International Journal of Information Management 28 (2008) 483–491

Contents lists available at ScienceDirect

International Journal of Information Management

journal homepage: www.elsevier.com/locate/ijinfomgt

Quantitatively assessing the vulnerability of critical information systems: A new method for evaluating security enhancements

Sandip C. Patel [a,*], James H. Graham [b], Patricia A.S. Ralston [b]

[a] Department of Information Science and Systems, Graves School of Business and Management, Morgan State University, Baltimore, MD 21251, USA
[b] J.B. Speed School of Engineering, University of Louisville, Louisville, KY 40292, USA

ARTICLE INFO

Article history:

Keywords:
Information security
Risk analysis
Information-security measurement
Security threats
Vulnerability measurement

ABSTRACT

This paper proposes a new approach for assessing the organization's vulnerability to information-security breaches. Although much research has been done on qualitative approaches, the literature on numerical approaches to quantify information-security risk is scarce. This paper suggests a method to quantify risk in terms of a numeric value or "degree of cybersecurity". To help quantitatively measure the level of cybersecurity for a computer-based information system, we present two indices, the threat-impact index and the cyber-vulnerability index, based on vulnerability trees. By calculating and comparing the indices for various possible security enhancements, managers can select the best security enhancement choice, prioritize the choices by their relative effectiveness, and statistically justify spending resources on the selected choice. By qualifying information security quantitatively, the method can also help managers establish a specific target of security level that they can track.

We illustrate the use of the proposed methodology on the security of supervisory control and data acquisition (SCADA) systems using data from the SCADA system test bed implemented at the University of Louisville as a case study, and then show the use of the proposed indices on this information system before and after two security enhancements.

© 2008 Elsevier Ltd. All rights reserved.

1. Introduction

1.1. Risk assessment

Risk assessment is the first, and usually the most difficult and error prone, step in the risk management process. Much research has been done on estimating the risk resulting from damages to the physical assets. This assessment is commonly performed by estimating two quantities, which are the magnitude of the potential financial loss L, and the probability p that the loss will occur. The risk R is calculated by the following formulae, where i refers to each incident that could result in a loss:

$$R_i = L_i p(L_i),$$
$$R_{\text{total}} = \Sigma[L_i p(L_i)]^i.$$

Estimating both of these key parameters—L and p, is quite difficult because of sparse historical data and also because of uncertainty in the measurements. To make things more difficult, risk management involves multiple metrics because of the interrelationship between these two quantities. In other words, a risk with a high probability and small potential loss is often treated differently than the reverse case.

Assessing the financial losses resulting from information-security attacks adds more complexity to the already challenging risk assessment procedure described above. An example of an attack on an information system, referred to as a cyber attack, is the denial of service attack, where an unauthorized intruder prevents a component of an information system, such as a server, from providing services by sending it a very large number of spurious messages. Formal risk assessment of information security serves two purposes: to identify existing weaknesses in the systems, and to cost justify and prioritize the cost of additional safeguards (Department of Homeland Security, 2006). However, probably because of the challenges listed above, there is limited amount of work on quantitatively assessing the damage done by cyber attacks. A literature review revealed that although a number of articles discussed qualitative approaches, the literature on numerical approaches to quantify information-security risk was scarce.

1.2. Information-system risk assessment

The research aim of this paper is to propose a new method to measure financial risk computer-based information systems face

* Corresponding author. Tel.: +10.164430.168850.163836.

0268-4012/$ – see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijinfomgt.2008.01.009

from cyber attacks by measuring the risk in terms of a numeric value or "degree of security". Such a method can help assess how much security is improved if a specific security enhancement is applied. This approach can help managers select between various enhancement choices, prioritize them by their relative effectiveness by measuring the improvement in the proposed degree-of-security indices, and make cost justifications. By qualifying security quantitatively, the method can also help set and track a specific numeric target for security level (Iheagwara, 2003; Mell, Scarfone, & Romanosky, 2006).

We apply the proposed method on a supervisory control and data acquisition (SCADA) system. SCADA systems control many critical national infrastructure components, including electrical power systems, water treatment and delivery systems, oil and gas pipelines, and much of the chemical and process industry. A successful cyber attack on these information systems could cause significant physical damage, economic damage, and social disruption (Peltier, 2001). A case study illustrates how the proposed methodology is used to validate two security improvements to a SCADA system test bed that is implemented at the University of Louisville, Kentucky, USA.

The second section of this paper presents a brief background on risk assessment approaches, followed by the presentation of the proposed risk management approach in Section 3. Section 4 presents results from a case study using this method for SCADA information systems. Section 5 presents conclusions and directions for future research.

2. Background

2.1. Literature review

Risk-assessment methods can be categorized into two groups—qualitative methods and quantitative methods. Qualitative risk analysis methods are often selected by managers who believe their risk-assessment calculations are simple; therefore, it is not necessary to quantify threat frequency, hence many non-technical issues are easily accounted for (Farahmand, Navathe, Sharp, & Enslow, 2005). Farahmand, Navathe, Enslow, and Sharp (2003), present a systematic qualitative approach to overall risk assessment for information systems. Especially helpful is their development of a three-axis view of the threat space, which organizes the problem of risk management and the presentation of a procedure for computing losses due to threats and benefits of countermeasures. Whitman and Mattord (2005) specify four factors: likelihood of the vulnerability, value of assets, current controls, and uncertainty that are used in quantitative risk analysis. These authors define the risk as "the likelihood of the occurrence of a vulnerability multiplied by the value of the information asset minus the percent of risk mitigated by current controls plus the uncertainty of current knowledge of the vulnerability".

