Managing Complex IT Security Processes with Value Based Measures

Robert K. Abercrombie, *Sr. Member, IEEE,* Frederick T. Sheldon, *Sr. Member, IEEE* and Ali Mili

Abstract—Current trends indicate that IT security measures will need to greatly expand to counter the ever increasingly sophisticated, well-funded and/or economically motivated threat space. Traditional risk management approaches provide an effective method for guiding courses of action for assessment, and mitigation investments. However, such approaches no matter how popular demand very detailed knowledge about the IT security domain and the enterprise/cyber architectural context. Typically, the critical nature and/or high stakes require careful consideration and adaptation of a balanced approach that provides reliable and consistent methods for rating vulnerabilities. As reported in earlier works, the Cyberspace Security Econometrics System provides a comprehensive measure of reliability, security and safety of a system that accounts for the criticality of each requirement as a function of one or more stakeholders' interests in that requirement. This paper advocates a dependability measure that acknowledges the aggregate structure of complex system specifications, and accounts for variations by stakeholder, by specification components, and by verification and validation impact.

I. Introduction

THE lack of sound and practical security metrics is severely hampering progress in the development of secure systems. The large number of potential threats and corresponding vulnerabilities that lurk in an information system represent a significant risk to any enterprise due to the potential of being exploited. As such systems become even more complex and pervasive we must certainly find better ways to manage and account for these risks toward protecting our information and critical infrastructure assets. A well managed rigorous process will not only help us better understand and prioritize risk mitigation efforts, but should help to solidify vulnerability classification and rating efforts in a consistent structured and fine grain manner.

The Cyberspace Security Econometrics System (CSES) offers the following advantages over traditional measurement systems: (1) CSES reflects the variances that exist among different stakeholders of the same system. Different stakeholders will typically attach different stakes to the same requirement or service (e.g., a service may be provided by an information technology system or process control system, etc.). (2) For a given stakeholder, CSES reflects the variance that may exist among the stakes one attaches to meeting each requirement. The same stakeholder may attach different stakes to satisfying different requirements within the overall system specification. (3) For a given compound specification (e.g., combination(s) of commercial off the shelf software and/or hardware), CSES reflects the variance that may exist among the levels of verification and validation (i.e., certification) performed on components of the specification. The certification activity may produce higher levels of assurance across different components of the specification than others. (4) For a given information system, account for, in a consistent and structured way, the various dimensions of information assurance (i.e., linear type classification/rating) yielding a priority ordering of vulnerabilities based on impact severity in terms of cost. A general blueprint for this quantitative framework for information security risk management can be seen in Figure 1. The CSES process provides the hard science that falls inside this framework.

II. Motivation

Traditionally, the verification and validation (V&V) effort is charged uniformly on all stakeholders, which may/ may not include the cost of information assurance (IA) and/or certification and accreditation[2]. With the quantification

The submitted manuscript has been authored by a contractor of the U.S. Government under contract DE-AC05-00OR22725. Accordingly, the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes.

R. K. Abercrombie is with the Cyberspace Sciences and Information Intelligence Research Group within the Computational Sciences and Engineering Division at Oak Ridge National Laboratory, P. O. Box 2008, Oak Ridge, TN 37831-6418 USA (phone: 865-241-6537; fax: 865-576-5943; e-mail: abercrombier@ornl.gov).

F. T. Sheldon is with the Cyberspace Sciences and Information Intelligence Research Group within the Computational Sciences and Engineering Division at Oak Ridge National Laboratory, P. O. Box 2008, Oak Ridge, TN 37831-6418 USA Niles, Jr., was with Rice University, Houston, TX 77005 USA. He is now with the Department of Physics, Colorado State University, Fort Collins, CO 80523 USA (e-mail: sheldonft@ornl.gov).

A. Mili is with the College of Computing Sciences at the New Jersey Institute of Technology, Newark, NJ 07102-1982 USA (e-mail: mili@cis.njit.edu).

