explained in tables, graphs, and written description. Variables include familiarity of attack tree analysis, costing analysis, probability analysis, and use of a structured query language (SQL) simulation model built in a computer program.

Demographics

The invitation to participate in the research was distributed via electronic mail to 59 candidates. Of the 59 electronic mail submissions, 3 were returned as rejected electronic mail addresses; therefore, the population size is considered to be 56. Of the 56 research candidates, 18 (32%) completed the pre-survey and 15 (27%) completed the post-survey.

The data collection time frame was extended from 1 month to 3 months to allow for participants to respond. The process included 2 follow up phone calls to 24 participants and 5 series of follow up electronic mail requests. Pilot data was combined with the research data allowing the numbers to reach an acceptable research sample. The pilot data consisted of a sample of 5 in which 4 (80%) completed the pre-survey and 4 (80%) completed the post-survey.

Familiarity with Attack Trees

The pre- and post-survey contained 8 questions used to identify the participants' familiarity with attack trees. Responses to the 8 questions were categorized as "yes" or

"no". Table 12 contains the data results from the chi-square test for independence that was applied using the hypotheses:

Question: What change in familiarity with attack tree analysis occurred within the sample as a result of the simulation exercise?

A chi-squared test of homogeneity was used to test whether there was a change in the way the questions were answered after the simulation exercise. The level of significance used was .05. The particular hypothesis set that was tested was:

H_0: The Pre-Yes and Post-Yes groups are distributed the same; the groups are homogeneous.

H_1: The Pre-Yes and Post-Yes groups are distributed differently; the groups are not homogeneous.

The calculated value of the test statistic, χ^2, was 2.3124, and the p-value was 0.9406 using 7 degrees of freedom, *df*. Since the p-value was greater than .05, the null hypothesis was not rejected. The data suggests that participants were familiar with attack tree analysis prior to the simulation exercise.

Table 12

Attack Tree Familiarity – Frequency Data

	Frequency Data		
Question	Pre-Yes	Post-Yes	p-value
Q1	18	15	
Q2	14	13	
Q3	7	8	
Q4	12	11	
Q5	6	4	
Q6	13	6	
Q7	13	13	
Q8	18	15	
Totals	101	85	0.9406

When reviewing the expected frequency in the chi-squared calculation, occurrences of expected frequency data are suspect if the value is less than 5. Table 13 displays the data set in which one datum is less than 5 in Q5, Post-Yes has a value of 4.57. Therefore, the results are suspect.

Table 13

Attack Tree Familiarity – Expected Frequency Data

	Expected Frequency Data	
Question	Pre-Yes	Post-Yes
Q1	17.9	15.1
Q2	14.7	12.3
Q3	8.15	6.85
Q4	12.5	10.5
Q5	5.43	4.57
Q6	10.3	8.68
Q7	14.1	11.9
Q8	17.9	15.1

Survey question 1, *I am familiar with the term attack tree,* showed an increase in participant familiarity with the term attack tree from the pre-survey rate of 94.06% to the post-survey rate of 100%. The data from question 1 identified one participant who was not familiar with the term attack tree. Table 3 lists the responses as to familiarity with the term attack tree.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 1 in the posttest was different than that for the pretest.

H_0: p1 – p2 >= 0

H_1: p1 – p2 < 0

The data indicates that H_0 is not rejected. The difference between p1 (p1=pretest) and p2 (p2=posttest) does not imply a significant improvement. The data suggests that participants did not significantly improve their familiarity with the term on attack trees based on their use of the simulation model in this study.

Table 14

I am familiar with the term attack tree

	Responses	
	Frequency	Percentage
Pre		
Yes	18	94.7
No	1	5.3
Post		
Yes	15	100.0
No	0	0.0

Null Hypothesis	p-value	Decision
p1 – p2 >= 0	0.1836	not reject

Survey question 2, *I am familiar with attack tree methodology,* showed that 73.7% of the participants were familiar with the attack tree methodology during the pre-survey. The post-survey response of 86.7% showed an increase in familiarity with attack tree methodology of 17.64%. Table 4 lists the responses to familiarity with the attack tree methodology.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 2 in the posttest was different than that for the pretest.

H_0: $p1 - p2 >= 0$

H_1: $p1 - p2 < 0$

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests the sample population may not have significantly improved their familiarity with the attack tree methodology based on their use of the simulation model in this study.

Table 15

I am familiar with the attack tree methodology

	Responses	
	Frequency	Percentage
Pre		
Yes	14	73.7
No	5	26.3
Post		
Yes	13	86.7
No	2	13.3

Null Hypothesis	p-value	Decision
$p1 - p2 < 0$	0.1763	not reject

Survey question 3, *I have created an attack tree,* showed that 36.8% of the participants had not created an attack tree prior to this experiment. The post-survey response of 53.3% showed an increase in participants who have created an attack tree. Therefore, at least one participant who had never created an attack tree prior to taking the pre-survey appears to have created an attack tree prior to completing the post-survey. Table 5 lists the responses on familiarity with the attack tree methodology.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 3 in the posttest was different from that for the pretest.

H_0: $p1 - p2 >= 0$

H_1: $p1 - p2 < 0$

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests the sample population may not have significantly improved their ability to create an attack tree based on their use of the simulation model in this study.

Table 16

I have created an attack tree

	Responses	
	Frequency	Percentage
Pre		
Yes	7	36.8
No	12	63.1
Post		
Yes	8	53.3
No	7	46.7

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.7184	Not reject

Survey question 4, *I understand attack trees well enough to create an attack tree,* showed that 63.2% of the participants believe they understand attack tree analysis well enough to create an attack tree. The post-survey response of 73.3% showed an increase in participants who believe they know attack tree analysis well enough to create an attack tree by -13.77%. Table 6 lists the responses as to the participants' belief in their ability to create an attack tree.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 4 in the posttest was different from that for the pretest.

H_0: $p1 - p2 >= 0$

H_1: $p1 - p2 < 0$

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly improved their understanding of attack trees well enough to create an attack tree based on their use of the simulation model in this study.

Table 17

I understand attack trees well enough to create an attack tree

	Responses	
	Frequency	Percentage
Pre		
Yes	12	63.2
No	7	36.8
Post		
Yes	11	73.3
No	4	26.7

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.2644	Not reject

Survey question 5, *I have used attack tree as a risk methodology,* showed that 31.6% of the participants have used attack tree analysis as a risk assessment methodology. The post-survey response of 26.7% showed a decrease in participants who have used attack trees as a risk methodology of 15.51%. Table 6 lists the responses as to the participants' belief in their ability to create an attack tree.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 5 in the posttest was different than that for the pretest.

H_0: $p1 - p2 >= 0$

H_1: $p1 - p2 < 0$

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly improved their use of attack trees as a risk methodology based on their use of the simulation model in this study.

Table 18

I have used attack trees as a risk methodology

	Responses	
	Frequency	Percentage
Pre		
Yes	6	31.6
No	13	68.4
Post		
Yes	4	26.7
No	11	73.3

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.6225	Not reject

Survey question 6, *We currently have a process to identify systems vulnerabilities,* showed that 68.4% of the participants believe they have a process to identify systems vulnerabilities. The post-survey response of 40.0% showed a decrease in participants who have a process to identify system vulnerabilities by 71.0%. Table 8 lists the responses as to the participants who currently have a process to identify systems vulnerabilities.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 6 in the posttest was different from that for the pretest.

H_0: $p1 - p2 >= 0$

H_1: $p1 - p2 < 0$

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly incorporated use of a current process to identify system vulnerabilities based on their use of the simulation model in this study.

Table 19

We currently have a process to identify systems vulnerabilities

	Responses	
	Frequency	Percentage
Pre		
Yes	13	68.4
No	6	31.6
Post		
Yes	6	40.0
No	9	60.0

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.3050	Not reject

Survey question 7, *An attack tree is an extremely useful tool when identifying security vulnerabilities,* showed that 68.4% of the participants believe an attack tree is an extremely useful tool when identifying security vulnerabilities. The post-survey response of 86.7% showed a decrease in participants who believe that attack tree analysis is an extremely useful tool when identifying security vulnerabilities by 21.11%. Table 9 lists the responses of the participants who currently have a process to identify systems vulnerabilities.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 7 in the posttest was different than that for the pretest.

H_0: p1 – p2 >= 0

H_1: p1 – p2 < 0

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests the sample population may not have significantly improved their belief that an attack tree is an extremely useful tool when identify security vulnerabilities based on their use of the simulation model in this study.

Table 20

An attack tree is an extremely useful tool when identifying security vulnerabilities

	Responses	
	Frequency	Percentage
Pre		
Yes	13	68.4
No	6	31.6
Post		
Yes	13	86.7
No	2	13.3

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.1065	Not reject

Survey question 8, *Attack tree analysis is a useful tool,* showed that 94.7% of the participants believe an attack tree is a useful tool. The post-survey response of 100.0% showed an increase in participants who believe an attack tree is a useful tool by 5.3%. Table 10 lists the responses of the participants who believe attack tree analysis is a useful tool.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 8 in the posttest was different than that for the pretest.

H_0: $p1 - p2 >= 0$

H_1: $p1 - p2 < 0$

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests the sample population may not have significantly improved their belief that attack tree analysis is a useful tool based on their use of the simulation model in this study.

Table 21

Attack tree analysis is a useful tool

	Responses	
	Frequency	Percentage
Pre		
Yes	18	94.7
No	1	5.3
Post		
Yes	15	100.0
No	0	0.0

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.1836	Not reject

Missing Data

This study incorporated the listwise deletion or complete case analysis approach to missing data. (Allision, 2002; Little & Rubin, 1987). The cases in which data was missing were omitted from the analysis. Listwise deletion often results in a substantial decrease in the sample size available for analysis; however, the important advantage of data missing at random leads to unbiased parameter estimation. During this research only one participant failed to complete the data. This participant completed the survey consent form, and then failed to respond to any of the questions.

There were 3 participants who completed the pre-survey, but failed to complete the post-survey. Data analysis was performed based on the percentages of the participants not the actually number of participants. Therefore, the pre-survey data in which the 3 participants did not complete a post-survey were included in the analysis.

Research Question 1

In this section, the results of research question 1 are analyzed based on the data produced from the pre- and post-survey sections that pertained to the first research question. Research question 1 evaluated attack tree analysis' effectiveness as to cost analysis and the ability to assist information systems managers responsible in making budgetary decisions. Specifically, research question 1 states "[h]ow effectively might the inclusion of attack tree analysis be incorporated into a computer cost analysis model capable of assisting information systems managers with budgetary decisions?" Survey questions 9 though 12 addressed the participant's current cost benefit analysis process as related to risk assessment. Table 11 contains the data results from the chi-square test for homogeneity that was applied using the hypotheses:

Research Question 1: How effectively might the inclusion of attack tree analysis be incorporated into a computer cost analysis model capable of assisting information systems managers with budgetary decisions?

The effectiveness of the inclusion of attack tree analysis incorporated into a computer cost analysis model capable of assisting information systems mangers with budgetary decisions was evaluated by using a chi-squared test of homogeneity was used to test whether there was a change in the way the questions were answered after the simulation exercise. The level of significance used was .05. The particular hypothesis set that was tested was

H_0: The Pre-Yes and Post-Yes groups are distributed the same; the groups are homogenous.

H_1: The Pre-Yes and Post-Yes groups are distributed differently; the groups are not homogenous.

The calculated value of the test statistic, χ^2, was 2.6193, and the p-value was 0.4541 using 3 degrees of freedom, *df*. Since the p-value was greater than .05, the null hypothesis was not rejected. The data suggests that the inclusion of attack tree analysis incorporated into a computer cost analysis model capable of assisting information systems managers with budgetary decisions as a result of the simulation exercise did not have significant improvement.

Table 22

Costing Analysis – Frequency Data

	Frequency Data		
Question	Pre-Yes	Post-Yes	p-value
Q9	6	2	
Q10	4	1	
Q11	3	3	
Q12	16	15	
Totals	29	21	0.4541

When reviewing the expected frequency data within the costing analysis data set, occurrences of expected frequency data are suspect if the value is less than 5. Table 12 displays the data set in which 6 data are less than 5. Therefore, the results are suspect.

Table 23

Costing Analysis- Expected Frequency Data

	Expected Frequency Data	
Question	Pre-Yes	Post-Yes
Q9	4.64	3.36
Q10	2.9	2.1
Q11	3.48	2.52
Q12	18	13

Survey question 9, *We currently have a process to identify prioritization of countermeasures from a costing perspective*, showed that 33.3% of the participants do currently incorporate a process to assist with costing analysis of risk assessment. The post-survey response of 14.3% showed a decrease by 57.06%. Table 13 lists the responses by the participants who currently incorporate a process to assist with costing analysis of risk assessment.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 9 in the posttest was different than that for the pretest.

H_0: $p1 - p2 >= 0$

H_1: $p1 - p2 < 0$

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly improved their incorporation of a process to identify prioritization of countermeasures from a costing perspective based on their use of the simulation model in this study.

Table 24
We currently have a process to identify prioritization of countermeasures from a costing perspective.

	Responses	
	Frequency	Percentage
Pre		
Yes	6	33.3
No	12	66.7
Post		
Yes	2	14.3
No	12	85.7

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.8915	Not reject

Survey question 10, *We currently have a process to identify the most effective allocation of funds offering the highest rate of return on security vulnerabilities*, showed that 22.2% of the participants do currently incorporate a process to identify the most effective allocation of funds offering the highest rate of return on security vulnerabilities. The post-survey response of 7.1% showed a decrease by 212.67%. Table 14 lists the responses by the participants who currently incorporate a process to identify the most effective allocation of funds offering the highest rate of return on security vulnerabilities.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 10 in the posttest was different than that for the pretest.

H_0: p1 – p2 >= 0

H_1: p1 – p2 < 0

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests the sample population may not have significantly improved their current process to identify the most effective allocation of funds offering the highest rate of return on security vulnerabilities based on their use of the simulation model in this study.