While evaluating investments on security enhancements, specific quantitative methods are used. General cost benefit analysis (CBA) methods compare the cost of deploying the security to the benefits. Whitman and Mattord (2005) define the CBA methods as those that determine whether or not the control alternative being evaluated is worth the associated cost incurred to control the specific vulnerability. Benefits are derived from prevented losses that could have resulted from a successful attack and include avoiding financial losses such as lost production and idle manpower. A variation to the basic CBA approach takes into account market-value losses and uses an event-study analysis. For example, using market valuations, Cavusoglu, Mishra, and Raghunathan (2004b) conducted a study to assess the impact of security breaches on the market value of breached firms. Firms with compromised security, on average, lost approximately 2.1% of their market values within 2 days (Cavusoglu et al., 2004b). Campbell, Gordon, Loeb, and Zhou (2003) examine the economic effects of security-breach reports by studying the stock market prices of publicly traded US companies. The authors focus on security breaches related to formal computer-based information systems, as opposed to informal communication systems. Broader studies, such as that by Dhillon (1997), examine information security breaches by in-depth case studies of individual organizations.

Financial impacts on capital market have also been studied for a specific type of attack. For example, Ettredge and Richardson (2002) measure the effects of security breaches on capital markets by investigating the stock market reaction to the February 2000 DOS attacks. Using the event-study methodology to study the period of 4.5 years, Hovav and D'Arcy (2003) report on the market-value impact on the companies that were compromised by the DOS attack. Using the same methodology to study data of 15 years, Hovav and D'Arcy (2004) assess the impact of virus attack announcements on the market value of affected companies.

The third approach to evaluating security alternatives, in addition to the CBA and the event-study analysis, is to use risk analysis framework models. This approach identifies and uses potential losses and their likelihoods for the calculations. Various models have been proposed, for example, the hierarchical holographic model (HHM; Longstaff, Chittister, Pethia, & Haimes, 2000) identifies and coordinates multiple, complementary decompositions of a complex system. Cavusoglu, Mishra, and Raghunathan (2004a) propose an analytical model that considers IT security technologies used for a typical security infrastructure made of preventive, detective, and response measures. Using calculus, Gordon and Loeb (2002) propose a model that considers how the vulnerability of information and the loss associated with such vulnerability affect the optimal level of resources that should be devoted to securing information. The model assumes that there is a single threat to an information set.

Our model offers several benefits over the current methods. First, it takes into account all known attacks launched in the past instead of a specific attack. The model also considers each specific attack and associates it with vulnerabilities that lead to the attack. Thus, the approach is more specific than the general formula provided by studies such as those of Whitman and Mattord (2005). Thirdly, as opposed to previous work that targets commercial firms, our work illustrates calculating the risk assessment for critical infrastructures and the proposed method can easily be generalized for most other information systems. Additionally, our work uses actual dollar amounts that could be potentially lost if an attack was successful. Finally, the proposed work provides indices that facilitate prioritizing various security options and tracking the progress made towards securing the system.

2.2. Risk assessment of critical systems

We provide a more detailed literature review of risk assessment methods for SCADA, other distributed control systems (DCS), and critical infrastructure systems in Ralston, Graham, and Patel (2006). Miller and Byres (2005) point out, and we concur, that many papers discussing vulnerabilities of control systems neglect the articulation of relative risk of particular implementations. Miller and Byres (2005) suggest that all resources that need protection and the vulnerabilities that can become threats must be identified. The proposed approach considers all known threats and vulnerabilities in an infrastructure implementation. We provide a short summary of the current research in risk assessment of critical systems.

The methods for assessing risk of critical infrastructures include the operationally critical threat, asset, and vulnerability evaluation (OCTAVE; Alberts, Dorofee, Stevens, & Woody, 2003) which is a framework for identifying and managing information-security risks developed at the Carnegie Mellon University, and the CORAS (Aagedal et al., 2002). CORAS is a tool-supported methodology for model-based risk analysis of security-critical systems. A Sandia National Laboratories report (Campbell & Stamp, 2004) attempts to classify risk assessment methods, according to the level of detail and approach in order for users to be able to select the most appropriate method. Leontief input–output models (Rinaldi, 2004) provide an aggregated, time-independent analysis of generation, flow, and consumption of commodities among infrastructure sectors. Such modeling and simulation abilities are integral to infrastructure risk analyses. Haimes, Kaplan, and Lambert (2002) describe a risk filtering, ranking, and management method (RFRM), which builds on the HHM (Haimes & Chittester, 2005) to identify risks, but then filters and ranks the risks so that the risks can be addressed in the order of priority. Haimes and Chittester (2005) use inoperability input–output modeling (IIM) to quantify economic losses and their propagation through the various economic sectors for large-scale civil infrastructures controlled by SCADA systems over internet protocol communication networks. The work by Nozick, Turnquist, Jones, Davis, and Lawton (2005) focuses on representing interdependent infrastructure networks using Markov and semi-Markov processes to reflect uncertain capacity on network links.

2.3. Proposed-methodology background

Stoneburner (2006) proposes the unified risk framework that provides a common language for risk management that may be used for describing risks. Quantitative risk analysis methods fall under the broad category of probabilistic risk assessment (PRA). A generally accepted definition of PRA is a systematic and comprehensive methodology to evaluate risks associated with a complex engineered technological entity. Although PRA technically includes a risk identification phase, it does not provide the guidance of methods like HHM, but rather assumes that the designer can identify the risks. PRA includes all fault/attack tree analyses (FTA), attack tree analysis (Schneier, 1999), event tree analysis (ETA), vulnerability tree analysis (Vidalis & Jones, 2003), failure mode and effect analysis (FMEA), or failure mode effect and criticality analysis (FMECA; Ayyub, 2003), and cause/consequence analysis (CCA; Pumfrey, 1999), as well as methods that use directed graphs and logic diagrams (Henley & Kumamoto, 1996). Most other methods are extensions or combinations of these. Many of the tools mentioned earlier incorporate these methods to varying degrees. To perform the fundamental security analysis of computer-based information systems, reasonings using Dijkstra's weakest precondition can be investigated. Yasinsac and Childs (2001) have done some initial work in this direction for general Internet security, which can be applied to information management systems.