[2] Certification and Accreditation (C&A) pulls its authority from 2 major areas of guidance. The Federal Information Security Management Act (FISMA) of 2002, and the Office of Management and Budget (OMB) Circular A-130, Management of Federal Information Resources. FISMA is the law, and OMB Circular is the OMB policy required of all federal agencies. The DoD Information Assurance Certification and Accreditation Process (DIACAP) is the Department of Defense (DoD) process to ensure that risk management is applied on information systems.

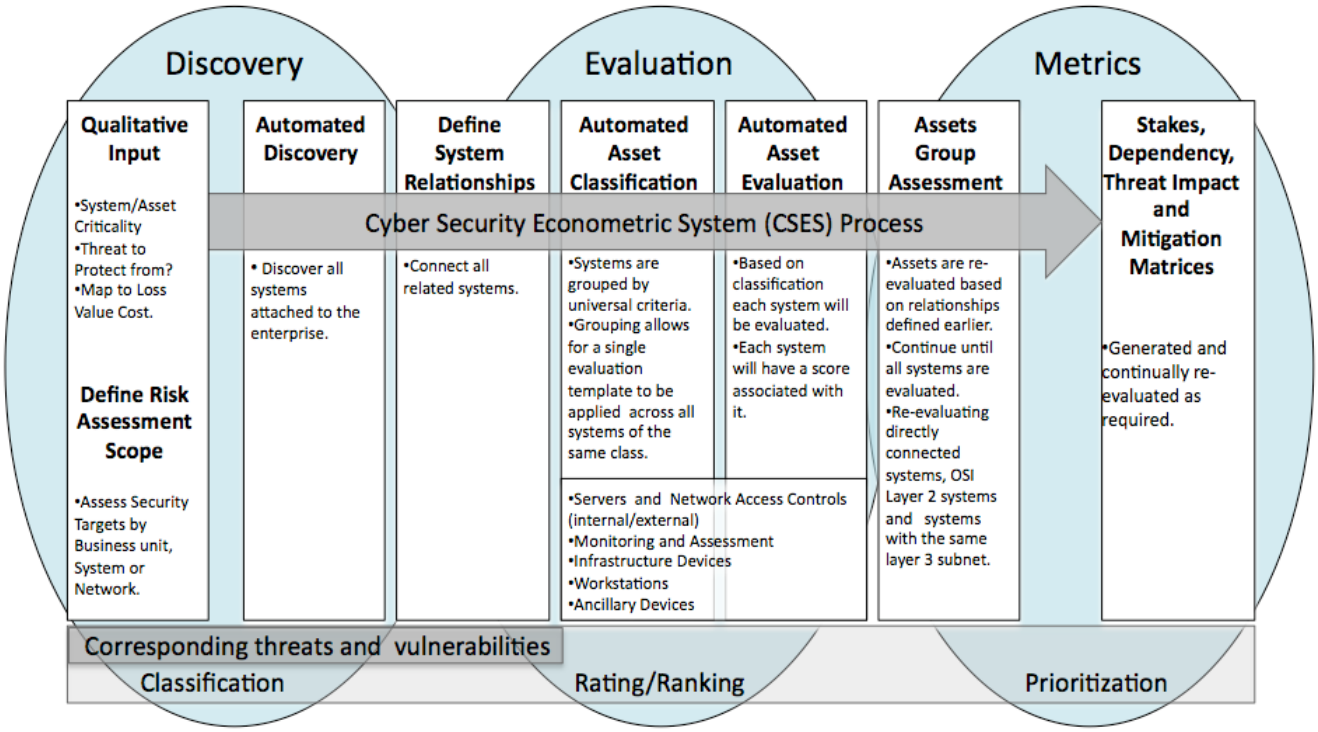

Fig. 1. A Quantitative Framework for Information Security Risk Management.

infrastructure that has been previously introduced to compute Mean Failure Cost (MFC), we can employ a scheme where the cost of any V&V effort is charged on the stakeholders according to what they stand to lose or gain [1]. Thus, if a particular V&V effort is aimed at improving the level of confidence that refines a component (i.e., that implements a service and/or satisfies a requirement), then stakeholders are charged according to the stake they have in satisfying said requirement. CSES also introduces and combines such measures as verification costs which consider the fact that certain requirements are easier to satisfy (and prove). Such costs depend on the system, the requirement and the selected verification method.