Table 25
We currently have a process to identify the most effective allocation of funds offering the highest rate of return on security vulnerabilities.

	Responses	
	Frequency	Percentage
Pre		
Yes	4	22.2
No	14	77.8
Post		
Yes	1	7.1
No	13	92.9

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.8781	Not reject

Survey question 11, *We currently are considering incorporating attack tree analysis to assist with budgetary decisions as related to the allocation of funds for security,* showed that 16.7% of the participants are considering the incorporation of attack tree analysis to assist with budgetary decisions as related to the allocation of funds for security. The post-survey response of 21.4% showed an increase by 21.96%. Table 15 lists the responses by the participants who are currently considering incorporating attack tree analysis to assist with budgetary decisions as related to the allocation of funds for security.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 11 in the posttest was different than that for the pretest.

H_0: $p1 - p2 >= 0$

H_1: $p1 - p2 < 0$

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly improved the process in which they are currently considering incorporating attack tree analysis to assist with budgetary decision as related to the allocation of funds of security based on their use of the simulation model in this study.

Table 26
We currently are considering incorporating attack tree analysis to assist with budgetary decisions as related to the allocation of funds of security.

	Responses	
	Frequency	Percentage
Pre		
Yes	3	16.7
No	15	83.3
Post		
Yes	3	21.4
No	11	78.6

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.3660	Not reject

Survey question 12, *I believe that attack tree analysis can be a useful process used to assist with budgetary decisions as related to the allocation of funds for security*, showed that 88.9% of the participants believe that attack tree analysis can be a useful process used to assist with budgetary decisions as related to the allocation of funds for security. The post-survey response of 92.9% showed a decrease by 4.49%. Table 16 lists responses by the participants who believe that attack tree analysis can be a useful process when used to assist with budgetary decisions as related to the allocation of funds for security.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 12h in the posttest was different than that for the pretest.

H_0: $p1 - p2 >= 0$

H_1: $p1 - p2 < 0$

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly improved their belief that attack tree analysis can be a useful process used to assist with budgetary decisions as related to the allocation of funds for security based on their use of the simulation model in this study.

Table 27
I believe that attack tree analysis can be a useful process used to assist with budgetary decisions as related to the allocation of funds for security.

	Responses	
	Frequency	Percentage
Pre		
Yes	16	88.9
No	2	11.1
Post		
Yes	13	92.9
No	1	7.1

Null Hypothesis	p-value	Decision

P1 – p2 >= 0	0.3820	Not reject

The post-survey also contained a series of 5 questions relating to costing analysis. The questions captured the participants response on a scale of 1 to 5 with 1=disagree and 5=agree. Two of the questions, b and e, were reverse-scored. The results of those two questions have been flipped for the analysis, and are listed in Table 17. The specific questions were:

(a) Attack trees can be used to identify the protection cost of a system.

(b) Attack trees cannot be used to identify the vulnerability cost of a system.

(c) Attack trees are an effective decision tool to be used in cost benefit analysis.

(d) I was able to receive cost benefit decision-making value while using attack tree analysis.

(e) Attack trees are an ineffective decision tool to be used in cost benefit analysis.

Table 28

Additional questions from the post-survey costing analysis section.

	Responses				
Question	Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Agree
a)	0	0	0	5	6
b)	0	3	2	2	2
c)	0	0	8	8	2
d)	1	0	2	2	2
e)	1	0	4	4	3
Totals	2	3	12	21	15
Percentages	3.8	5.7	22.6	39.6	28.3

Figure 1 contains a bar graph from the post-survey on cost benefit analysis. The data shows 22.6% are neutral when asked if attack tree analysis would assist with cost benefit. Participants who either disagree or somewhat disagree included 9.5%. While participants who somewhat agree or agreed that attack tree analysis may assist with cost benefit totaled 67.9%. The bar graph data suggests that attack tree analysis may be useful in assisting with cost benefit analysis.

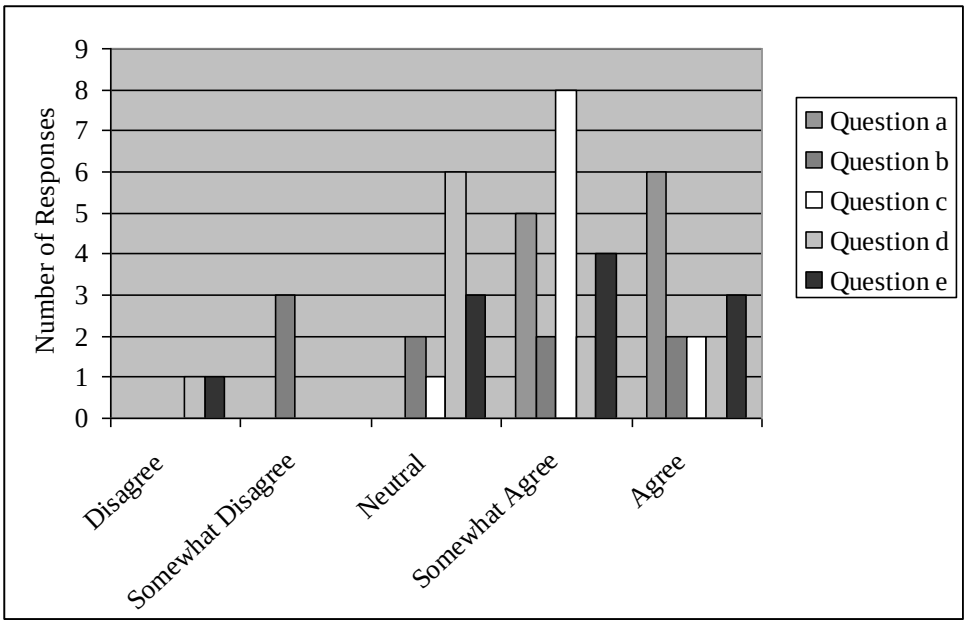

Figure 24. A bar graph built from the post-survey data on cost benefit analysis.

Research Question 2

In this section, the results of research question 2 are analyzed based on the data produced from the pre- and post-survey sections that pertained to the second research question. Research question 2 evaluated attack tree analysis' effectiveness as to leveraging a probability model and evaluating the ability to assist information system managers in making human resource allocation decisions. Specifically, research question 2 states "[h]ow effectively might the inclusion of attack tree analysis be incorporated into a computer probability model capable of assisting information systems managers with human resource allocation?" Survey questions 13 though 16 addressed the participant's current probability process as related to risk assessment. Table 18 contains the data results from the chi-square test for independence that was applied using the hypotheses:

Research Question 2: How effectively might the inclusion of attack tree analysis be incorporated into a computer probability model capable of assisting information systems managers with human resource allocation?

The effectiveness of the inclusion of attack tree analysis being incorporated into a computer probability model capable of assisting information systems managers with human resource allocation was evaluated by using a chi-squared test of homogeneity was used to test whether there was a change in the way the questions were answered after the simulation exercise. The level of significance used was .05. The particular hypothesis set that was tested was

H_0: The Pre-Yes and Post-Yes groups are distributed the same; the groups are homogenous.

H_1: The Pre-Yes and Post-Yes groups are distributed differently; the groups are not homogenous.

The calculated value of the test statistic, χ^2, was 2.817, and the p-value was 0.9634 using 3 degrees of freedom, *df*. Since the p-value was greater than .05, the null hypothesis was not rejected. The data suggests that the inclusion of attack tree analysis incorporated into a computer probability model capable of assisting information systems managers with human resource allocation as a result of the simulation exercise did not have a significant improvement.

Table 29

Probability – Frequency Data

	Frequency Data		
Question	Pre-Yes	Post-Yes	p-value
Q13	4	4	
Q14	2	1	
Q15	5	4	
Q16	11	8	
Totals	22	17	0.9634

When reviewing the expected frequency data within the probability data set, occurrences of expected frequency data are suspect if the value is less than 5. Table 19 displays the data set in which 5 data are less than 5. Therefore, the results are suspect.

Table 30

Probability – Expected Frequency Data

	Expected Frequency Data	
Question	Pre-Yes	Post-Yes
Q13	4.51	3.49
Q14	1.69	1.31
Q15	5.08	3.92
Q16	10.7	8.28

Survey question 13, *We currently have a process to help identify prioritization of countermeasures from a human resource allocation,* showed that 22.2% of the participants do currently have a process to help identify prioritization of countermeasures from a human resource allocation. The post-survey response of 28.6% showed a decrease by 22.38%. Table 20 lists the responses by the participants who currently have a process to help identify prioritization of countermeasures from a human resource allocation.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 13 in the posttest was different from that for the pretest.

H_0: $p1 - p2 >= 0$

H_1: $p1 - p2 < 0$

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly improved their incorporation of a current process to help identify prioritization of countermeasures from a human resource allocation based on their use of the simulation model in this study.

Table 31
We currently have a process to help identify prioritization of countermeasures from a human resource allocation.

	Responses	
	Frequency	Percentage
Pre		
Yes	4	22.2
No	14	77.8
Post		
Yes	4	28.6
No	10	71.4

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.1219	Not reject

Survey question 14,*We currently have a process to identify effective allocation of human resources offering the highest rate of return on security vulnerabilities*, showed that 11.1% of the participants do currently have a process to identify effective allocation of human resources offering the highest rate of return on security vulnerabilities. The post-survey response of 7.1% showed a decrease in participants who currently incorporate a process by 56.34%. Table 21 lists the responses by the participants who currently have a process to identify effective allocation of human resources offering the highest rate of return on security vulnerabilities

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 14 in the posttest was different than that for the pretest.

H_0: p1 – p2 >= 0

H_1: p1 – p2 < 0

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly improved their incorporation of a current process used to identify effective allocation of human resources offering the highest rate of return on security vulnerabilities based on their use of the simulation model in this study.

Table 32
We currently have a process to identify effective allocation of human resources offering the highest rate of return on security vulnerabilities.

	Responses	
	Frequency	Percentage
Pre		
Yes	2	11.1
No	16	88.9
Post		
Yes	1	7.1
No	13	92.9

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.6488	Not reject

Survey question 15, *We currently are considering incorporating attack tree analysis to assist with staffing assignment decisions as related to the allocation of human resources,* showed that 27.8% of the participants are considering incorporating attack tree analysis to assist with staffing assignment decisions as related to the allocation of human resources. The post-survey response of 28.6% showed a decrease by 2.80%. Table 22 lists the responses by the participants who are currently considering incorporating attack tree analysis to assist with staffing assignment decisions as related to the allocation of human resources.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 15 in the posttest was different than that for the pretest.

H_0: p1 – p2 >= 0

H_1: p1 – p2 < 0

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly improved their consideration of incorporation attack tree analysis to assist with staffing assignment decisions as related to the allocation of human resources based on their use of the simulation model used in this study.

Table 33
We currently are considering incorporating attack tree analysis to assist with staffing assignment decisions as related to the allocation of human resources.

	Responses	
	Frequency	Percentage
Pre		
Yes	5	27.8
No	13	72.2
Post		
Yes	4	28.6
No	10	71.4

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.4802	Not reject

Survey question 16, *I believe that attack tree analysis can be a useful process used to assist with staffing assignment decisions as related to the allocation of human resources*, showed that 61.1% of the participants believe that attack tree analysis can be a useful process when used to assist with staffing assignment decisions as related to the allocation of human resources. The post-survey response of 72.7% showed a decrease by 15.96%. Table 23 lists the responses by the participants who believe that attack tree analysis can be a useful process when used to assist with staffing assignment decisions as related to the allocation of human resources.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 16 in the posttest was different than that for the pretest.

H_0: p1 – p2 >= 0

H_1: p1 – p2 < 0

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly improved their belief that attack tree analysis can be a useful process used to assist with staffing assignment decisions as related to the allocation of human resources based on their use of the simulation model in this study.

Table 34
I believe that attack tree analysis can be a useful process used to assist with staffing assignment decisions as related to the allocation of human resources.

	Responses	
	Frequency	Percentage
Pre		
Yes	11	61.1
No	7	38.9
Post		
Yes	8	72.7
No	3	27.3

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.6737	Not reject

The post-survey also contained a series of 5 questions relating to probability. The questions captured the participants response on a scale of 1 to 5 with 1=disagree and 5=agree. Two of the questions, b and e, were reverse-scored. The results of those two questions have been flipped for the analysis, and are listed in Table 24. The specific questions were:

(a) Attack trees can be used to identify the protection probability of a system.

(b) Attack trees cannot be used to identify the vulnerability probability of a system.

(c) Attack trees are an effective decision tool to be used in probability analysis.

(d) I was able to receive probability decision-making value while using attack tree analysis.

(e) Attack trees are an ineffective decision tool to be used in probability analysis.

Table 35

Additional questions from the post-survey probability section.

	Responses				
Question	Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Agree
a)	0	0	2	5	4
b)	0	3	3	3	2
c)	0	0	4	6	1
d)	0	0	5	3	2
e)	0	0	5	3	2
Totals	0	3	19	20	11
Percentages	0	5.7	35.8	37.7	20.8

Figure 2 contains a bar graph from the post-survey on probability analysis. The data shows 35.8% are neutral when asked if attack tree analysis would assist with probability analysis. Participants who either disagree or somewhat disagree included 5.7%. While participants who somewhat agree or agreed that attack tree analysis may assist with cost benefit totaled 58.5%. The bar graph data suggests that attack tree analysis may be a useful incorporating probability analysis to assist information system mangers with human resource allocation decisions.