The proposed method, referred to as the augmented vulnerability tree method, combines the concepts of vulnerability tree analysis (Vidalis & Jones, 2003), fault tree analysis (Ayyub, 2003), attack tree analysis (Schneier, 1999), and the cause–consequence technology, developed by the Risø Laboratories (Pumfrey, 1999). All of the tree-analysis methods are very similar, but they differ primarily in what defines the top event. That is, the fault tree/attack tree/vulnerability tree method is a deductive process, where the topmost undesirable event is first postulated. Then, the ways for this event to happen are deduced. The deduction process results in a tree that includes all failure components that could contribute in causing the top event. Thus, a vulnerability tree is a logical model representing the logic of system failures qualitatively. A tree diagram is often constructed as a graphical illustration showing the stepwise cause resolution using formal logic symbols.

The evaluation of a tree may be done qualitatively or quantitatively (also know as probabilistic evaluation). During a qualitative evaluation, the tree is rearranged for computational effectiveness. To perform this logical rearrangement, fault tree cut sets are determined using Boolean mathematics for events. Various approaches, including top-down, bottom-up, modularization, and Monte-Carlo simulations, may be used for determining cut sets. In the quantitative approach, evaluation measures such as the probability of the occurrence of the top events, and reliability or unreliability of the top events are calculated.

3. Proposed risk assessment methodology

In this section, two indices representing the vulnerability of an information system are presented. The threat-impact index is a value between 0 and 100 showing the financial impact of a probable cyber threat. The lower the value, the smaller the impact from a successful cyber attack. The cyber-vulnerability (CV) index, also a value between 0 and 100, represents system vulnerability to cyber attacks. That is, this index is a numerical value representing system flaws or undesirable events that would help an intruder launch attacks. A lower index value represents a more secure system because the fewer the options an intruder has, the smaller the possibility of a successful attack. The following subsections describe the calculation of the proposed indices.

3.1. Step 1: construct the augmented vulnerability tree

A logic block diagram, called an augmented vulnerability tree, is constructed in a manner similar to a fault tree. To construct the augmented vulnerability tree, a top undesirable event is first postulated; it represents a pivotal event for a particular failure scenario. The possible means, referred to as attacks, for the top event to occur are systematically deduced. In other words, the attacks are the events that cause the top event to occur. The top event would require one or more attacks, connected using notations similar to decision trees, to compromise the system. As shown in Figs. 1 and 2, an attack is represented using a rectangular box. Any one of the attacks connected is sufficient for the top event to occur.

Each attack is then meticulously analyzed to deduce system flaws, referred to as base events, which can cause an attack. Each attack has one or more base events. A small solid circle is used to draw possible vulnerability paths, just as the decision tree draws possible outcomes. As shown in Figs. 1 and 2, a base event is represented using a hollow circle. The base events can include one or more vulnerabilities that help an intruder to launch an attack. These vulnerabilities are listed inside the hollow circle.

3.2. Step 2: construct damage analysis table and calculate threat-impact indices

The construction of the damage analysis table requires determining its columns and rows, and calculating the table-cell values. A list of all the attacks from the augmented vulnerability tree comprises the rows of the table. The first set of columns comprises a list of the financial damage each attack from the augmented vulnerability can cause. After the list is completed, it is combined to create the normalized list by categorizing the related damages together and removing duplicate list items. Appended to the first set of columns are columns for total financial damages, the probability of each attach, financial risk, and the normalized damages.