III. Related Works

The proposed measure is consistent with the spirit of Value Based Software Engineering [2-4]. While the Mean Time to Failure (MTTF) is an abstract quantity that reflects the failure rate of a system, the MFC quantifies the impact of failures by providing a failure cost per unit of time. This cost must be balanced against the benefit of operating the system for the same unit of time, to determine the desirability of operating the system at all.

A. Value Based Approaches

The history of Value Based Software Engineering, to date, is well documented [5]. Originally, software engineering only dealt with the technical challenges. This has changed over the years, especially in industry projects, when value was introduced to aid in the decision making process [6]. Historically, software engineering practice and research has been conducted in a value-neutral setting [3]. This approach led to an underestimation of the need to align the incentives of success-critical stakeholders [7]. Introduction of financially responsible approach to requirements prioritization enhanced value created potential [8]. Many evaluation approaches analyze costs, benefits and risks associated with Information Technology in general (e.g., cost-oriented approaches, multi-dimensional approaches, market-oriented approaches, strategy-oriented approaches, customer-oriented approaches and process-oriented approaches) [9]. Finally, controlling-oriented approaches unified the concepts of earned valued management and target costs. This was in turn influenced by Value Based Software Engineering [9]. Tracing value-base requirements and their impact has always been a challenge. A case study on value-based requirements tracing, that systematically supported project managers in tailoring requirements tracing precision and effort based on parameters stakeholder value, requirements risk/volatility, and tracking costs, illustrated this [10]. Other studies describe techniques required for distributed priority ranking of strategic requirements for information systems in economic organization [11]. Within the last few years, studies have made attempts to understand the stakeholder view of quality [12]. The mapping from SSE-CMM process areas to the patient-centered healthcare domain has the potential to establish a set of metrics to assess security risks for patient-centered healthcare systems [13]. Recently additional cybersecurity risk assessment and management endeavors include the Bayesian techniques [14] with Defense Graphs [15] and mitigation techniques [16]. This specific subject domain of cybersecurity has application for expansion to other areas of engineering, science, manufacturing business, management and public policy [17].

B. Classification Approaches

There are two popular classification schemes that are currently used to rate vulnerabilities. First, consider the X-force scheme [18] that uses vulnerability ratings of high, medium and low. Such course grain schemes have limitations when applied to different organizations that have different threat concerns. The second three-part scheme [19] combines measures of popularity, simplicity and impact into a numerical rating of 1 through 10 within each group. This rating scheme has limitations in the category selections as well as course differences between the categories that may lead to inconsistencies in determining the ratings.

Other classification schemes either address entire systems and broad levels of security [20] or develop general classifications for a given class of security systems, such as intrusion detection systems [21]. Other related work concentrates on how an organization should manage security vulnerabilities and the steps they should take to minimize risk [22]. Additional efforts focus on how to analyze vulnerabilities using a model checker to simulate attack scenarios [23].

C. Defining Vulnerability

Most vulnerability classification schemes assume: *A vulnerability is a weakness in a system that allows an attacker*[3] *to illegitimately gain information or access, gain increased privileges, deny the use of the system, impersonate the identity of some legitimate user, or help hide the detection of an attack.* Moreover, the following items are considered to be vulnerability points within a system:

- Allowing a machine to be port scanned through a firewall
- Default permissions on directories
- Default permissions on registry settings
- Improper log settings
- Password cracking
- Denial of service attacks

Most systems have some vulnerabilities of one form or another and most default installations of operating systems incorporate a large number of vulnerabilities. Consequently, it is critical to identify and characterize vulnerabilities as well as rate the severity and priority of the vulnerability using reliable, consistent metrics.

D. Current Vulnerability Rating Approaches

Most existing vulnerability rating schemes employ a linear rating scale. The most common ratings use three descriptors, high, medium and low [18]. These descriptors are defined as follows:

High – any vulnerability that provides an attacker with immediate access into a machine, gains super user access, or bypasses a firewall. *Example:* intruder executes commands on a mail server.

Medium – Any vulnerability that provides information that has a high potential of providing system access to an intruder. *Example:* intruder obtains a password file that could contain an account with a guessable password.