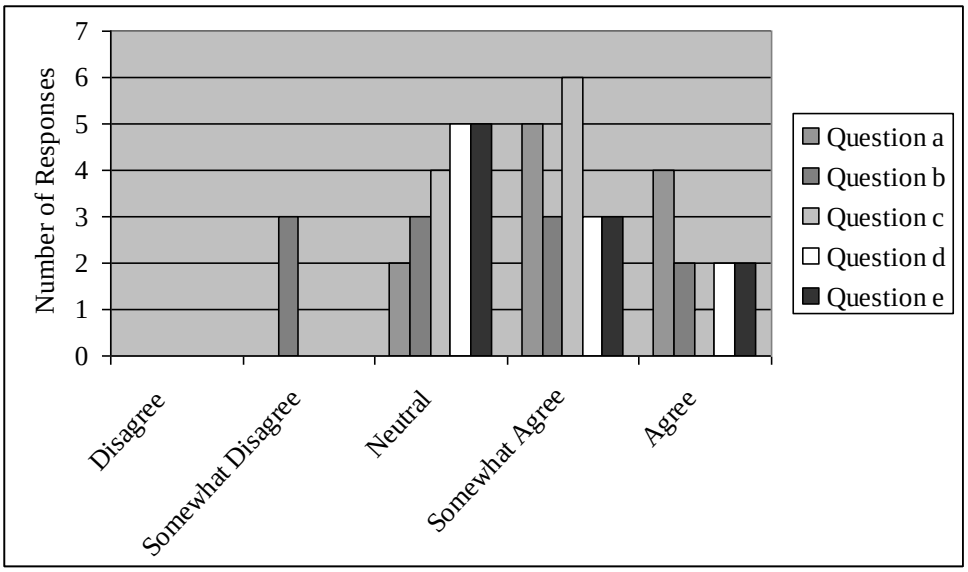

Figure 25. A bar graph built from the post-survey data on probability analysis.

Research Question 3

In this section, the results of research question 3 are analyzed based on the data produced from the pre- and post-survey sections that pertained to the third research question. Research question 3 evaluated the premise that the inclusion of a structured query language (SQL) would simplify, for information systems managers, the process of cost analysis and the allocation of human resources using a probability model. Specifically, research question 3 states "[h]ow effectively might the inclusion of a structured query language (SQL) database program be implemented to simplify the use of a cost analysis model and a probability model to assist information systems managers with costing and human resource allocation decisions? " Table 25 contains the data results from the chi-square test for independence that was applied using the hypotheses:

Research Question 3: How effectively might the inclusion of a structured query language (SQL) database program be implemented to simplify the use of a cost analysis model and a probability model to assist information systems managers with costing and human resource allocation decisions?

The effectiveness of the inclusion of a structured query language (SQL) database program implemented to simplify the use of a cost analysis model and a probability model to assist information systems managers with costing and human resource allocation decisions was evaluated by using a chi-squared test of homogeneity was used to test whether there was a change in the way the questions were answered after the simulation exercise. The level of significance used was .05. The particular hypothesis set that was tested was

H_0: The Pre-Yes and Post-Yes groups are distributed the same; the groups are homogenous.

H_1: The Pre-Yes and Post-Yes groups are distributed differently; the groups are not homogenous.

The calculated value of the test statistic, χ^2, was 1.5684, and the p-value was 0.6666 using 3 degrees of freedom, *df*. Since the p-value was greater than .05, the null hypothesis was not rejected. The data suggests that the inclusion of a structured query language (SQL) database program implemented to simplify the use of a cost analysis model and a probability model to assist information systems managers with costing and human resource allocation decisions as a result of the simulation exercise did not have significant improvement.

Table 36

Structured Query Language – Frequency Data

	Frequency Data		
Question	Pre-Yes	Post-Yes	p-value
Q17	1	1	
Q18	0	1	
Q19	15	10	
Q20	12	11	
Totals	34	23	0.6666

When reviewing the expected frequency data within the structured query language data set, occurrences of expected frequency data are suspect if the value is less than 5. Table 26 displays the data set in which 4 data are less than 5. Therefore, the results are suspect.

Table 37

Structured Query Language – Expected Frequency Data

	Expected Frequency Data	
Question	Pre-Yes	Post-Yes
Q17	1.1	0.9
Q18	0.55	0.45
Q19	13.7	11.3
Q20	12.6	10.4

Survey question 17, *Our current process used to identify security cost benefit analysis is automated,* showed that 5.6% of the participants do currently have an automated process used to identify security cost benefit analysis. The post-survey response of 7.1% showed an increase by 26.78%. Table 27 lists the responses by the participants who currently have an automated process used to identify security cost benefit analysis.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 17 in the posttest was different than that for the pretest.

H_0: $p1 - p2 >= 0$

H_1: $p1 - p2 < 0$

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly improved their incorporation of an automated process used to identify security cost benefit analysis based on their use of the simulation model used in this study.

Table 38

Our current process used to identify security cost benefit analysis is automated.

	Responses	
	Frequency	Percentage
Pre		
Yes	1	5.6
No	17	94.4
Post		
Yes	1	7.1
No	13	92.9

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.4270	Not reject

Survey question 18, *Our current process used to identify security human resource allocation is automated,* showed that 0.0% of the participants currently have an automated process used to identify security human resource allocation. The post-survey response of 7.1% showed an increase in participants who currently incorporate a process by ∞. Table 28 lists the responses by the participants who currently have an automated process used to identify security human resource allocation.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 18 in the posttest was different than that for the pretest.

H_0: p1 – p2 >= 0

H_1: p1 – p2 < 0

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly improved their current process used to identify security human resource allocation based on their use of the simulation model used in this study.

Table 39
Our current process used to identify security human resource allocation is automated.

	Responses	
	Frequency	Percentage
Pre		
Yes	0	0.0
No	18	100.0
Post		
Yes	1	7.1
No	13	92.9

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.1247	Not reject

Survey question 19, *I believe that attack tree analysis can be a useful process when incorporated into a SQL program,* showed that 83.3% of the participants believe that attack tree analysis can be a useful process when incorporated into a SQL program. The post-survey response of 71.4% showed a decrease by 16.67%. Table 29 lists the responses by the participants who believe that attack tree analysis can be a useful process when incorporated into a SQL program.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 19 in the posttest was different from that for the pretest.

H_0: p1 – p2 >= 0

H_1: p1 – p2 < 0

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly improved their belief that attack tree analysis can be a useful process when incorporated into a SQL program based on their use of the simulation model used in this study.

Table 40
I believe that attack tree analysis can be a useful process when incorporated into a SQL program

	Responses	
	Frequency	Percentage
Pre		
Yes	15	83.3
No	3	16.7
Post		
Yes	10	71.4
No	4	28.6

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.7905	Not reject

Survey question 20, *Attack tree analysis using a structured query language database program is capable of pruning attack tree scenarios,* showed that 66.7% of the participants believe that attack tree analysis using a structured query language database program is capable of pruning attack tree scenarios. The post-survey response of 78.6% showed an increase by 15.14%. Table 30 lists the responses by the participants who believe that attack tree analysis using a structured query language database program is capable of pruning attack tree scenarios.

A test for equality of proportions was done to determine whether the proportion of those answering "yes" to question 20 in the posttest was different than that for the pretest.

H_0: p1 – p2 >= 0

H_1: p1 – p2 < 0

The data indicates that H_0 is not rejected. The difference between p1 and p2 implies no significant improvement. The data suggests that the sample population may not have significantly improved their belief that attack tree analysis using a structured query language database program is capable of pruning attack tree scenarios based on their use of the simulation model used in this study.

Table 41
Attack tree analysis using a structured query language database program is capable of pruning attack tree scenarios

	Responses	
	Frequency	Percentage
Pre		
Yes	12	66.7
No	6	33.3
Post		
Yes	11	78.6
No	3	21.4

Null Hypothesis	p-value	Decision
P1 – p2 >= 0	0.2287	Not reject

The post-survey also contained a series of 5 questions relating to SQL simulation model. The questions captured the participants' response on a scale of 1 to 5 with 1=disagree and 5=agree. The specific questions were:

(a) Attack tree analysis using a structured query language database program is an effective process capable of incorporating "what-if" statements.

(b) The automated attack tree analysis program is an effective process capable of assisting with vulnerability risk assessment.

(c) The automated attack tree analysis program is an effective process capable of assisting with cost analysis of security decisions.

(d) The automated attack tree analysis program is an effective process capable of assisting with human resource allocation of security decisions.

(e) I was able to receive decision-making value while using the attack tree SQL program.

Table 42

Additional questions from the post-survey SQL simulation model.

	Responses				
Question	Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Agree
a)	0	0	1	4	6
b)	0	0	3	3	5
c)	0	0	4	4	3
d)	1	1	3	4	2
e)	0	0	5	2	4
Totals	1	1	16	17	20
Percentages	1.8	1.8	29.1	30.9	36.4

Figure 3 contains a bar graph from the post-survey on cost benefit analysis. The data shows 29.1% are neutral when asked if the SQL simulation model assisted with attack tree analysis. Participants who either disagree or somewhat disagree included 3.6% and 67.3% of participants somewhat agree or agreed that SQL simulation model did assist with attack tree analysis. The bar graph data suggests that a SQL simulation model may assist with attack tree analysis.

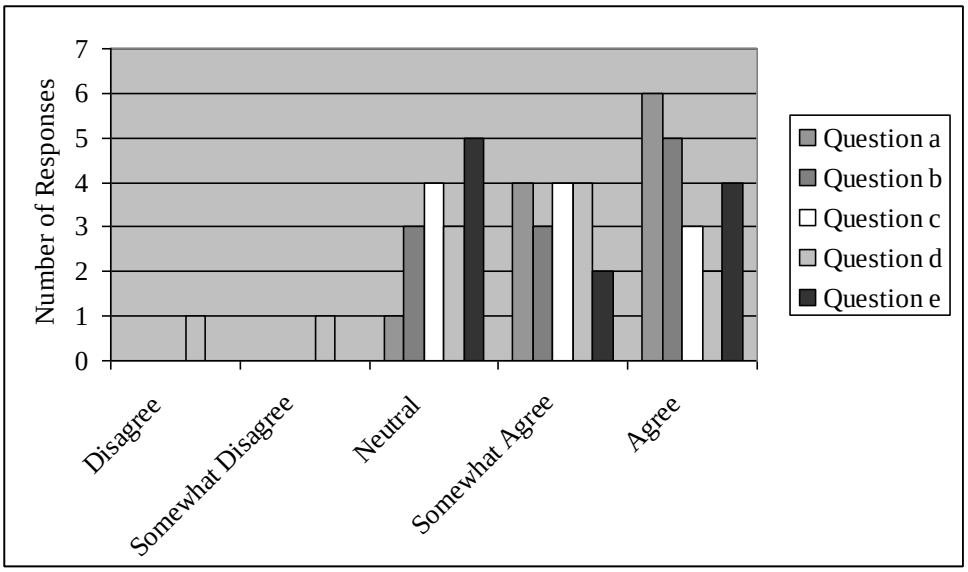

Figure 26. A bar graph built from the post-survey data on SQL simulation model.

Summary

This chapter presented the findings of the study in written, table, and graph form. The data was displayed to show how the various participants responded to the research questions from the pre- and post-survey instruments. The questions were analyzed by the grouping of (a) familiarity with attack trees, (b) cost benefit, (c) probability, and (d) SQL simulation model.

Among the findings, the results suggested that the sample was familiar with attack tree analysis. A trend implied that attack tree analysis may assist with costing analysis and probability while being used in a SQL based simulation model. The following chapter presents a discussion of the findings.

CHAPTER 5: CONCLUSIONS AND RECOMMENDATIONS

Introduction

Chapter 1 of this paper contained an introduction to the study that detailed why the topic of an evaluation of attack tree analysis using an SQL based simulation model is an area of vital importance to the quickly advancing field of information systems management and enterprise security. The chapter reviewed the significant and the social implications of this study and how governments, corporations, and individuals may benefit from the findings. Chapter 2 contained a literature review of the leading risk assessment models used in the information systems discipline. This research covered leading risk assessment models as well as the variables which are incorporated in the risk assessment methodology used to assist information systems managers with the decision making process, specifically using cost benefit and probability to assist with risk assessment. Chapter 3 contained information on the details of the survey methods and design used in this study, and chapter 4 presented the data in narrative, tabular, and graphical format.

This chapter contains an introduction, the three research questions including conclusions, and recommendations of each. A discussion of the results including interpretations and conclusions drawn from the findings are also included in this chapter. The limitations of the study are explored as well as the implications these findings may have on information systems managers. Suggestions for future research are offered as a guide that additional research may build upon. Finally, this chapter concludes with a summary.

Research Question 1

How effectively might the inclusion of attack tree analysis be incorporated into a computer cost analysis model capable of assisting information systems managers with budgetary decisions?

The results of this research question indicate that most participants, 85.7%, do not currently have a process to prioritize or identify the most effective allocation of funds for computer security. The results also indicated that 92.9% believe that attack tree analysis can be a useful process when used to assist with budgetary decisions as related to the allocation of funds for security. Finally, the results displayed a consistent increase, 88.9 – 92.9%, in the participants' belief that attack trees may be used with costing decisions from the pre- to the post-survey, indicating a strengthening in consensus as the participants became more familiar with attack tree analysis process through use of the SQL simulation model.

Conclusions

The data suggests that many information systems experts and managers do not appear to realize the degree to which internal processes are lacking relative to the support of funding decisions related to computer security. The data indicated that participants became more aware of their lack of processes after exposure to the SQL simulation model. For example, post-survey results indicated an increase of 212.67% in the participants' loss of confidence in their current process for cost analysis as related to security after exposure to the SQL simulation model. One may conclude from these results that information systems managers lack a sufficient understanding of the security

threats and vulnerabilities of their organization, and the ability to identify the steps that must be taken, also known as countermeasures, to secure the threat. This lack of an understanding of the steps precludes ones ability to associate a cost with the process.

As implied by analysis of the data, 92.9% of information systems managers indicated that attack tree analysis can be a useful process in assisting with budgetary decisions as related to the allocation of funds for security. However, only a marginal percentage of 21.4% are considering implementing attack tree analysis as a tool to assist with funding decisions. The data did not reveal why 92.9% of participants believe the process can be helpful, but only 21.4% are now considering implementing attack tree analysis to assist with budgetary decisions.