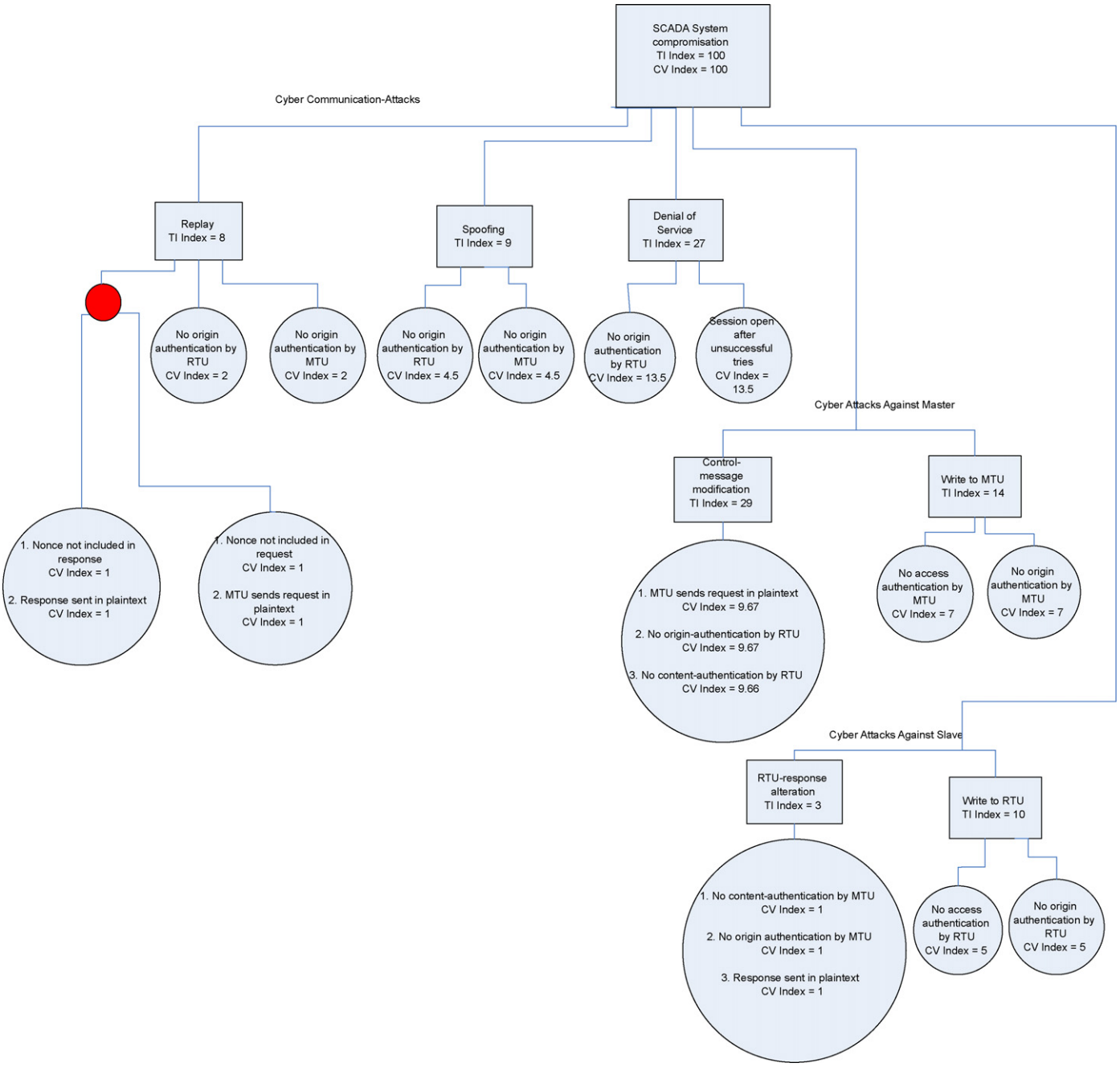


Fig. 1. Augmented vulnerability tree with threat-impact and cyber-vulnerability indices.

The table values for the financial damage for each attack are calculated by interviewing the operators, engineers, accountants, and managers. The probability of each attack is estimated using the frequency of attack taken from the history/logs or by using methods such as those described by Cohen (1999). If and when a new attack is launched, it would be logged and the attack probabilities will need to be modified slightly with an additional attack. The risk values are calculated by multiplying the probability by the total damage for each row. This multiplication gives long-term risk values, since frequent small-damage attacks could lead to bigger total losses in the long run and vice versa. In the last column, the risks are normalized by dividing each row by the total risk so that each risk value ranges from 0 to 100, and referred as the threat-impact values.

3.3. Step 3: add threat-impact index values to augmented vulnerability tree

A threat-impact index is a proposed attempt to quantify the impact of a cyber threat. The threat-impact values, taken from the last column of the damage analysis table, are marked on each attack on the augmented vulnerability tree. The total of all the threat-impact indices represents the threat-impact index for the entire information system, which is listed for the top event. A system without any implemented security has the threat-impact index of 100. After security enhancements are applied, this value should be reduced if one or more attacks are eliminated by the enhancement.

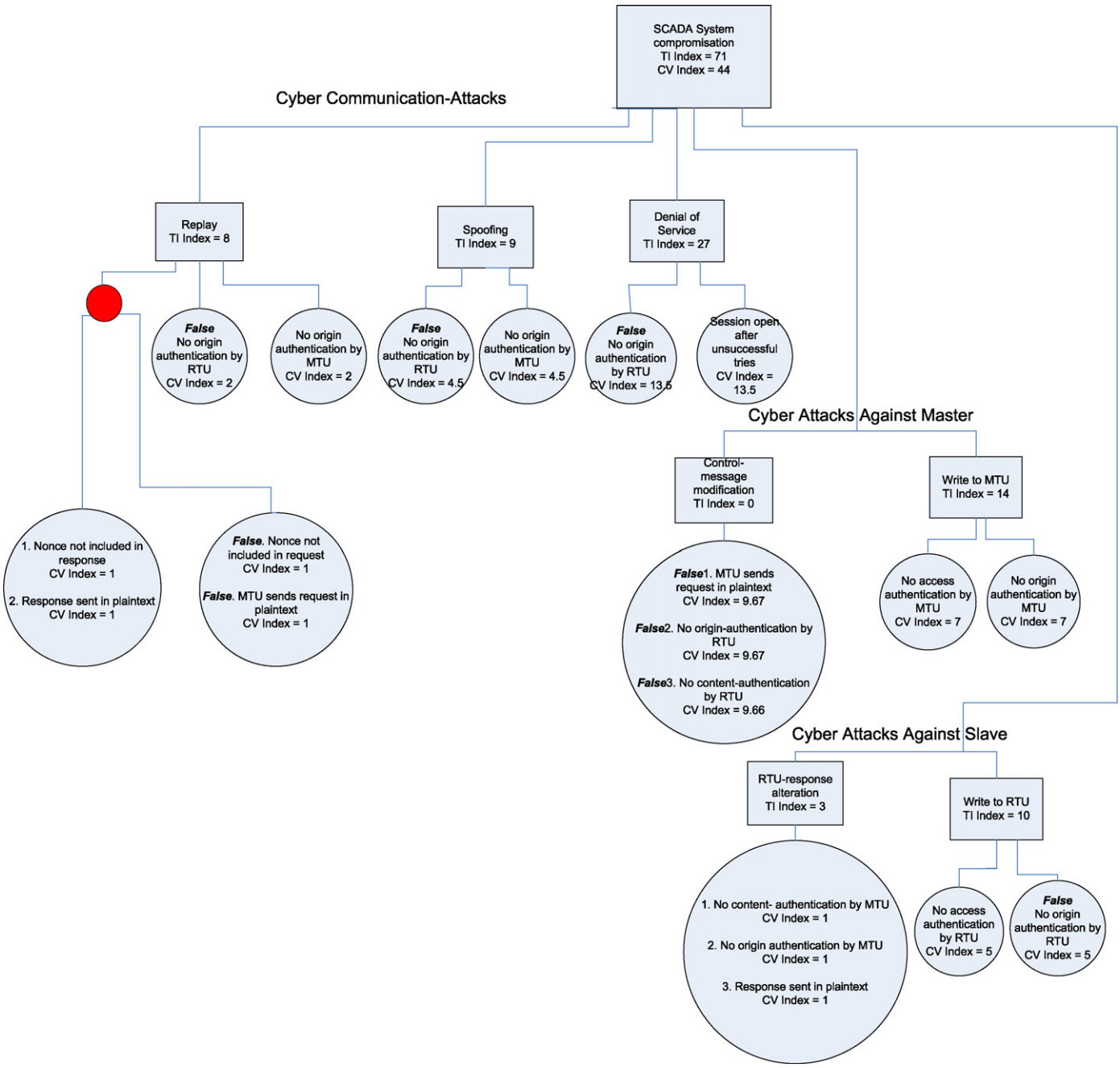


Fig. 2. Revised augmented vulnerability tree for a security-enhanced system.