Low – Any vulnerability that provides information that potentially could lead to compromise. *Example:* intruder identifies who is online including potentially password crackable accounts.

These ratings, though easy to calculate, have several critical limitations as described in Section E.

Another rating scheme is the *Delphi methodology* that uses subject matter experts (SMEs) and works toward achieving consensus for the classification of vulnerabilities. Helmer and Dalkey developed the general methodology in the 1950s [24]. Delphi relies on providing an open-ended questionnaire to a carefully selected panel of experts on a particular topic. These experts remain anonymous to each other and feedback on the results of the questionnaires is provided by an independent reviewer/coordinator. This process usually involves three iterations. After the first round, questionnaire results are tabulated and returned to the panel without identifying the respondents. Subsequent rounds require the participants to rate the relative importance of individual items and to make changes to the phrasing or substance of the items. Typically, after three rounds, the process produces a consensus. Modifications are sometimes incorporated into the basic Delphi method, including beginning the process with a set of carefully researched and selected items to improve the initial round response quality.

E. Disadvantages

Simple, multi-level, course-grain rating schemes have several critical drawbacks:

- Subjective and inflexible
- Sensitivity to assessment errors
- Inconsistencies among analysts who do the rating
- Conflicts and overlap within rating groups
- One dimensional aspect
- Most vulnerabilities overlap rating categories

These types of approaches consist of a small number of groups with little room for error. Consider three rating levels: high, medium and low. A bad choice has an inherent error of 33%, which would likely cause serious consequences for any assessment. The Delphi consensus process (above) has inherent problems that can be pinpointed when assigning vulnerability levels and are summarized as follows:

- Forecasts are heavily dependent on the particular SMEs
- Questionnaires may inadvertently introduce bias toward a particular outcome
- Such processes are highly sensitive questionnaire ambiguities used for data collection in each round

[3] An attacker is any unauthorized user of the system or anyone that is using his or her access in a way that it was no intended to be used. The second part is important because some people would say that an authorized user of the system, who sniffs traffic to read passwords is not unauthorized, but they are still considered an attacker.

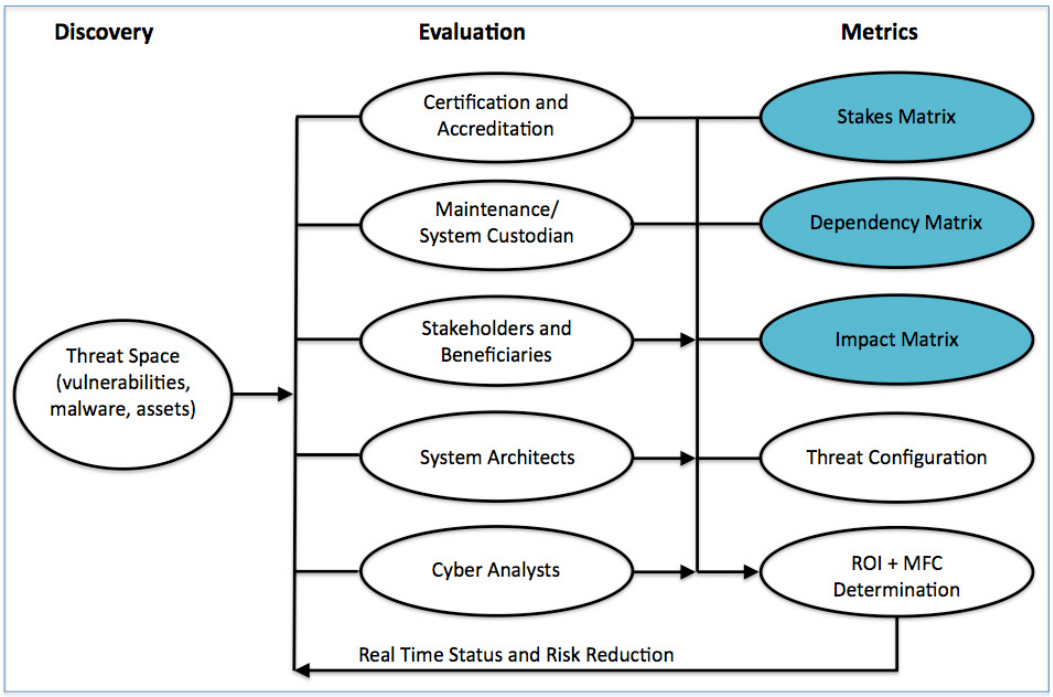

Fig. 2. Cyber Security Econometric System (CSES).