Recommendations

Information systems managers should identify the value of assets, including information and data, currently under their area of responsibility. Once the value has been identified, one has a framework of operation. Assets should be inventoried including cost of assets, cost of compromise, and cost of protection (countermeasure implementation).

Secondly, the system vulnerabilities must be identified. Attack tree analysis will help in creating a textual or graphical display of all assets and their known weak or penetration points. One is then able to assign a dollar value to implementation of the countermeasure, or process to protect the asset.

Finally, a process or methodology should be implemented to assist with the security funding. This process, combined with the knowledge of asset value and countermeasure implementation will assist information systems managers in budgetary

decisions. It appears as thought attack tree analysis may offer assistance to this process by allowing weights to be assigned to each point of consideration.

Research Question 2

How effectively might the inclusion of attack tree analysis be incorporated into a computer probability model capable of assisting information systems managers with human resource allocation?

The results of this research question indicate that most participants, 71.4%, do not currently have a process to prioritize or identify the most effective allocation of human resources to implement computer security. The results also indicated that 72.7% believe that attack tree analysis can be a useful process when used to assist with human resource allocation decisions as related to security. Finally, the results displayed a consistent increase of 15.96% in the participants' belief that attack tree analysis may be used with human resource allocation from the pre- to the post-survey, indicating a strengthening in consensus as the participants became more familiar with attack tree analysis process through use of the SQL simulation model.

Conclusions

The data indicates that 28.6% of the participants have a process to assist with the prioritization of countermeasure implementation with human resource allocation; however, only 7.1% believe their current process is capable of providing the managers with the highest rate of security return for the resource investment. These numbers may indicate a corporate culture in which managers are assigning employees in a reactive manner. The data showed that allocation of 71.4% of the work force is done without a

prioritization process. This may indicate an environment in which managers are assigned resources based on a reactionary manner as opposed to preventive maintenance.

Information system managers appear to be assigning only one tenth of their staff to tasks which the managers feel will provide them with the greatest return for their investment. The results failed to ascertain the reasoning behind this datum.

The results indicate that 72.7% of information systems managers believe attack tree analysis can be a useful process is assisting with human resource allocation decisions as related to the allocation of staff for security; however, only 28.6% are considering implementing attack tree analysis as a tool to assist with staffing decisions. The data did not revel why 72.7% of participants believe the process can be helpful, but only approximately 28.6% are now considering implementing attack tree analysis to assist with staffing decisions.

Recommendations

Information system managers should identify the countermeasures that are required to be implemented within their departments and areas of responsibility. The countermeasures should then be prioritized relative to the greatest return on the investment. The data indicates that only 28.6 % of participates have a process with which to help identify prioritization of countermeasures from human resource allocation.

One may conclude from the data results that information system managers are assigning human resource based on a reactive manner. The data indicates that 7.1% of human resources are assigned to tasks in which effective allocation of human resources offers the highest rate of return on security investment. Information system managers

should assign human resources on quantitative proactive manners based on the prioritized countermeasure data gained from the previously mentioned recommendation.

Information system managers should also incorporate a methodology to be used with human resource allocation. The data indicates 92.9% of participants do not have a process to assist with staffing. Lack of a process may lead to the ineffectiveness of an information systems manager's ability to proactively allocate his/her staff and the ability to objectively review the staff's performance.

Research Question 3

How effectively might the inclusion of a structured query language (SQL) database program be implemented to simplify the use of a cost analysis model and a probability model to assist information systems managers with costing and human resource allocation decisions?

The data results of research question 3 indicate that only 7.1% of participants have an automated process to be used for cost benefit analysis. Research question 1 evaluated the use of costing analysis to assist with attack tree analysis. Research question 3 evaluates the automation of such a process.

The data indicates that 7.1% of participants have a process for human resource allocation. Research question 2 evaluated the use of probability analysis to assist with attack tree analysis. Research question 3 evaluates the automation of such a process.

When evaluating the effectiveness of attack tree analysis incorporated into a structured query language, 71.4% of participates believe attack tree analysis can be a useful process when incorporated into a structured query language model.

Examining the structured query language model's capabilities from a deeper perspective, 78.6% of participants believe that attack tree analysis is a structured query language program is capable of pruning attack tree scenarios. The process of pruning an attack tree equates to the procedure of branch removal. An attack tree is a hierarchical representation. Pruning of the attack tree allows one to temporarily remove limbs of the tree, or paths of the hierarchies that are not of importance. This process allows one to focus more intently on areas of the tree that have been identified as important based on some criteria of relevance.

Conclusions

The data suggests that most participants, 92.9%, do not incorporate an automated process used to identify security cost benefit analysis. This process includes the automation of a costing process that does not necessarily include attack tree analysis. One may conclude based on data from the participants' response, that most costing decisions are manual. One may also include the process is then highly subjective. Subjective processes tend to be more error prone then objective processes.

Based on the data from this research, 92.9% of participants do not incorporate an automated process used to identify security human resource allocation. This process includes the automation of a security human resources process that does not necessarily include attack tree analysis. One may conclude based on data from the participants' response, that most human resource allocation decisions are manual. One may also conclude that the process is then highly subjective. Subjective processes tend to be more error prone then objective processes.

The data suggests the achievability of attack tree analysis automation. Salter, Saydjari, Schneier, & Wallner (1998) suggested that attack tree analysis could not be automated. The data shows that 71.4% of the participants believe that the process can be automated, and automation of the process in a structured query language is a useful model.

One of the advantages of tree analysis lies in the ability to prune sections of the tree that no longer require focus. The participants of this research, 78.6% believe that attack trees built using the SQL simulation model were capable of pruning. The SQL simulation model, incorporating complex SQL macros, provided participants a process for attack tree pruning.

Recommendations

The data suggests that an automated process, such as a structured query language program, may be useful in assisting information system managers with cost analysis and human resource allocation decisions. Information systems mangers may benefit from creating a repeatable measurable automatic process. The data suggests that the current process incorporated for costing and human resource allocation decisions is subjective, based on the lack of automation. Information system managers may experience a benefit by applying an objective quantifiable process.

Information system managers should automate the process used with security costing decisions. Automation may take the form of a structured query language based on complex SQL macros. The automation of costing decisions allows information system mangers to incorporate an objective methodology that may reduce the margin of error on security costing analysis.

Information system managers should automate the process used with security human resource allocation decisions. Automation may take the form of a structured query language based on complex SQL macros. The automation of human resource allocation decisions may allow information system mangers to incorporate an objective methodology that may reduce the margin of error on human resource allocation costing analysis.

The automation of attack tree analysis is a useful tool for information system managers. The analysis tools should be incorporated into the security departments of organizations. The tool can be purchased by external vendors or developed internally in a computer programming language as simple as SQL using complex macro development and a relational data base.

Discussions

This research effort requested participation from the world's leading authorities on attack tree analysis and computer security including practitioners and scholars. Experts were identified by peer review publications, research projects, and publications. Many of the leading academic professors chose not to participate in this study. Most of the leading security experts in the field, practitioners, did respond and a few did participate.

The data fails to capture the scholarly perspective from tier 1 research universities based on their decision not to participate in this effort. The data is missing input from tier 1 universities.

The leading security experts who have incorporated attack tree analysis within major security engagements at Fortune 100 companies have struggled with the process.

Participant comments suggest that attack tree analysis works well in theory, and in small models, but is not implementable in a large organization. One participant concluded that "[a]ttack trees are a useful theory, but we have not found them useful in practice. After publishing initial work on attack trees in 'Building Secure Software', we have abandoned the approach at [Fortune 100 Company]."

The basis for the difficulty with attack tree analysis lies within the ability to create an attack tree. Attack tree creation relies on the security experience of the creator. For example, if one is to create an attack tree on breaking into a safe, one must know the different ways that one may break into a safe. Also, a level of complexity lies in the creation of different branches within the attack tree that include many of the same functionality. For instance, if one were to create an attack tree on breaking into a house. The branch on *Entering Through a Window* would have many similarities as the branch on *Entering Though a Door.* One may simply open the window or door, one may destroy the window or door, and one may remove the hinges of the window or door. This process of branch reuse is complex. The process is magnified as the complexity of the attack tree grows. One participant concluded that he is "not convinced that attack trees are useful across the spectrum to convey commonality or true measurement comparisons. Also, since each tree is individually drawn and created, it is largely up to the expertise, inventiveness and creativity (read: sinister) of the creator to come up with enough scenarios to provide value."

The attack tree for this research was built by the researcher. Participating subjects did not have to create the attack tree. All attack tree node values were assigned by the researcher. The process of creating an attack tree was beyond the scope of this research.

The process is difficult and time consuming and most likely would have reduced the amount of participants.

The structured query language (SQL) simulation model was built by the researcher and included an installation program plus a one hundred page user manual. Although this model was acceptable for this research, a complete attack tree analysis model built using a computer program or modeling language would have been more comprehensive and would have required more time. For example, the SQL simulation model used in this study did not incorporate a dynamic visual representation of an attack tree. The dynamic attack tree was represented in textual format. Dynamic attack tree is defined as the computer program's ability to adjust the produced attack tree report based on new data. Even though the attack tree was pre-built, participants had the ability to modify the attack tree. This modification was accurately represented when the textual attack tree was viewed by the participants. Participant comments also included the request for a *wizard*. Computer programs which guide the users through a series of questions thereby simplifying the user participation in the computer program is referred to as a wizard.

The participants also commented on the ability to incorporate attack tree analysis into a SQL program. The data suggests that a SQL program was acceptable for a small pilot program, such as the attack tree built and used during this research effort. However use of a more comprehensive computer programming modeling language may be required for larger, more complex attack trees. Specifically, one participant commented that the "SQL program would need to be replaced with a comprehensive statistical analysis tool and probably would benefit from being implemented in Prolog or another expert system language."

Limitations

The pre- and post-survey instruments may require restructuring. Each instrument contained four questions focusing on costing analysis, four questions focusing on probability analysis, and four questions focusing on the SQL simulation model. Of the four questions targeting costing and probability, 75% evaluated the participant's current processes. The instruments should be extended containing more questions that focus on attack tree analysis used with costing analysis and attack tree analysis used with probability analysis. The pre- and post-survey instruments may have been limited in their ability to ascertain comprehensive data on utilizing attack tree analysis to participate in costing analysis and human resource allocation based on probability.

The researcher created the attack tree used in this research. The limitations as to the complexity of the attack tree lie within the researcher's level of security experience. This level of experience may have been inadequate to fully represent a complex attack tree.

The size of the attack tree may also have limited the research results. The attack tree was constructed to fit within the scope of this research and the limitations of anticipated time investment by the participants. As attack trees become larger and therefore more complex, it follows that there is a much larger time investment required of the individual study participants. A larger more complex attack tree may have offered a more comprehensive scale in which the participants could have formulated the notion of using attack tree analysis to assist with cost benefit analysis and probability analysis when using a risk assessment model for managerial budgetary and resource planning.

The computer model used in this research was built using complex structure query language (SQL) macros and custom computer programming including Microsoft Access, Microsoft Office Studio Tools, and Microsoft Visual Basic. The tool offered fair modeling capabilities satisfying the scope of this research effort. The tool came pre-populated with an attack tree built by the researcher. The participants were able to manipulate the attack tree by creating, reading, updating, and deleting all nodes of the attack tree. The SQL simulation model did not provide a dynamic graphical attack tree representation. A graphical attack tree may have provided a better mechanism for participants with predominant visual learning characteristics allowing them the ability to process the concepts visually.

Attack trees are fairly new to computer security professionals and academics; therefore, in order to target knowledgeable participants, a sample population was created using published authors from the public forum, researchers, and computer security experts. The total population identified was 56, of which only 53 had valid electronic mail addresses. The number of participants who chose to participate in this research effort was 18. This number was reached by extending the data collection window by 300%. Statistical analysis incorporated the use of chi-square of homogeneity. When reviewing the expected frequency data of the research questions comparing pre- and post-survey data, values less than 5 are suspect. The data results are suspect. The suspect results are a result of a small data size. A larger sample population may produce data results that are less suspect.

Contributions

This study has added to the existing body of knowledge for the risk assessment of computer security systems by providing an academic evaluation of attack trees whose viability and usefulness may extend to information systems managers, government agencies, military organizations, and private citizens who have home computers connected to the Internet. As requested by Salter, Saydjari, Schneier, and Wallner (1998, p. 2), this study provided a step in bridging the gap and facilitating "dialog among academia, industry, and government toward securing the global information infrastructure."

The process of developing attack trees was automated by a computer program that housed the mathematical properties contained within computer algorithms that incorporated probability, Boolean algebra, and cost benefit analysis that may have aided information systems managers and security consultants in system analysis (Schneier, 1999, 2000). This program and process may aid in the ability to run countermeasure scenarios and "what-ifs" also adding to the security of information systems.

Many aspects of society that incorporate information systems may have benefited from the results of this study. This positive social change includes a process to achieve a more secure society obtained in a cost effective manner identifying the best use of human resource allocation. The positive social change may be founded in the research algorithms used to assist with the cost and probability protocols. Social entities that may benefit include governmental organizations that may be able to reduce the risk of terrorism by identifying vulnerabilities and penetration points previously unrecognized and left unprotected. Additional social entities that may benefit include corporations such as electrical companies and the airline industry, which may be able to identify where to

invest funding in order to achieve the highest benefit in countering terrorism and threats. As requested by Salter, Saydjari, Schneier, and Wallner (1998, p. 2), this study appeared to have contributed positive movement towards bridging the gap and creating a foundation for the facilitation of "dialog among academia, industry, and government toward securing the global information infrastructure." This research may help to guide society into a more secure information technology infrastructure by evaluating a risk assessment model capable of identifying and reducing vulnerabilities in systems, processes, and policies.