3.4. Step 4: calculate the cyber-vulnerability index values

The CV indices are assigned to the base events by using the threat-impact index of the attack under which the events are displayed. That is, base events get their index from the parent node in an augmented vulnerability tree. The threat-impact index is equally divided among all the base events at the same level. Dividing the parent-level values ensures that each security enhancement results in an improved CV index of the information system, even if the enhancement fails to improve the system's threat-impact index. Some security enhancements fail to improve the threat-impact index of the system when the enhancement does not eradicate an attack altogether, because one or more ways to launch the attack still remains available to the intruder. However, such enhancements do reduce the chances of a successful attack, since they limits options available to an intruder.

3.5. Step 5: complete augmented vulnerability tree by adding cyber-vulnerability indices

Once all the base events have the CV indices assigned to them, the CV index for the entire tree is calculated by summing all the CV indices. This index value represents the CV index of the information system. After the completion of this step, the augmented vulnerability tree is complete. The proposed method uses the probability—derived from the past attack records—of each type of attack to get the damage value for that type of attack. These damage values are eventually used to assign a CV index to

each vulnerability. Since a CV index is based on probabilities, the index represents damages caused by the vulnerability in the long run.

3.6. Step 6: repeat steps 2 through 5 for proposed security enhancements

When an enhancement is ready to be evaluated, new values of the system's threat-impact index and the CV index are calculated using the tree marked with changes created by the enhancement.

4. Case study

This section describes the configuration for the SCADA test bed at the University of Louisville and illustrates how the proposed method was applied to the system component that performs information communication. The method is also illustrated on the same component after two security enhancements were applied to the protocols used for information communication.

4.1. Analysis of vulnerabilities in information communication

The research objective of the SCADA test bed at the Intelligent Systems Research and Chemical Engineering Laboratories at the University of Louisville is to evaluate operation of a SCADA system under carefully monitored and safe operating conditions. The specific control objective is to regulate the fluid level in a multiple tank and pump system through supervisory control. Details on the process operation are available online (Patel, Tantalean, Ralston & Graham, 2005). The communication component of this test bed sends the control data entered by engineers to a remote terminal unit (RTU) and sends real-time parameter values from the RTU to the master terminal unit (MTU). These message exchanges are done via software programs, written in Java, and use a subset of the Distributed Network Protocol version 3 specifications. The component uses a client–server architecture where the MTU acts as a client and the RTU acts as a server. The server waits for any requests, such as writing to a file, from the client. The details of the architecture have been presented (Patel, 2006; Patel, Graham, Ralston, & Tantalean, 2005).

4.1.1. Step 1: construct the augmented vulnerability tree

The cyber communication attacks listed in the tree are the general cyber attacks, which can be launched against both the MTU and RTU. Hence, the base events for these attacks refer to both the MTU and the RTU. The attacks against the master are the specific attacks that can be launched against the MTU. The proposed approach divides all the attacks into three categories: attacks against the master, attack against a slave, and attacks on the communication between the master and the slave. The possible attacks against a master include an intruder trying to manipulate the MTU itself (hardware or software) or trying to modify the information used by the MTU by modifying or removing a control message. Similarly, possible attacks against an RTU can be categorized as RTU-response alteration or writing to an RTU. The possible attacks on the cyber communication could be categorized as an intruder trying to take a valid message and resending (replay attack) it, trying to send a fake message by spoofing, or trying to disable the communication (DOS attack). Attacks such as eavesdropping (unauthorized interception of information in transit through the use of methods other than wiretapping), traffic analysis (intercepting and examining messages to deduce information from patterns in communication), and man-in-the-middle (gaining information by sniffing or tapping a line between two unsuspecting parties) are excluded from the augmented vulnerability tree diagram since they are passive forms of attacks which do not directly impact SCADA security.

Table 1
Attack descriptions

Replay	Capture message and resend later
Spoofing	Pretend to be an MTU or RTU
Denial of service	Overwhelm system with a very large number of spurious messages to disable it.
Control message modification	Capture and modify a control message
Write to MTU	Add or modify files on MTU
RTU response alteration	Modify response and send to MTU
Write to RTU	Add or modify values on RTU

Fig. 1 shows the base events that include the flaws such as nonce not included in the response. A response is the messages sent from an RTU in reference to a request message from an MTU. A nonce, marked in one of the base events, is a value that is different every time a message is sent. Plaintext refers to readable information that is in its normal form, which anyone or any system can read, as opposed to its encrypted form. The base event marked as "MTU sends request in plaintext" implies that the entire message is sent in plaintext. The origin authentication refers to a security measure designed to establish the validity of the originator of a message. A session is an established connection between an RTU and an MTU. The content authentication refers to a security measure designed to establish the validity of the content of a message. An access authentication is a security measure that validates the identity of the person who attempts to use the system.

4.1.2. Step 2: construct damage analysis table and calculate threat-impact values

The attacks from Table 1 are listed as rows in the damage analysis table. To generate columns, lists showing all possible damages resulting from each attack are produced. The damages are grouped and duplicates are removed. The first eight columns of Table 2 show the resulting damages. The loss of control or view is the financial damage caused while trying to run the process without the full use of the SCADA control system. This cost includes the extra operator hours required to run the process manually. Loss of production includes the process shutdown costs and material lost. Loss of staff time is the loss resulting from idle personnel, and is calculated by multiplying the number of operators, their idle hours, and their salary per hour. Equipment damage or loss is the amount of damage done to the process equipment. The cost of remedying intrusion includes fixing software problems and process equipments.