IV. CSES: Concepts and Assumptions

When assessing cybersecurity, we must consider many dimensions. Primarily, these dimensions are focused on either the left or right side of boom (e.g., a significant even such as a break-in). On the left side are preemptive/protective measures including all steps prior to the "system's" deployment as well as those that are designed to support measures on the right (post deployment). The right side includes damage assessment and recovery measures. This framework is designed to enable comprehensive exploration of the "likely" consequences of the various trade-offs on both sides. For example, we may (i) identify vulnerabilities and provide options to mitigate those at their earliest stages before they become more pernicious, (ii) codify the concomitant methodologies and processes that consider the full range of stakes (criticality/assets) and associated (operational) risks, and (iii) manage explicit investments (countermeasures, certification and accreditation (C&A) among the many feasible left side courses of action). Ultimately, as the system evolves the precision (and accuracy) of the assessments will help all aspects from C&A, intrusion avoidance to attribution including such measures as return on investment (ROI) and MFC.

A. Assumptions

In Figure 2, we see the essential input/output components and phases (i.e., discovery, evaluation and metrics) including data collection/analysis and consisting of the following entities [25, 26]:

- **System Stakeholders** are any person or organization that has a stake in the operation of the system (i.e., users, operators of the system, hosts of the systems, etc.).
- **Security Specification** used in the same way that correctness is a relative attribute (a system is correct with respect to its functional specification) and refers to a representation of the security attributes that a system must satisfy to be deemed secure.
- **Security Requirement** used in the same way that a complex functional specification is typically composed of simpler components (representing elementary functional properties), and is composed of simpler security requirements.
- **Mean Failure Cost (MFC)** used in the operational sense because the lack of security within the system may cause damage, in terms of lost productivity, lost business, lost data, resulting in security violations. We represent this loss by a random variable, and define MFC as the mean of this random variable [1]. As discussed further, this quantity is not intrinsic to the system, but varies by stakeholder [27].
- **For a given information system**, account for the various dimensions of information assurance (i.e., linear type classification/rating) yielding a priority ordering of vulnerabilities based on impact severity in terms of MFC.

B. Step-Wise Process of CSES

To estimate the MFC of a system for a set of stakeholders, we initially identify and then maintain (from the discovery phase) the following information: (1) the set of stakeholders of the system, and (2) the set of security specifications and thus security requirements that are to be satisfied by the system. (3) For each stakeholder and each security requirement, the stake that the selected stakeholder attaches to the selected service (or conversely, the cost that the stakeholder incurs if the service is disrupted). This information is provided by stakeholders. (4) For each component of a specific security requirement, the likelihood that the system provides that service as specified. This information is computed in light of the V&V measures (inspection, verification, testing, security measures, firewalls, vulnerability removal, threat mitigation, etc) that the system has been subjected to. In particular, estimating the likelihood of delivering a service requires that we determine to what degree the components involved in delivering a service have been validated. Thus, following the CSES vertical process of the Metrics Engine proceeds in three steps (applying Stake Estimation to generate the Stakes

Matrix, Bayesian Analysis to generate the Dependency Matrix, and Threat Analysis to generate the Impact Matrix) by the subject matter experts as described in the vertical Evaluation Engine components [25, 26]. CSES encompasses not only failure costs but also mitigation costs, specifically verification costs. Once the basic matrices are populated, a baseline for the particular instantiation of the CSES is established and all changes to the baseline are maintained in a way that track the enterprise's evolution to provide near real-time assessments.

C. *Definitive Instantiation*

As shown in Figure 2, the system follows a defined process. The initial inputs (1) organization mission (and components thereof), (2) value of its objectives and assets if uninterrupted, and (3) the components of the enterprise system that support each mission component, are determined by stakeholders.