The results of this study and the processes appear to have produced a model that may aid all interested parties in the effort to reduce the risks of exposure. These risks often exist external to the software applications themselves, and mitigating these risks in the most cost effective manner ensures the highest probability for success. These risks include the ability to identify penetration points that appear attractive to terrorist organizations. Attack tree analysis allows one to view a target from the attacker's perspective. This non-restrictive protocol offers an additional vantage point from the securing entity. Attack tree analysis appears to provide society a model for positive social change in the creation of a safer world, not only from an information technology perspective, but from an all-inclusive methodology extending to all entities, such as shipping ports, air travel, military intelligence, construction, facilities, and transportation to name a few.

Implications for Future Research

This study provided an evaluation of attack tree analysis based on a structure query language simulation model. The data indicated that 92.9% of the participants

believe that attack tree analysis can be a useful process used to assist with budgetary decisions as related to the allocation of funds for security. Only 21.4%, however, are considering incorporating attack tree analysis to assist with budgetary decisions as related to the allocation of funds for security. This trend also appears in the allocation of human resources inasmuch as 72.7% of the participants believe attack tree analysis to be an effective process when used to assist with staffing assignments decisions as related to the allocation of human resources. Nevertheless, only 28.6% of the participants are considering incorporation attack tree analysis to assist with staffing assignments decisions as related to the allocation of human resources. The delta between these two data points is 71.5% for costing analysis and 44.1% for probability analysis. Further research is required to identify why the participants see the usefulness of the attack tree process, yet are not willing to incorporate attack tree analyses into their organizations.

Utilization of a structure query language computer program incorporating complex macros and customer computer programming offered an adequate model for this research effort; however, the programming environment was restricted due to the capabilities of SQL. Future research which includes a more complex programming language, such as the computer programming language of Prolog or an expert system language. Utilizing an expert system computer programming language may allow additional complexities to be added to the simulation model used in this research.

The attack tree used in this research was created by the researcher. This allowed the participants to focus on attack tree concepts and not attack tree creation. Further research could allow the participants to create the attack tree. This process may be implemented in case study methodology. One such example may be a cost / benefit data case study on a high payback project.

Further research is required to evaluate the use of attack tree analysis in a large complex setting. The data indicated that attack tree analysis added value as a risk assessment model assisting with costing and probability analysis; however, the data also suggested that attack trees, though useful in theory, may reach a point of uselessness in large organizations. A large attack tree implementation is required to explore this hypothesis.

The three groups of data associated with the respective research questions on (a) costing analysis, (b) probability analysis, and (c) structure query language simulation model were all evaluated by using a chi-squared test of homogeneity. The chi-squared test contained expected frequency data in which each datum must be not less than 5 otherwise the results are suspect. Each of the three chi-squared tests of homogeneity performed on each group of data pertaining to the research questions contained expected frequency data with 1 to 3 data less than 5; thereby producing suspect data results in which the null was not rejected. The data results were suspect since chi-squared assumes a large population. Additional research may proceed by running the same research as performed here with a larger sample size.

The test for equality of proportions was performed on each of the 20 questions contained on the pre and post-instruments. All 20 statistical tests produced results in which the null was not rejected. This may be due to the research effort sampling the wrong population. The purposeful sample included in this research targeted participants who were knowledgeable of attack trees, thereby resulting in pre and posttests indicating no significant improvement from the pre to post test. Future research may include the performance of similar research on a random population.

Finally, future implications may include extending the study of attack tree analysis beyond the computer security discipline. While attack trees have been introduced into the realms of computer security, the process incorporated appears to extend beyond the information systems discipline. For example, Home Land Security may be able to use attack tree analysis to identify vulnerabilities at a shipping port. Public Policy may be able to use attack tree analysis to assist with securing a water dam or a city's electrical grid. The symbolic relationship that exists between public policy and information systems management appears to provide a potential opportunity for methodology, policy, and procedural reuse.

Summary

This chapter presented a discussion of the results including interpretations and conclusions drawn from the findings. The three research questions were discussed including conclusions and recommendations. This chapter also included a discussion of the qualitative data. The limitations of the study were explored as well as the implications the findings may have on information systems managers. Contributions to the body of knowledge, society, and social change were all discussed in chapter 5. Implications for future research were provided. The chapter concluded with a summary of the conclusions and recommendations.

This research evaluated the effectiveness of attack tree analysis incorporated into an information system computer security risk assessment methodology. The problem is that attack tree analysis is in its infancy lacking an in-depth academic study and rigorous testing (Salter, Saydjari, Schneier, & Wallner, 1998). This issue has created the current gap between applied and theoretical notions of the attack tree model. This research

explored the effectiveness of using attack tree analysis to assist with costing decisions, probability analysis, and the viability of using structured query language (SQL) computer program.

The research design of this study was evaluation research. The data-gathering technique included a purposive sample of 56 computer security experts and leading academic authorities of attack tree analysis. Data presentation included a mixed model approach that includes qualitative and quantitative analysis. Pre- and post-assessment surveys were developed to ascertain the effectiveness of using attack tree analysis. This research also included the design, develop, and use a SQL computer program model.

Many aspects of society that incorporate information systems may be able to benefit from the results of this study. Corporations and governmental organizations may be able to reduce the risk of terrorism by identifying vulnerabilities and penetration points previously unrecognized and may also be able to best allocate funding and human resources to minimize vulnerabilities to all known threats in the most time-efficient, cost-effective manner.

REFERENCES

Aczel, A. D., & Sounderpandian, J. (Eds.). (2002). *Complete business statistics* (5th ed.). New York: McGraw-Hill.

Allison, P.D. (2002). *Missing data.* Thousand Oaks, CA: Sage.

Alreck, P., & Settle, R. (1985). *The survey research handbook.* IL: Irwin.

Amenaza Technologies Limited (2001). *Understanding IT risk through threat tree modeling.* Retrieved October 10, 2003, from Amenaza Web Site: http://www.amenaza.com/request_methodology.html

Andrews, J. D., & Dunnett, S. J. (Eds.). (1997). *Event tree analysis using binary decision diagrams.* Loughborough University: Department of Mathematical Sciences.

Andrews, J. D., & Moss, T. R. (1992). *Reliability and risk assessment*. Fairfield, NJ: American Society of Mechanical Engineers.

Andrews, J. D., & Moss, T. R. (2002). *Reliability and risk assessment* (2nd ed.). Fairfield, NJ: American Society of Mechanical Engineers.

AS/NZS 4360 (1999). *Australian / New Zealand standard for risk management 4360:1999.* Risk Management Unit: The University of New South Wales, Sydney, Australia.

Aven, T. (Ed.). (1992). *Reliability and risk analysis.* New York: Elsevier Applied Science.

Babbie, E. (1990). *Survey research methods* (2nd ed.). Blemont, CA: Wadsworth.

Bauer, M. D. (Ed.). (2002). *Building secure servers with Linux.* Sebastopol, CA: O'Reilly Books.

Bieber, G. (2000, March 14). *DOD information system security education, training, awareness & products*. Retrieved October 10, 2003, from Department of Defense Web Site: http://csrc.nist.gov/organizations/fissea/presentations/2000/Gbieber_FISSEA.ppt

Blyth, A. J. (Ed.). (2001). *Information assurance: Computer communications & networks.* New York: Springer-Verley.

Bray, T., Paoli, J., Sperberg-McQueen, C. M., & Maler, E. (2000, October 5). *Extensible markup language (XML) 1.0 (second edition)*. Retrieved October 15, 2003, from World Wide Web Consortium Web Site: http://www.w3.org/TR/2000/REC-xml-20001006

Burns, N., & Grove, S.K. (1993). *The practice of nursing research: Conduct, critique, & utilization* (2nd ed.). Philadelphia: W. B. Sanders Company.

CERT. (2004). *CERT/CC statistics 1998-2003*. Retrieved April 5, 2004, from CERT Coordination Center Web Site: http://www.cert.org/stats/#incidents

Cohen, F. (2003 July 18). *Risk management: Concepts and frameworks* (ISSN 1048-4620). Midvale, UT: Burton Group.

Cohen, F., Phillips, C., Swiler, L. P., Gaylor, T., Leary, P., Rupley, F., et al. (1998, September). *A preliminary classification scheme for information system threats, attacks, and defenses; A cause and effect model; and some analysis based on that model.* Retrieved September 4, 2003, from U.S. Department of Energy Web Site: http://www.all.net/journal/ntb/cause-and-effect.html

CORAS. (2003). *CORAS, a platform for risk analysis of security critical systems*. Retrieved September 25, 2003, from IST-2000-25031 Web Site Web Site: http://www.nr.no/coras/

Daley, K., Larson, R., & Dawkins, J. (2002). A structural framework for modeling multi-stage network attacks. *proceedings of the international conference on parallel processing workshops, 1530-2016/02*.

Dimitrakos, T., Raptis, D., Ritchie, B., & Stolen, K. (Eds.). (2000). *Model based security risk analysis for web applications, the CORAS approach.* United Kingdom: CORAS.

Durfee, E. H., Clement, B., & Pappachan, P. (2000, February). *Multilevel coordination mechanisms for real-time autonomous agents*. Ann Arbor, MI: University of Michigan.

Durrett, J. R. (2003, January). *Threat modeling and risk management*. Retrieved October 10, 2003, from Texas Tech University Web Site: http://jdurrett.ba.ttu.edu/6342/Notes/ThreatModeling.ppt

Elliott, J. B. (Ed.). (1998). *Risk analysis.* Booth Scientific, Incorporated Hendersonville, NC: The Validation Consultant.

Ellison, R. J., & Moore, A. P. (2003, March). *Trustworthy refinement through intrusion-aware design (TRIAD)* (CMU/SEI-2003-RT-002). Pittsburg, PA: Carnegie Mellon University.

Ellison, R. J., & Moore, A. P. (Eds.). (2001). *Architectural refinement for the design of survivable systems.* Pittsburg, PA: Carnegie Mellon University.

Ericson, C. A. (Ed.). (1999). *Fault tree analysis - a history.* The Boeing Company; Seattle, Washington: Proceedings of the 17th International System Safety Conference.

Expert Choice. (2003). *Advanced decision support software & services for better, faster, more justifiable decisions*. Retrieved September 25, 2003, from http://www.expertchoice.com/

Franklin, J.L. & Thrasher, J.H. (1976). *An introduction to program evaluation.* New York: John Wiley & Sons.

Fullwood, R. R., & Hall, R. E. (Eds.). (1988). *Probabilistic risk assessment in the nuclear power industry.* London: Pergamon Press.

Fumy, W., De Soete, M., Humphreys, T., Ohlin, M., & Conveners, W. G. (2003 August 10). *Information technology - security techniques - A framework for IT security assurance - Part 1 Overview and framework model* (ISO/IEC DTR 15443-1). Berlin, Germany: International Organization for Standardization.

Gan, C., & Scharf, E. (2003). *Building an experience factory for a model-based risk analysis framework.* Retrieved October 24, 2003, from Department of Electronic Engineering, University of London Web Site: http://wm2003.aifb.uni-karlsruhe.de/workshop/w06/GWEM2003-slides%20CW%20Gan.pdf

Gomolski, B. (Ed.). (2003). *Gartner 2003 IT spending and staffing survey Results.* Stamford, CT: Gartner Group.

Harrington, H. J., & Anderson, L. C. (Eds.). (1999). *Reliability simplified, going beyond quality to keep customers for life.* New York: McGraw-Hill.

Helmer, G., Wong, J., Slagell, M., Honavar, V., Miller, L., & Lutz, R. (Eds.). (2000). *A software fault tree approach to requirements analysis of an intrusion detection system.* Ames, IA: Iowa State University.

Henley, E. J., & Kumamoto, H. (Eds.). (1992). *Probabilistic risk assessment.* Washington, DC: IEEE Press.

Herzog, A., & Shahmehri, N. (Eds.). (2001). *Towards secure e-services: Risk analysis of a home automation service.* Linkopings University: Department of Computer and Information Science.

Howard, J. D. (1997). *An analysis of security incidents on the Internet 1989-1995 (PhD thesis).* Carnegie Mellon University, Carnegie Institute of Technology

Huang, G. Q., Shi, J., & Mak, K. L. (Eds.). (1999). *Failure mode and effect analysis (FMEA) over the WWW.* Hong Kong: Department of Industrial and Manufacturing Systems Engineering, The University of Hong Kong.

IEC (2000). *Functional safety of electrical/electronic/programmable safety related systems.* (61508).

Internet Software Consortium. (2005, April). *Internet domain survey, Jan 2005, number of hosts advertised in the DNS.* Retrieved April 5, 2005, from Internet Software Consortium Web Site: http://www.isc.org/index.pl?/ops/ds/

ISO/IEC (2001). *Information technology - Guidelines for the management of IT Security - Part 1: Concepts and models for IT Security* (TR 133335-1). Geneva, Switzerland: International Organization for Standardization.

ISO/IEC 10746 (1995). *Basic reference model for open distributed computing*. Geneva, Switzerland: International Organization for Standardization.

ISO/IEC 17799 (2000). *Information technology - Code of practice for information security management*. Geneva, Switzerland: International Organization for Standardization.

Jacobson, I., Rumbaugh, J., & Booch, G. (1999). *The unified software development process.* New York: Addison-Wesley.

Jacobson, I., Rumbaugh, J., & Booch, G. (2000). *Universal modeling language distilled* (2nd ed.). New York: Addison-Wesley.

Kabay, M. E. (Ed.). (1996). *Enterprise security: Protecting information assets*. New York: McGraw-Hill.

Keong, T. H. (1997). *Risk analysis methodologies*. Retrieved March 4, 04, from Pacific Internet Web Site: http://home1.pacific.net.sg/~thk/risk.html

Knott, C. L., & St. James, M. (Eds.). (2002). *An analytic hierarchy process methodology solution for celebrity endorser decisions in marketing.* The George Washington University: Department of Management Science.

Krsul, I. V. (1998). *Software vulnerability analysis (PhD thesis).* West Lafayette, IN: Purdue University.

Lawrence, J. D. (1995, October). *Software safety hazard analysis* (UCRL-ID-122514). Berkeley, CA: U.S. Nuclear Regulatory Commission, University of California.