The average damage values are used to calculate the dollar amount. The damages are calculated using various resources systematically. First, case studies and other literature on critical-system attacks were studied. Specifically, we estimated the damages using the data in Rakaczky (2005). For the readers interested in using a large volume of data on damages, we recommend the British Columbia Institute of Technology Industrial Security Incident Database (Byres and Lowe, 2004), which provides one of the most complete incident databases for cyber-attack records on critical systems. Next, we interviewed the SCADA industrialists to find a typical range of damages they encountered or estimated. Using their feedback, considering the costs of various components of the SCADA test bed, and finally conducting several meetings among ourselves to scrutinize the cost values using our experience with the critical systems, we arrived at the final damage values listed in Table 2. A similar process was followed to estimate the probabilities.

Table 2
Damage analysis table

	Loss of control or view ($)	Loss of production ($)	Loss of staff time (operator salary)				Equipment damage/loss ($)	Cost of remedying intrusion (people cost, $)	Total damage ($)	Prob.	Prob × damage ($)	Normalized damage (previous-column values÷previous column total, %)
			# Op	# Hours (h)	Pay/h ($)	Total ($)						
Replay	1000	150.16000	1	40	35	1400	0	200.16000	370.16400	0.2	7480	8
Spoofing	4000	1000.16000	1	80	35	2800	250.16000	350.16000	1660.16800	0.05	8340	9
Denial of service	6000	1000.16000	2	50	35	3500	5000	150.16000	1290.16500	0.2	250.16900	27
Control message modification	4000	4000.16000	5	120	35	210.16000	750.16000	500.16000	5500.16000	0.05	270.16500	29
Write to MTU	0	200.16000	1	40	35	1400	250.16000	200.16000	660.16400	0.2	130.16280	14
RTU-response alteration	1000	150.16000	1	40	35	1400	0	100.16000	270.16400	0.1	2740	3
Write to RTU	0	200.16000	1	40	35	1400	5000	200.16000	460.16400	0.2	9280	10
Total		–	–				–	–	–	1	940.16520	100

4.1.3. Step 3: enhance the augmented vulnerability tree with the threat-impact index values

The last column of the damage analysis table shows the threat-impact index values for each attack. These values are marked on the augmented vulnerability tree. The total of all these values adds to 100 in this tree since no security enhancement has been applied yet.

4.1.4. Step 4: calculate the cyber-vulnerability index values

The threat-impact index of each attack is now subdivided among the base events. For example, as shown in Fig. 2, a threat-impact index of 27 for the denial of service attack is divided equally into two of its base events, with each event having an index of 13.5. Similarly, since the control-message modification attack has three events, one third of this attack's threat-impact index value is assigned to each of its base events. As noted earlier, we assume a simple, unweighted distribution, and divide the index equally among base events. Therefore, the threat-impact index for the replay attack is divided into two at the first level, giving the CV index of 4 to the base events of no-origin authentication. The other two events are at a lower level and hence each of these base events is assigned the CV index of 2.

4.1.5. Step 5: complete the augmented vulnerability tree

The CV index of the system is represented by the top event in the augmented vulnerability tree. This value, which is 100, is the total of the CV indices of all the base events listed as the following. Since the same base event could be listed under more than one attacks, it is helpful to construct a list such as the following, which can later be used to quickly calculate the new CV index of the system.

- Nonce not included in response: one base event with a CV index of 1.
- Response sent in plaintext: one base event with a CV index of 1 and another base event with a CV index of 1, giving a total of 2.
- Nonce not included in request: one base event with a CV index of 1.
- MTU sends request in plaintext: one base event with a CV index of 1 and another with 9.67, giving a total of 10.67.
- No-origin authentication by RTU: one base event with a CV index of 4.5, one with 2, one with 13.5, one with a CV index of 9.67, and one with a CV index of 5, giving a total of 34.67.
- No-origin authentication by MTU: one base event with a CV index of 4.5, one with 2, one with 1, and one with 7, giving a total of 14.5.
- Session open after unsuccessful tries: one base event with a CV index of 13.5.
- No-content authentication by RTU: one base event with a CV index of 9.66.
- No access authentication by MTU: one base event with a CV index of 7.
- No-access authentication by RTU: one base event with a CV index of 5.
- No-content authentication by MTU: one base event with a CV index of 1.

The complete augmented vulnerability tree is shown in Fig. 1. The cyber-vulnerability indices are marked as CV indices and the threat-impact indices are marked as TI indices.

4.2. Security enhancements

Two SCADA communication security enhancements were proposed (Patel, 2006) and implemented on the test bed described above. These enhancements include the Authentication Octets and challenge–response authentications. The Authentication Octet

enhancement is based on the digital-signature algorithm. In this enhancement, additional bytes of information, referred to here as Authentication Octets, are appended to each message from an MTU to an RTU for authentication. First, the MTU performs hash on the selected bytes of the message it intends to send, which results in generating a hash digest. The MTU encrypts this digest by using its private key and then the MTU sends the encrypted digest along with the message. The message itself is not encrypted to save processing time during the encryption. Other data that vary with time are time stamped and sent as nonce for additional checks. When the RTU receives the message, it decrypts the hash digest by using the MTU's public key and compares this digest with its own calculated digest. The RTU concludes that the message came from an authentic source if it can decrypt the digest using the MTU's private key. The RTU also concludes that the message contents are unaltered by an intruder if the two digests match. In the augmented vulnerability tree, the Authentication Octet enhancement makes the base events of nonce not included in the request, no-origin authentication by RTU and no-content authentication by RTU *false*. Since a part of the message is sent in encrypted form, the base event of the MTU sending request in a plaintext is also marked as *false* in the new tree.

The challenge–response authentication verifies the identity of an MTU by using the challenge–response cryptography. An RTU initiates a challenge to authenticate an MTU that shares a secret. The RTU sends a random challenge to the MTU. Using the secret and the challenge, MTU calculates a hash digest and then sends this digest as the response to the challenge. The RTU checks the response against its own calculation of the expected hash digest. If the digests match, RTU acknowledges the authentication, otherwise the RTU terminates the connection. Whenever an authentication is required, RTU sends a new and different challenge and the rest of the above steps are repeated. In the augmented vulnerability tree, the challenge-response enhancement changes the base event of no-origin authentication by RTU to *false*.