Leiner, B. M., Cerf, V. G., Clark, D. D., Kahn, R. E., Kleinrock, K., & Lynch, D. C. (2003). *A brief history of the Internet*. Retrieved October 10, 2003, from Internet Society Web Site: http://www.isoc.org/internet/history/brief.shtml#JCRL62

Little, R. J. A. & Rubin, D. B. (1897) *Statistical analysis with missing data.* New York, Wiley.

McDermid, J. A., et al. (1995). Experience with the application of HAZOP to computer-based systems. *COMPASS '95. Proceeding of the 10th Annual Conference on Computer Assurance,* 37-48.

Menon, A., & Varadarajan, P. R. (1992). A model of marketing knowledge use within firms. *Journal of Marketing, 56 (October),* 53-71.

Menon, A., & Wilcox, J. (1994). *USER: A scale to measure use of market research: Technical working paper* (Number 94-108). Cambridge, Massachusetts: Marketing Science Institute.

Mertens, D. M. (Ed.). (1998). *Research methods in education and psychology: Integrating diversity with quantitative and qualitative approaches.* Thousand Oaks, CA: Sage.

Microsoft. (2003). *Access office access*. Retrieved October 23, 2003, from Microsoft Office Product Suite Web Site: http://www.microsoft.com/office/access/prodinfo/default.mspx

Microsoft. (2004). *Microsoft office on-line Excel home page*. Retrieved April 20, 2004, from SPSS Web Site: www.microsoft.com/office/excel/default.asp

Microsoft. (2003). *Threat modeling*. Retrieved October 10, 2003, from Microsoft Developers Network Web Site: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnnetsec/html/THCMCh03.asp

Moberg, F. (Ed.). (2001). *Security analysis of an information system using an attack tree-based methodology (masters thesis).* Goteborg: Chalmers University of Technology.

Moore, A. P., Ellison, R. J., & Linger, R. C. (2001). *Attack modeling for information security and survivability* (CMU/SE-2001-TN-001). Pittsburg, PA: Carnegie Mellon University.

Pfleeger, C. P. (Ed.). (1997). *Security in computing.* Upper Saddle River, NJ: Prentice Hall.

Putt, A., & Springer, J. (1989). *Policy research: Concepts, methods, and application.* New Jersey: Prentice Hall.

Raafat, H. (2002). *Hazard analysis and risk assessment - risk assessment methodologies* (RT 2002). Aston University: Mechanical Engineering and Product Design, Health and Safety Unit.

Rajgopal, J., & Mazumdar, M. (2002). Modular operational rest plans for inferences on software reliability based on a Markov model. *IEEE Transactions of Software Engineering, 28*(4), 353-363.

Rasmussen, N. C. (Ed.). (1975). *Reactor safety study: An assessment of accidents risks in US commercial nuclear power plants.* Washington, D.C.: Nuclear Regulatory Commission.

Rea, L. M., & Parker, R. A. (Eds.). (1997). *Designing and conducting survey research: A comprehensive guide* (2nd ed.). San Francisco: Jossey-Bass.

Robertson, L. (2003). *The WTC was designed to survive the impact of a Boeing 767, so why didn't it?* Retrieved April 5, 2004, from Vancouver Independent Media Center Web Site: http://vancouver.indymedia.org/news/2003/07/56715_comment.php

Rosall, J. (2002). *E-commerce software market forecast and trends, 2002-2006.* Stamford, CT: Gartner Group.

Salter, C., Saydjari, O. S., Schneier, B., & Wallner, J. (1998). *Toward a secure engineering methodology.* : National Security Agency.

Saunders, J. H. (1994, July). *A comparison of decision accuracy in AHP and point allocation.* In (Ed.) Third Intl Symposium on the Analytic Hierarch Process: Washington D.C.

Saunders, J. H. (2000). *A risk management methodology for information security: The analytic hierarchy process*. Retrieved September 12, 2003, from Information Resources Management College Web Site: http://www.johnsaunders.com/papers/risk-ahp/risk-ahp.htm

Sawma, V. D. (Ed.). (2002). *A new methodology for deriving effective countermeasures design models (master thesis).* Ottawa, Ontario, Canada: School of Information Technology and Engineering, University of Ottawa.

Schneier, B. (1999). Attack trees, Modeling security threats. *Dr. Dobb's Journal, December*, 21-29.

Schneier, B. (2000). Attack trees. In C. Long (Ed.), *Secrets & lies, digital security in a networked world* (pp. 318-333). New York: John Wiley & Sons, Incorporated.

Selliah, S. (Ed.). (2001). *Mobile agent based attack resistant architecture for distributed intrusion detection system (masters thesis).* West Virginia: College of Engineering and Mineral Resources, West Virginia University.

Sheatsley, P. B. (1983). Questionnaire construction and item writing. In P. H. Rossi, J. D. Wright, & A. B. Anderson (Eds.), Handbook of survey research (pp. 195-230). San Diego, CA: Academic Press.

Singleton, R. A., & Straits, B. C. (Eds.). (1999). *Approaches to social research* (3rd ed.). New York: Oxford University Press.

Smith, S. P., & Harrison, M. D. (Eds.). (2003). *Supporting reuse in hazard analysis.* York, United Kingdom: The Dependability Interdisciplinary Research Collaboration, Department of Computer Science.

Sproull, N.L. (1995). Handbook of research methods: a guide for practitioners and students in the social sciences, (2nd ed.). Metuchen: Scarecrow Press.

SPSS. (2004). *SPSS survey and market research*. Retrieved April 20, 2004, from SPSS Web Site: http://www.spss.com/vertical_markets/survmkt_research/

Stolen, K., Braber, F. d., Dimitrakos, T., Fredriken, R., Gran, B. A., Houmb, S., Lund, M. S., Stamatiou, Y. C., & Aagedal, J. O. (2002). *Model-based risk assessment - the CORAS approach* (). :CORAS consortium.

Storey, N. (Ed.). (1996). *Safety-critical computer systems.* Upper Saddle River, NJ: Addison-Wesley.

Sudman, S., & Bradburn, N. D. (1982). *Asking questions: A practical guide to questionnaire design.* San Francisco: Jossey-Bass.

SurveyMonkey.com. (2004). *SurveyMonkey.com because knowledge is everything*. Retrieved April 20, 2004, from Surveymonkey.com Web Site: http://www.surveymonkey.com

Sutton, I. S. (Ed.). (1992). *Process reliability and risk management.* New York: Van Nostrand Reinhold.

Thunstrom, C., & Ahs, M. (Eds.). (2003). *Security analysis of a system connected to a future Network Based Defense. (masters thesis).* Gothenburg: Chalmers University of Technology.

Tidwell, T., Larson, R., Fitch, K., & Hale, J. (2001). Modeling Internet Attacks. *Proceedings of the 2001 IEEE workshop on information assurance and security, United States Military Academy, West Point NY, 5-6 June 2001,* 54-59.

Varner, P. E. (2001, May 11). *Vote early, vote often, and VoteHere: A security analysis of VoteHere*. Retrieved September 5, 2003, from University of Virginia Web Site: http://www.cs.virginia.edu/~pev5b/writing/academic/thesis/

Vidalis, S., & Jones, A. (2003, June). *Using vulnerability trees for decision making in threat assessment* (CS-03-2). Wales, United Kingdom: School of Computing, University of Glamorgan.

Viescas, J. (Ed.). (1999). *Running Microsoft Access 2000.* Redmond, WA: Microsoft Press.

Vincoli, J. W. (Ed.). (1994). *Basic guide to accident investigation and loss control.* New York: Van Nostrand Reinhold.

Webopedia. (2003). *SQL*. Retrieved October 23, 2003, from http://www.webopedia.com/TERM/S/SQL.html

Welch, I., Warne, J., Ryan, P., & Stroud, R. (2003 February 3). *Malicious and accidental fault tolerance for Internet applications, architectural analysis of MAFTIA's intrusion tolerance capabilities* (Project IST-1999-11583). Australia: The University of Newcastle.

White, D. (1995). Application of system thinking to risk management: a review of the literature. *Management Decision, 33*(10), 35-45.

Winzip. (2004). *The zip file utility for Windows*. Retrieved April 20, 2004, from Winzip Web Site: http://www.winzip.com

Witty, R., Dubiel, J., Girard, J., Graff, J., Hallawell, A., Hildreth, B., et al. (2001 June 8). *The price of information security* (R-11-6534). Stamford CT: Gartner Group.

APPENDIX A: Pre-Assessment Survey

PARTICIPANT PRE-ASSESSMENT SURVEY

Directions: Please circle the appropriate number indicating your opinion toward each of the following statements. The textual responses represent a Boolean response of either Yes = agree or No = disagree.

Familiarity Attack Trees:

1. I am familiar with the term attack trees.	No	Yes
2. I am familiar with the attack tree methodology.	No	Yes
3. I have created an attack tree.	No	Yes
4. I understand attack trees well enough to create an attack tree.	No	Yes
5. I have used attack trees as a risk methodology.	No	Yes
6. We currently have a process to identify systems vulnerabilities.	No	Yes
7. An attack tree is a useful tool when identifying security vulnerabilities.	No	Yes
8. Attack tree analysis is a useful tool.	No	Yes

Comments on Attack Trees:

Cost Benefit:

9. We currently have a process to identify prioritization of countermeasures from a costing perspective.	No	Yes
10. We currently have a process to identify the most effective allocation of funds offering the highest rate of return on security vulnerabilities.	No	Yes
11. We currently are considering incorporating attack tree analysis to assist with budgetary decisions as related to the allocation of funds for security.	No	Yes
12. I believe that attack trees analysis can be a useful process used to assist with budgetary decisions as related to the allocation of funds for security.	No	Yes

Comments on Costing:

Probability:

13. We currently have a process to help identify prioritization of countermeasures from a human resource allocation. No Yes

14. We currently have a process to identify the most effective allocation of human resources offering the highest rate of return on security vulnerabilities. No Yes

15. We currently are considering incorporating attack tree analysis to assist with staffing assignment decisions as related to the allocation of human resources. No Yes

16. I believe that attack trees analysis can be a useful process use to assist with staffing assignment decisions as related to the allocation of human resources. No Yes

Comments on Probability:

Structured Query Language (SQL) Program:

17. Our current process used to identify security cost benefit analysis is automated. No Yes

18. Our current process used to identify security human resource allocation is automated. No Yes

19. Attack tree analysis using a structured query language database program is capable of pruning attack trees scenarios. No Yes

20. I believe that attack tree analysis can be a useful process when incorporated into a SQL program. No Yes

Comments on SQL Program:

APPENDIX B: Post-Assessment Survey

PARTICIPANT POST-ASSESSMENT SURVEY

Directions: Please circle the appropriate number indicating your opinion of each of the following statements.

The textual responses represent a Boolean response of either Yes = agree or No = disagree.

The numerical responses represent a continuum in which 1 = disagree strongly and 5 = agree strongly.

Familiarity Attack Trees:

1. I am familiar with the term attack trees.	No	Yes
2. I am familiar with the attack tree methodology.	No	Yes
3. I have created an attack tree.	No	Yes
4. I understand attack trees well enough to create an attack tree.	No	Yes
5. I have used attack trees as a risk methodology.	No	Yes
6. We currently have a process to identify systems vulnerabilities.	No	Yes
7. An attack tree is a useful tool when identifying security vulnerabilities.	No	Yes
8. Attack tree analysis is a useful tool.	No	Yes

Cost Benefit:

9. We currently have a process to identify prioritization of countermeasures from a costing perspective.	No	Yes
10. We currently have a process to identify the most effective allocation of funds offering the highest rate of return on security vulnerabilities.	No	Yes
11. We currently are considering incorporating attack tree analysis to assist with budgetary decisions as related to the allocation of funds for security.	No	Yes
12. I believe that attack trees analysis can be a useful process used to assist with budgetary decisions as related to the allocation of funds for security.	No	Yes

	Disagree....Agree
13. Attack trees can be used to identify the protection cost of a system	1 2 3 4 5
14. Attack trees cannot be used to identify the vulnerability cost of a system.	1 2 3 4 5
15. Attack trees are an effective decision tool to be used in cost benefit analysis.	1 2 3 4 5
16. I was able to receive cost benefit decision-making value while using attack tree.	1 2 3 4 5
17. Attack trees are an ineffective decision tool to be used in cost benefit analysis.	1 2 3 4 5

Probability:

18. We currently have a process to help identify prioritization of countermeasures from a human resource allocation.	No	Yes
19. We currently have a process to identify effective allocation of human resources offering the highest rate of return on security vulnerabilities.	No	Yes
20. We currently are considering incorporating attack tree analysis to assist with staffing assignment decisions as related to the allocation of human resources.	No	Yes
21. I believe that attack trees analysis can be a useful process used to assist with staffing assignment decisions as related to the allocation of human resources.	No	Yes

	Disagree....Agree
22. Attack trees can be used to identify the protection probability of a system.	1 2 3 4 5
23. Attack trees cannot be used to identify the vulnerability probability of a system.	1 2 3 4 5
24. Attack trees are an effective decision tool to be used in a probability analysis.	1 2 3 4 5
25. I was able to receive probability decision-making value while using attack tree analysis.	1 2 3 4 5
26. Attack trees are an ineffective decision tool to be used in probability analysis.	1 2 3 4 5

Structured Query Language (SQL) Program:

27. Our current process used to identify security cost benefit analysis is automated.	No	Yes
28. Our current process used to identify security human resource allocation is automated.	No	Yes
29. I believe that attack tree analysis can be a useful process when incorporated into a SQL program.	No	Yes

30.	Attack tree analysis using a structured query language database program is capable of pruning attack trees scenarios.	No Yes
		Disagree....Agree
31.	Attack tree analysis using a structured query language database program is an effective process capable of incorporating "what-if" scenarios.	1 2 3 4 5
32.	The automated attack tree analysis program is an effective process capable of assisting with vulnerability risk assessment.	1 2 3 4 5
33.	The automated attack tree analysis program is an effective process capable of assisting with cost analysis of security decisions.	1 2 3 4 5
34.	The automated attack tree analysis program is an effective process capable of assisting with human resource allocation of security decisions.	1 2 3 4 5
35.	I was able to receive decision-making value while using attack tree SQL program.	1 2 3 4 5

Concluding Questions:

36. What aspects of the attack tree analysis costing model were not helpful?
37. What aspects of the attack tree analysis probability model were not helpful?
38. What aspects of the attack tree analysis SQL program were not helpful?
39. What recommendations do you have for improving attack tree analysis?
40. How did attack tree analysis compare with any previous risk assessment methodology experience(s) you may have had?
41. Any additional comments/recommendations:

APPENDIX C: Consent Form

Consent Form
An Evaluation of Attack Tree Analysis Using a SQL Based Simulation.
Walden University

You are invited to participate in a research study pertaining to attack tree analysis. You were selected as a possible participant because of your knowledge and/or experience related to the topic. Please read this form and ask any questions you may have before accepting the invitation to participate in this study.

This study is being conducted by: Michael S. Pallos, a doctoral candidate at Walden University.

Background Information:

The purpose of this study is to research the effectiveness of attack trees incorporated into an information system computer security risk assessment methodology. This research will explore the effectiveness of using attack trees to assist with costing decisions, probability analysis, and to explore the viability of using attack trees in order to identify additional penetration points in systems which may be exploited by terrorists or attackers who were not considered in the initial design of a system.

This research seeks to answer the following questions:

1: How effectively might the inclusion of attack tree analysis be incorporated into a cost analysis model capable of assisting information systems managers with budgetary decisions?

2: How effectively might the inclusion of attack tree analysis be incorporated into a probability model capable of assisting information systems managers with human resource allocation?

3: How effectively might the inclusion of a Structured Query Language (SQL) database program be implemented to simplify the use of a cost analysis model and a probability model to assist information systems managers with costing and human resource allocation decisions?

Procedures:

If you agree to participate in this study, you will be asked to do the following:

1. Complete a pretest survey (approximately a ten minute investment).
2. Work with an SQL program, supplied to you fully populated with the attack trees built. (The time you invest in working with the program is at your discretion.)
3. Complete a posttest survey (approximately a twenty minute investment).

Voluntary Nature of the Study:

Your participation in this study is strictly voluntary. Your decision whether or not to participate will not affect your current or future relations with this researcher or Walden University. If you initially decide to participate, you are still free to withdraw later without affecting those relationships.

Risks and Benefits of Being in the Study:

There are no known risks associated with this study.

The benefits of participation may include the satisfaction of being part of an academic study that researched the validation of attack tree analysis used in costing analysis, probability analysis, and the creation of a SQL program to assist with the analysis.

In the event you experience stress or anxiety during your participation in the study you may terminate your participation at any time. You may refuse to answer any questions you consider invasive or stressful.

Compensation:

No compensation will be provided for this research study.

Confidentiality:

The records of this study will be kept private. In any report of this study that might be published, the researcher will not include any information that will make it possible to identify a participant. Research records will be kept in a locked file; only the researcher will have access to the records.

Contacts and Questions:

The researcher conducting this study is Michael S. Pallos. His adviser is Dr. Pamela Wilson. You may direct any questions you have of either of them. If you have questions later, you may address them to Michael S. Pallos, MPallos@waldenu.edu, (863) 709-1611 (Eastern Time Zone), 1235 Brighton Way, Lakeland FL 33813, and Dr. Pamela Wilson, Pwilson2@waldenu.edu , (321) 724-8997 (Eastern Time Zone), 1612 Glendale Ave. NW, Palm Bay, Florida 32907. The Research Participant Advocate at Walden University is Dale Good. You may contact him at 1-800-925-3368, x 1210 if you have questions about your participation in this study

You may keep a copy of this consent form.

Statement of Consent:

If you agree to participate in this survey, click on the link below and fill out the pretest survey form. *(SurveyMonkey.com link to pre-survey was added here.)*

CURRICULUM VITAE

Michael S. Pallos
1235 Brighton Way
Lakeland, Florida 33813 USA
MPallos@gte.net
(863) 709-1611 (ET)

Summary Statement

I have an extensive and varied background in the information technology (IT) and other industries. This extensive IT background and practical business experience combined with an M.B.A and Ph.D., including dissertation research on computer security, provides me with a unique blend of skills needed to be an effective teacher, mentor, and/or practitioner. I am a published author and continue to be a featured speaker at industry conferences as well as consultant to some of the largest corporations in the world.

Education

Current (ABT), Walden University, School of Management, Minneapolis, MN
Ph.D., Applied Management and Decision Science, Information Systems Management
Thesis, An Evaluation of Attack Tree Analysis Using a SQL Based Simulation.

1999, Nova Southeastern University, Fort Lauderdale, FL
M.B.A., School of Business and Entrepreneurship.

1997, Massachusetts Institute of Technology, Cambridge, MA
Summer Program, Data and Models: Theory and Computer Practice

1996, Nova Southeastern University, Fort Lauderdale, FL
B.S. Professional Management

1988, St. Petersburg Junior College, Clearwater, FL
A.S. Microcomputer Applications

Education Experience

I earned my master degree (MBA) through a distance learning program. As a result, I have an appreciation for the distant student's challenges of balancing work, family, and studies. My observations with distance learning is that most students have a

full time job, heavy responsibilities, and fall within the thirty-five to forty-five year-old age range. Often these individuals are directors and managers looking to advance to an executive level.

Professional Experience

2004-Present, IBM Corporation, World Wide
Senior Solution/Software Architect

I joined IBM as part of IBM's acquisition of Candle Corporation in June of 2004. During this transition to IBM, I continued to support the Federal Sales team retaining my interactions with senior customer executives CIO/CTO/COO, to support the delivery, sales, and pre-sales force with complex integration efforts. I have continued to expand my responsibilities by volunteering as the security architect and Internet integration architectural resource for my technical peers in the Americas. As a technical resource, some of the technologies used in the projects I am involved in are architecture, design, and development and include Service Oriented Architecture (SOA), Component Based Development, Security, EJB, J2EE, WebSphere Application Server (WAS), XML, SOAP, Java, C++, C, .NET, CORBA, DCE, MQ-Series, MQSI, DB2, UDB, and SQL using UML and Case technologies.

1999-2004, Candle Corporation, World Wide
Senior Solution/Software Architect

As a Senior Solution Architect, I am responsible for presentations/interactions with senior customer representatives, CIO/CTO, and supporting the sales force with the Federal Sales team in 2003. During this time, Candle created a new role known as the Franchise Owner. The Franchise Owner retained all responsibilities within a specific region. My region was the federal government and my responsibilities include business development director, consulting manager, project manager, solution architect, and mentoring of sales forces. In an effort to establish technical and business credibility, I have published in journals multiple times each year and am often a featured speaker at industry conferences, such as, WebSphere Technical Conference, IBM's Transaction and Messaging, SHARE, and Gartner ITXpo.

I became the senior solution architect responsible for assisting in the development, training, and mentoring of enterprise architectures throughout the Americans. These job responsibilities include the internal training and mentoring of software architects internationally, developing, and presenting courses to internal and external client's world wide. While supporting the architects thought the Americas, I continued to support the sales team in large scale enterprise architectures. For my efforts I received *Consultant of the Quarter* 2002.

As a Solution Architect, I am responsible for presentations/interactions with senior customer representatives, CIO/CTO, and supporting the sales force with EAI (enterprise application integration) efforts. Within the sales team, I am the technical leader that architects enterprise solutions providing application integration in diverse verticals. I also assume the role of Project Manager and Architect responsible for the creation and implementation of multiple simultaneous projects averaging over twenty companies per year. I have architected and implemented numerous proof-of-concepts, enterprise migration plans, phase-out plans, asynchronous architectures, RFPs, and third party integration (such as SAP / PeopleSoft / Oracle Financials). I also participated in the development of internal methodologies and the development of marketing, sales and educational programs for Roma®, eBusniess Platform™ and CASP™ product lines. I received *America's South East Consultant of the Quarter* 2000.

1995-1999, Computer Task Group (CTG), Incorporated, National Delivery Team, USA
Principal Consultant, Enterprise Architect

As a Principal for BankBoston, assuming Chief Enterprise Architect responsibilities, I led the Redesign Architecture Subteam (RAS). This twenty person team's primary responsibilities included rearchitecting BankBoston's infrastructure by adhering to market segment alignment, while leveraging legacy systems and maximizing reuse. The effort included project planning and budgetary constraints totaling 400+ million dollars. The infrastructure contained a component-based CORBA/COM backbone.

As a Principal Consultant for the Archdiocese of Newark, I led the Archdiocese in the application selection and integration process for the Fund Management Systems. The process followed the Joint Technology Selection (JTS) methodology. For my efforts, I received a letter of commendation from the Secretary of Development.

As a Principal Consultant for Signet, I led the architecture team in defining and prioritizing business drivers/requirements for a large reengineering effort as Signet planned their move from a legacy environment to distributed client/server architecture. I conducted a series of facilitated sessions and provided recommendations and costing information for architectural components such as Middleware, Customer Support/Service Desk software packages, development tools, and various financial applications.

As a Principal Consultant and Enterprise Architect for Xerox, I was a member of the Xerox Global Security Architecture team. The team provided the guidelines, foundation, and template for Xerox's global security architecture.

As an Enterprise Architect for The Vanguard Group, my responsibilities included middleware architecture and prototyping of recommended technologies defining tactical and strategic middleware direction for The Vanguard Group.

Within CTG, I was instrumental in chartering the National Internet Virtual Team. This team complemented CTG's Internetworking team and grew Internet related skills internally for the company in this emerging technology. I led a series of research and development efforts designed to educate CTG staff on various Internet products and new Internet development environments.

As a Program Manger for GTE TSI, I led the Clone Detector 4.0 team to an on-time within budget completion of this aggressive project.

As an Enterprise Architect for Ramsey County Sheriff's Department, my responsibilities included the creation of the Enterprise Model which established the architectural foundation for the migration of diverse legacy systems to Client/Server technology. This model provided the foundation for the application and technical architectures, including the integration of all agencies involved in criminal justice. Additional responsibilities included Client/Server Architect focusing on the Technology, Network and Application Architectures. The project included the development of a single application that allowed for seamless deployment into a heterogeneous environment, while adhering to specific federal and state application development and security mandates.

As a Client/Server Architect for ISI Systems, I was responsible for directing ISI through the transition to a Client/Server infrastructure. The project was critical to the business, requiring a proof-of-concept to be conducted before committing to an approach and specific technology. The proof-of-concept consisted of creating a presentation layer foundation front-ending six legacy systems. An industry standard *look-and-feel* was developed allowing for a new comprehensive system capable of communicating and sharing data among the legacy systems in a non-invasive manner.

1993-1995, Titan Client/Server Technologies, Tampa, Florida
Program Manager, Project Leader, Client/Server Architect

Client/Server Architect member of architecture OO (object oriented) design team for IBM CBSS project. Design philosophies include a Client/Server Architecture utilizing OSF's DCE (open standards foundation, distributed computing environment) approaching a CORBA (common object request broker architecture). Essentially this project encompasses the creation of IBM's own internet with a forecasted customer base of 3-4 million client users. Incorporated DCE's Directory (XDS CDS & X.500), File,

and Security Services utilizing RPCs (remote procedure calls) and Threads transported via ATMs.

I was the Program Manger who led a team in the design/development of System Services, an integral part of GTE TELOPS process re-engineering infrastructure. These OO-based (object oriented), cross-platform (e.g. Windows, UNIX, mainframe) Client/Server applications consisted of the PrintService, FileTransferService, and a Kerberos-based SecurityService. Tools used: UNIX C++, Visual C++, HP Soft Bench, xdb, Windows, Motif, RCS, ESQL/C, AWK, JCL, and Rational Rose Design tools. Of special note was the PrintService. This service generated JCL (job control language) wrappers for the mainframe print jobs prior to transmission. C Bindings were created for all services to allow for C applications use, as well as C++. For my efforts, I received the GTE IRON MAN award March 1995 and the IC Markets Team Award in November 1994.

I was the project leader for the Access Customer Gateway (ACG) REPAIR life-cycle (new development and production) Client/Server application. This program provided front-end functionality for a mainframe-based Trouble Administration System (TAS). I managed a staff of 12 engineers and performed job estimation, scheduling, resource management, interface management with customers and other software groups, conducted project reviews and personnel recruiting. I was the senior technical resource responsible for integration and systems administration of the REPAIR application.

I also developed a peer-to-peer interface between two diverse systems, one UNIX and the other MVS. The interface was created using an EDI (electronic data interchange) product that utilized remote procedures calls (RPCs).

Finally, acting as configuration manager, I completely automated the application integration environment that was responsible for application builds and application code distribution to test and production systems. The previous process to build and distribute the application code took three days to completely process. Upon implementation of the new processes I created, the time frame was reduced from three days down to one hour. For my efforts, I received the Quest for Quality award in December of 1993. The tools used to implement the new process I designed included: AWK, SCCS, Make, Informix, EDA, and TCP/IP.

1991-1993, Time Customer Service, Inc. Tampa, Florida
Project Leader/Senior Systems Analyst, Designer

I was the technical project leader on the *Letter Generation System* for Time Warner's periodical publication system. The system drove mainframe printers capable of generating over fourteen pieces of mail per week to Time Warner customers. I was the

lead architect and project head whose responsibilities included architecting, designing, developing, and leading the project team. Specifics of the system included a distributive client/server -based UNIX network (Macintosh, PC, IBM 3090, HP, and SUN Sparc) integrated for the purpose of mailing generation and distribution. The project also included a smaller database evaluation project that created internal benchmarks using Oracle, Sybase, and Informix to determine optimum product for system.