4.3. Augmented vulnerability tree for security-enhanced system

The Authentication Octets and challenge–response algorithms were applied to the information-communication process of the test bed. As shown in the following subsections, the calculations of the updated threat-impact index and the CV index were performed by repeating the steps 2–5 for the enhanced system.

4.3.1. Steps 2 and 3 for the system with security enhancements

As shown in Fig. 2, the new augmented vulnerability, reconstructed with security enhancements, has nine base events marked *false*, which have been eliminated by the security enhancements. Since all the base events for the control-message-modification attack are eliminated, the threat-impact index for this attack is now changed from 29 to 0, since this attack in no longer possible. The tree analysis shows that the rest of the attacks could still be launched. The eliminated attack reduces the threat-impact index of the system by 29, giving the new threat-impact index for the enhanced system as 71.

4.3.2. Steps 4 and 5 for the system with security enhancements

The new CV index for the system can be calculated by removing the base events that were eliminated by the enhancements. Such events are marked *false* in the following list of base events:

- Nonce not include in the response: CV index of 1.
- Response sent in plaintext: total: CV index of 2
- *False*: Nonce not included in request.
- *False*: MTU sends request in plaintext.
- *False*: No-origin authentication by RTU.
- No-origin authentication by MTU: CV index of 14.5.
- Session open after unsuccessful tries: CV index of 13.5.
- *False*: No content authentication by RTU.
- No-access authentication by MTU: CV index of 7.
- No-access authentication by RTU: CV index of 5.
- No-content authentication by MTU: CV index of 1.

The completed augmented tree with the two security enhancements is shown in Fig. 2. The CV index can be calculated using the tree or directly using the above list. Adding all the CV indices for the base events, the top event in the augmented vulnerability tree diagram now has a CV index of 44, which represents the new CV index for the system with the security enhancements.

5. Conclusions and future research

This paper has presented a new risk assessment method, based on vulnerability trees, augmented with two new indices, for quantifying the risks resulting from cyber threats on a computer-based information system. These indices include the threat-impact index representing quantitative financial risk; and the CV index representing system vulnerability. We illustrated their use on a SCADA test bed without any security enhancement and a system with enhanced communication security. The proposed method can help information-system managers determine the current level of security and help them select security mechanisms (hardware or software).

Several directions are open for future research. In the augmented vulnerability tree diagram, a cut set (a combination of events resulting in the undesirable event) can be constructed that would lead to the risk of certain dollar amount losses. Alternately, a risk assessment curve in the form of Farmer's curve (Ayyub, 2003), showing a constant dollar-risk line, can be created using a chart plotting consequences in dollars against the expected number of occurrences of an undesirable event. Thirdly, in the damage analysis table, probabilities have not been assigned to damage represented by each column. In future work, probabilities can be added to each damage category, with the dollar amount multiplied by the assigned probability. All three of these extensions would help to further quantify the risk associated with information systems.

References

Aagedal, J., Braber, F., Dimitrakos, T., Gran, B. A., Raptis, D., & Stolen, K. (2002). Model-based risk assessment to improve enterprise security. In *Proceedings of the sixth international distributed object computing conference, September 17–20* (pp. 51–62). Switzerland: Lausanne.

Alberts, C., Dorofee, A., Stevens, J., & Woody, C. (2003). Introduction to the OCTAVE approach. Retrieved, from ⟨http://www.cert.org/octave/approach_intro.pdf⟩.

Ayyub, B. (2003). *Risk analysis in engineering and economics*. Boca Raton, FL: CRC Press.

Byres, E. & Lowe, J. (2004). The myths and facts behind cyber security risks for industrial control systems, In *VDE Congress*, Berlin, October 18–20, 213–218.

Campbell, K., Gordon, L. A., Loeb, M. P., & Zhou, L. (2003). The economic cost of publicly announced information security breaches: empirical evidence from the stock market. *Journal of Computer Security*, *11*(3), 431–448.

Campbell, P., & Stamp, J. (2004). A classification scheme for risk assessment methods. Sandia National Laboratory report SAND2004-4233, August.

Cavusoglu, H., Mishra, B., & Raghunathan, S. (2004a). A model for evaluating IT security investments. *Communications of the ACM*, *47*(7), 92–97.

Cavusoglu, H., Mishra, B., & Raghunathan, S. (2004b). The effect of internet security breach announcements on market value: Capital market reactions for breached firms and internet security developers. *International Journal of Electronic Commerce*, *9*(1), 69–104.

Cohen, F. (1999). Simulating cyber attacks, defenses, and consequences. *Computers and Security*, *18*(6), 479–518.

Department of Homeland Security. (2006). National Infrastructure Protection Plan: Base Plan, revised draft V2. National Infrastructure Institute.