1989-1991, ASCOM/Timeplex, Inc. Clearwater, Florida
Network Product Specialist, Systems Analyst, Developer

Timeplex had an elite group of network WAN (wide area network) specialists responsible for complex WAN designs and implementations. The group offered additional services of custom programming to work with the WAN specialist providing customers with the unique benefit of custom software development. I was the person responsible for overseeing all aspects of the customer software development. For many projects I would lead a team of subcontractors, product vendors, and in-house technical resources to produce the required customer focused solution. Software development included OOA (object oriented analysis) / OOD (object oriented design), C/C++, UNIX, AWK, shell scripts (Bourne, C, Korn), Informix and C Tree.

Highlighted software development projects included:

- Union Carbide: I was involved in the development of a wide-area network reporting system in which the network data was retrieved using shell and AWK scripts. Data was tokenized, parsed, and then stored in a relational database. Once stored in the relational database the information was accessible on-line and for inclusion in reports. The system was developed in C++, Motif and Informix. Reports were formatted via a Postscript *black-box.*
- Ford Motor Company: Development of a trouble ticket reporting system using shell and AWK scripts with ESQL/C.
- Johnson & Johnson: Development of a CASH (Cost Allocation System Hub), a telecommunications charge-back system.
- Prudential-Bache: Development of an auto-reconfiguration of wide-area network for video-teleconferencing.

1987-1995, Accelerated Software, Inc. Bay Area, Florida
Proprietor

During evenings and weekends in the late eighties, I designed, developed, and successfully marketed MedSoft III©/Medico Mas®, a physician billing and patient tracking system. These systems are a multi-user, multi-language system that manages patient/treatment histories, produces multiple reports, analytical data and generates HCFA 1500-based electronic bills to Medicare and other insurance clearing houses. During the early nineties, working with a civil engineering firm, I developed another

application, BaseCoat™, a complex coating system. This system was marketed to customers such as Disney, Shell Oil, and Texaco. When reaching a point of critical mass with the time commitment required maintaining these systems, I chose to sell the copyrights and maintain my day job. Both products are still being sold as of 2004.

1987-1989, Comtech Systems, Inc. Tampa, Florida
Software Engineer

As a local consultant for Comtech Systems, I participated on numerous projects for various customers in the Tampa Bay area. Customers included numerous projects for GTEDS (GTE Data Systems) and GTEIS (GTE Information Systems), Conservoc, and Lighthouse Motors. Highlighted projects included a master translator engine for documentation, electronic dental forms processing system, EDI-based system for the exchange of manufacturing and order data and the production of reports, electronic medical claims clearing system, and an alarm processing system for a traffic data collection system. Additional projects included the development of a workman's compensation data processing system and an automotive dealership accounting and maintenance data tracking system. Systems were developed using the following technologies: C/C++, Motif, EDI, 4GL, Report Writer, ESQL/C, Foxpro, and Clipper.

1985-1987, ACE Beauty Company, Inc. Largo, Florida
Analyst Programmer

My first full time computer programmer job directly out of college involved being the single “computer guy” for a beauty supply company. This regional company included a warehouse and sixteen regional sales stores. During this timeframe, three outside consultants were enlisted creating a four person team that developed an on-line point of sale and inventory tracking system. The inventory data was collected at each store with a MSI handheld data collection terminal, and then submitted to a central location, the company warehouse. My responsibilities included managing the team while obtaining business requirements, architecting, designing, programming, developing user documentation, and installing all hardware and software. As the sole IT employee of Ace Beauty Company, I led the team; however, I was mentored by the more experienced senior consultants. This role provided an excellent information technology (IT) foundation and the development of a “can do” attitude since most electrical devices became my responsibilities. Tasks performed at Ace included all aspects of IT such as drilling holes in concrete which allowed me to pull computer cables, to all aspects of technical support for the stores, network configuration, LAN (local area network) administrator, and software developer Technologies used to implement systems include the programming languages C, DBXL, QuickSilver, and Clipper operated on a Novell LAN.

Copyrights

1994 BASE.Coat,
BASE.Coat is a maintenance coating software program used by civil engineers. The development of this program was a joint venture between my partner and a civil engineering firm. The civil engineering firm supplied all of the coating business knowledge, while I supplied the technical program managerial skills. The completed program was sold to the civil engineering firm in 1995.

1992, Medico Mas III,
A commercial physicians billing system. (English/Spanish). The third iteration of MedSoft contained so many enhancements that an independent copyright was obtained. The copyright and program with all rights was sold to a company from Miami after nine years of ownership in 1998 (Copyright sale included the MedSoft application also).

1989, MedSoft, A commercial physicians billing system. (English/Spanish).
MedSoft is a physicians billing system that was wholly created by myself. The project began as a physician friend of mine asked me to develop a system to replace the manual paper peg-board system in his office. After development of the English version, I was approached by a company from Miami to create a Spanish version of the system to be sold in Miami and Central/South America.

Publications

Pallos, M. S. (2004). Using the Thread Pool Funnel to Optimize WebSphere Application Server Performance. WebSphere Advisor, September, 26-28

Pallos, M. S. (2004). Attack Trees: It's a Jungle Out There. WebSphere, February, 12-14.

Pallos, M. S. (2003). WebSphere Application Server and Database Performance Tuning, Part II. WebSphere Developers Journal, June, 16-18.

Pallos, M. S. (2003). WebSphere Application Server and Database Performance Tuning, Part I. WebSphere Developers Journal, May, 28-37.

Pallos, M. S. (2002). Best Practices for Better WebSphere Performance. WebSphere Advisor, Nov/Dec, 22-25.

Pallos, M. S. (2002). Service Oriented Architecture, International Business & Management Research Conference, Honolulu, HI

Pallos, M. S. (2001). Service Oriented Architecture: A Primer. EAI Journal, December, 32-35.

Pallos, M. S. (2001). Component-Based Development with MQSeries Workflow. Business Integrator Journal, Summer, 22-26.

Whitepapers

2003, WebSphere Application Server (WAS) and Database Performance Tuning, Candle Corporation

2002, Designed WebSphere Applications using Best Practices, Candle Corporation

2001, Service Oriented Architecture (SOA), A Primer for Information Technology and Business Management, Candle Corporation

1999, COM & CORBA Interoperability, Candle Corporation

1998, Common Object Request Broker Architecture (CORBA) 2.0, Computer Task Group (CTG)

1996, Distributed Computing Environment (DCE); A Standard to Watch, Computer Task Group (CTG) Achievement Forum

Papers Presented

2004, SHARE Exchange, New York, NY
I presented the content of my paper on best practices for software development including the persistence layer. This presentation was highly technical in nature focusing on the Internet distribution channel connecting to a corporation's backend system. This presentation offered developers' eighteen best practices that have been proven to increase processing speed in Internet applications.

2003, WAS Users Group Central Conference, Detroit, MI
I presented the content of my paper on best practices for software development including the persistence layer. This presentation was highly technical in nature focusing on the Internet distribution channel connecting to a corporation's backend system. The presentation offered developers' eighteen best practices that have been proven to increase processing speed in Internet applications.

2003, IBM MQ & CICS Conference, Las Vegas, NV

The content of the presentation included the implementation of a theory presented by IBM. The MDB (message driven beans) concept was rather new, and sound implementation details were lacking. I proposed a methodology used to implement MDB and the WebSphere Application Server paradigm.

2003, SHARE Exchange, Dallas, TX
WebSphere Application Server was somewhat new to the market place and implementations were struggling with slow processing speed. The persistence layer was identified as on area of optimization that, if properly implemented, would drastically increase the systems processing time. I authored a paper on performance optimization to the persistence layer based on best practices that software developers may implement. The paper was presented at the SHARE conference.

2002, WAS Canadian Users Group, Calgary, Canada
The company I worked for, Candle Corporation, was attempting to gain creditability with developers in the middleware market specifically WebSphere developers. I authored a paper that includes eighteen of the best practices developers could implement to increase the applications processing time. This paper was presented at multiple technical conferences.

2002, International Business & Management Research Conference, Honolulu, HI
Service Oriented Architecture (SOA) was a new concept introduced in 2001. Building a prototype application, Loan Application Demo, I was able to implement the concepts of SOA. Using the Loan Application Demo as a baseline for demonstration, I authored a whitepaper and published an article as a SOA Primer for Management. This speaking engagement was the presentation of that paper.

2002, GartnerGroup ITxpo, Orlando, FL
See the 2002 WAS Canadian Users Group above

2002, WebSphere Advisor, Baltimore, MD
See the 2002 WAS Canadian Users Group above

2002, IBM WebSphere Technical Exchange, Las Vegas, NV
See the 2002 WAS Canadian Users Group above

Research Interests

My research interests are in the area of information technology security, specifically, a risk assessment methodology known as attack tree analysis. Attack tree analysis, created by Schneier (1999; 2000), is a risk assessment methodology used to identify system vulnerabilities and penetration points of a system. Attack trees describe the security or

vulnerability of a system based upon the goals of the attacker. A hierarchical representation of the attack goal is created building a tree containing nodes (or leafs) which represent each penetration point of a system. These nodes also provide a location to implement countermeasures. Countermeasures are the processes implemented to secure each respective node. Values can also be assigned at the node level, such as costing and probability, allowing analysis to be performed on the attack tree.

Awards

2006, National Registers' Who's Who in Executives and Professionals 2005–2006 edition
Due to my accomplishments as a practitioner providing consultative services to some of IBM and Candle Corporations largest customers in addition to frequently speaking at industry conferences, the National Registers' Who's Who in Executives and Professional included me in the 2005 – 2006 edition.

2004, IBM, Team of the Quarter for Federal CMS Project in 4th Quarter 2004
I was the Performance and Available Architect member for the federal government project with the Center for Medicare and Medicaid Services (CMS). For our exceptional efforts, the entire IBM team received the Team of Quarter award in December of 2004.

2002, Candle Corporation, Consulting and Services Contributor of the Quarter.
I became the senior solution architect responsible for assisting in the development, training, and mentoring of enterprise architectures throughout the Americas. These job responsibilities included the internal training and mentoring of software architects internationally, developing, and presenting courses to internal and external clients world wide. While supporting the architects thought the Americas, I continued to support the sales team in large scale enterprise architectures. For my efforts as the architect to the architects I received the Contributor of the Quarter.

2000, Candle Corporation, Consulting and Services Contributor of the Quarter
Candle offered a middleware integration product that lacked effective marketing. This product offered the foundation to implement a new concept known as Service Oriented Architecture. I proposed a solution to build a comprehensive demo, the Loan Application Demo, which was a full blown product, implementing the SOA paradigm, built using a Candles tool that lacked marketing. This international effort was completed with a two-month time frame as I worked in America, England, and Italy. For my efforts I received Consultant & Services Contributor of the quarter for 2000, and Candle received a new marketing tool.

1998, CTG, Archdiocese of Newark, Letter of Commendation from the Secretary of Development

As Principal Consultant I worked on a Joint Technology Assessment, JTA, for the Archdiocese of Newark. This effort included a business assessment of the Archdiocese fund management needs, the creation of a report card to be used for scaling, the evaluation of candidates which met the Archdiocese profile, and the implementation of the final selection. For my efforts I received a Letter of Commendation for the Secretary of Development.

1995, GTE IRON MAN award, March
Working as a Program Manager and Object Oriented Architect as a consultant to GTE (Verizon) corporation I was awarded the GTE IRON MAN award. This award was one for which all of my employees, peers, and executives voted.

1994, GTE IC Markets Team Award, November
As project leader for a major project for GTE, my team delivered a multimillion dollar project under budget and on schedule. For our accomplishments we each received the IC Markets Team Award.

1993, Titan Client/Server Technologies, Quest for Quality award, December.
As configuration manager of the ACG Repair System, I optimized the application build process. The optimization resulted in a reduction of the build process from three days down to one hour. This substantial savings resulted in my earning the Quest for Quality award for Titan due to process improvement.

Certifications / Additional Training

IBM Certified On Demand Business Solution Advisor
IBM Certified WebSphere MQ Solution Expert
IBM Certified e-Business Solution Designer
IBM Certified e-Business Solution Expert
IBM Trained Enterprise Java Bean, Architecture
IBM Trained J2EE, Java, Java Beans, Java Server Pages, VisualAge
IBM Trained WebSphere Application Server
IBM Trained CICS, z/OS

Skills

Applications:	Microsoft Office, Visio, Microsoft Project, Lotus SmartSuite
Architectural:	Enterprise architecture planning, design, development, integration, configuration management (source code control), testing, application development, methodology development, process flow improvement, infrastructure reengineering, JAD (joint application development) facilitation, project management, systems design, right-sizing, multi-tier, legacy integration.
Business:	Change management, customer value, managing organizational and operational systems, entrepreneurial and strategic thinking, managing organizational behavior, migration planning, organizational change, process improvement/optimization, six sigma, strategic planning, total quality management, value based leadership, vision creation.
Computer Languages:	C, C++, C#, COBOL, FORTRAN, HTML, Java, JavaScript, Pascal, SOAP, Visual Basic (VB), XML
Development Tools:	JBuilder, WSAD (WebSphere Studio Application Developer), Visual Café, Microsoft Visual Studio/Tools, FrontPage, VisualAge for Java, Rational Rose, System Architecture, STP (Software Through Pictures), CASE tools, SCCS, PowerBuilder, .NET
Framework:	Enterprise Java Beans (J2EE), Design Patterns, Distributed Computing Environment (DCE), Common Object Request Broker Architecture (CORBA) Web Services, Service Oriented Architecture (SOA), WebSphere (WBI/WAS)
Methodologies:	Software Development Life Cycle (SDLC), Microsoft Solution Architecture, IBM GSMethod, Rational Unified Process, Object Oriented, TeAMethod, Component Based, Client/Server
Operating Systems:	UNIX, AIX, HP-UX, Linux, SUN Solaris, OS/390, z/OS, Windows NT/2000/XP
RDBMS:	Access, dBase, Clipper, QuickSilver, Informix, Sybase, Oracle, UDB, DB2, SQL Server
Shell Scripts	AWK, Bourne, C, DOS, Korn, T