⟨http://www.ni2ciel.org/NIPC/Revised-Draft-NIPP-v2.0.pdf⟩. Accessed on January 5, 2007.
Dhillon, G. (1997). *Managing information system security*. London, UK: Macmillan Press Ltd.
Ettredge, M., Richardson, V. J. (2002). Assessing the risk in e-commerce. In *Proceedings of the 35th Hawaii international conference on system sciences*, Big Island, HI, January 7–10, 194.
Farahmand, F., Navathe, S. B., Enslow, P. H., & Sharp, G. P. (2003). Managing vulnerabilities of information systems to security incidents. In *Proceedings of the fifth international conference on electronic commerce* (pp. 348–354). Pittsburgh: Pennsylvania. September 30–October 03
Farahmand, F., Navathe, S. B., Sharp, G. P., & Enslow, P. H. (2005). A management perspective on risk of security threats to information systems. *Information Technology and Management*, *6*(2–3), 203–225.
Gordon, L. A., & Loeb, M. P. (2002). The economics of information security investment. *ACM Transactions on Information and System Security*, *5*(4), 438–457.
Haimes, Y. Y., & Chittester, C. G. (2005). A roadmap for quantifying the efficacy of risk management of information security and interdependent SCADA systems. *Journal of Homeland Security and Emergency Management*, *2*(2), 1–21.
Haimes, Y. Y., Kaplan, S., & Lambert, J. H. (2002). Risk filtering, ranking, and management framework using hierarchical holographic modeling. *Risk Analysis*, *22*(2), 381–395.
Henley, E., & Kumamoto, H. (1996). *Probabilistic risk assessment* (2nd ed.). New York: IEEE Press.
Hovav, A., & D'Arcy, J. (2003). The impact of denial-of-service attack announcements on the market value of firms. *Risk Management and Insurance Review*, *6*(2), 97–121.
Hovav, A., & D'Arcy, J. (2004). The impact of virus attack announcements on the market value of firms. *Information Systems Security*, *13*(3), 32–40.
Iheagwara, C. (2003). More effective risk assessment. *Computer Security Journal*, *19*(2), 8–20.
Longstaff, T., Chittister, C., Pethia, R., & Haimes, Y. (2000). Are we forgetting the risk of information technology? *IEEE Computer*, *33*(12), 43–51.
Mell, P., Scarfone, K., & Romanosky, S. (2006). Common vulnerability scoring system. *IEEE Security and Privacy*, *4*(6), 85–89.
Miller, D., & Byres, E. (2005). Risk assessment: The first step. *InTech*, *52*(3), 68–69.
Nozick, L. K., Turnquist, M. A., Jones, D. A., Davis, J. R., & Lawton, C. R. (2005). Assessing the performance of interdependent infrastructures and optimizing investments. *International Journal of Critical Infrastructures*, *1*(2-3), 144–154.
Patel, S. (2006). Secure internet-based communication protocol for SCADA networks. Ph.D. dissertation, University of Louisville, Louisville, Kentucky.
Patel, S., Graham, J., Ralston, P., & Tantalean, R. (2005). Secure SCADA communications, monitoring, and control over the Internet. In *Proceedings of the 18th computer applications in industry and engineering* (pp. 169–174). Hawaii: Honolulu. November 9–11.
Patel, S., Tantalean, R., Ralston, P., & Graham, J. (2005). *Supervisory control and data acquisition remote terminal unit testbed. Intelligent Systems Research Laboratory technical report TR-ISRL-05-01, Department of Computer Engineering and Computer Science*. Louisville, Kentucky: University of Louisville.
Peltier, T. R. (2001). *Information security risk analysis*. Boca Raton, Florida: Auerbach Publications.
Pumfrey, D. J. (1999). The principled design of computer system safety analyses. D. Phil. thesis, University of York, UK.
Rakaczky, E. (2005). Building a security business case. Process control systems forum, October 25–27, Chicago, Illinois, ⟨www.pcsforum.org/events/2005/fall/pdf/Building%20a%20Security%20Business%20Case2a.pdf⟩. Accessed on January 5, 2007.
Ralston, P. A., Graham, J. H., & Patel, S. C. (2006). *Literature review of security and risk assessment of SCADA and DCS systems. Intelligent Systems Research Laboratory technical report TR-ISRL-06-01, Department of Computer Engineering and Computer Science*. Louisville, KY: University of Louisville.
Rinaldi, S. M. (2004). Modeling and simulating critical infrastructures and their interdependencies. In: *Proceedings of the 37th Hawaii international conference on system sciences*, January 05–08, Big Island, HI, pp. 1–8.
Schneier, B. (1999). Attack trees. *Dr. Dobb's Journal of Software Tools*, *24*(12), 21–29.
Stoneburner, G. (2006). Toward a unified security/safety model. *Computer*, *39*(8), 96–97.
Vidalis, S., & Jones, A. (2003). *Using vulnerability tress for decision making in threat assessment. School of Computing technical report CS-03-02*. Pontypridd, Wales, UK: University of Glamorgan.
Whitman, M., & Mattord, H. (2005). *Principles of information security* (2nd ed.). Boston: Course Technology.
Yasinsac, A., Childs, J. (2001). Analyzing internet security protocols. In *Proceedings of the sixth international conference on high assurance systems*, Boca Raton, FL, October 2001, pp. 149–159.

Sandip C. Patel is an assistant professor in the Department of Information Science and Systems at the Morgan State University. He received his Ph.D. in computer science from the University of Louisville and Master of Science in information and computer science from the Georgia Institute of Technology. He has over 10 years of industry experience. Dr. Patel has worked as a consultant with IBM and as a systems analyst and engineer at AT&T (formerly, NCR Corporation). He has also established and operated a computer consulting company. His research interests include risk assessment and the security enhancements to Supervisory Control and Data Acquisition (SCADA).

James H. Graham is the Henry Vogt Professor and Chair of Computer Science and Engineering at the University of Louisville. He has over 30 years of experience in computing and electrical engineering. Prof. Graham has served as a faculty member at the Rensselaer Polytechnic Institute and as a product engineer with General Motors Corporation. He received his M.S. and Ph.D. from the Purdue University in 1978 and 1980, respectively. He is a senior member of the Institute of Electrical and Electronics Engineers and a registered professional engineer. His research interests involve information security, algorithms for computational science, intelligent systems, distributed computing, computer simulation, and supply chain management applications.

Patricia A. S. Ralston is a professor of Chemical Engineering and Acting Chair of the Department of Engineering Fundamentals at the J. B. Speed School of Engineering at the University of Louisville. She received her Master of Engineering and Ph.D. degrees from the University of Louisville in 1980 and 1983, respectively. Her fields of expertise include process modeling, simulation, and process control. She has specific research interests in process monitoring, fault detection, and security of SCADA systems. Dr. Ralston is also a member of the Engineering Mathematics Core (EMCORE) Faculty and teaches mathematics courses for all engineering undergraduates.