The stakeholder/customer, with assistance from Subject Matter Experts (SMEs), defines their criteria for evaluating their assets. For example, the criteria may include:

• Financial basis (e.g., operational cost of downtime per unit of time defined with hardware/software costs, HVAC, staffing, etc., versus profit); which is the quantitative measurement to be used within the CSES.

• Federal Information Security Management Act (FISMA) of 2002, customer derived value of assets per NIST 800-60, and/or FIPS 199/200 (February 2004, Standards for Security Categorization of Federal Information and Information Systems) dictated requirements.

• Customer defined requirements; acceptable and unacceptable impact levels against cost value related to Information Assurance tenets of confidentiality, availability and integrity may also be examined.

Variances exist among different stakeholders of the same system. Different stakeholders will attach different stakes to the same requirement or service (e.g., a service may be provided by an information technology system or process control system, etc.). For a given stakeholder, CSES will reflect the variance that may exist among the stakes the stakeholder attaches to meeting each requirement. The same stakeholder may attach different stakes to satisfying different requirements within the overall system specification. Once CSES is base lined and it evolves, compound specification (e.g., combination(s) of commercial off the shelf software and/or hardware) will become apparent. For a given compound specification, CSES will reflect the variance that may exist among the levels of V&V (i.e., certification) performed on components of the specification. The certification activity may produce higher levels of assurance across different components of the specification than others.

For each component of a specific security requirement, the likelihood that the system provides that service as specified. This information is computed in light of the V&V measures (inspection, verification, testing, security measures, firewalls, vulnerability removal, threat mitigation, etc) that the system has undergone. In particular, estimating the likelihood of delivering a service requires that we analyze to what degree the components that are involved in delivering this service have been validated.

V. SUMMARY

This paper advocates a dependability measure that acknowledges the aggregate structure of complex system specifications, and accounts for variations by stakeholder, by specification components, and by verification and validation impact. Furthermore, we have outlined an information security risk assessment process leading towards quantitative information systems risk management. First we must quantify risk to an organizations mission based on the systems and stakeholders supporting that mission: 1) define the mission(s), 2) identify all systems that directly support the mission(s) rating each system according to its criticality to the mission (accounting for both direct and indirect applicability), 3) classify assets based on type (server, workstation, router, firewall etc.), 4) apply V&V criteria (using both manual and automated testing)[4], and 5) analyze all the system components/groups to generate an aggregate score that can be trended over time[5].

CSES, as it relates to metrics/benchmarks for information assurance risk assessments, supports the following vision of operational capabilities to identify, collect, and analyze IA benchmarks and metrics to:

- quantify key indicators for improvements in IA processes,
- provide meaningful input to decision-making processes for IA risk management,
- reduce risk by providing a measure that demonstrates evidence of risk mitigations,
- offer an understanding of the costs and benefits of IA across the enterprise.

Current security metrics include:

- numbers of violations recorded in security audit logs of individual systems
- number of systems within an organization that were vulnerability tested over the course of a year.
- number of vulnerabilities identified per code segments/ shared libraries
- time to patch systems per level of criticality to the mission (organization or stakeholder) and level of severity of threat.

Effective security metrics will be used to identify security weaknesses, determine trends to better utilize security resources, and measure the success or failure of implemented security solutions. Ultimately, the metrics should help characterize an organization's overall security posture from risk/threat/vulnerability, budgetary, and regulatory standpoints.

---

[4] For example, to evaluate the effectiveness of a firewall. The manual analysis would include its placement while the automated portion can review its rule-set and generate a score based on how tightly the access control lists are written (e.g., high risk ports are given a higher weight, etc.)

[5] Automated portions of the testing can be run frequently to track changes over time.

VI. Conclusions

The CSES process proceeds in three steps (Generation of Stakes Matrix, Dependency Matrix and Threat Matrix). CSES encompasses not only failure costs but also mitigation costs, specifically verification costs. CSES provides:

- A framework for measuring the appropriate attributes that support the decisions necessary to (1) design security countermeasures, to choose between alternative security architectures, (2) respond to events such as intrusions or attacks and, (3) improves security (including reliability and safety) during both design and operational phases.
- A comprehensive basis for choosing courses of action that have the highest risk reduction return on investment (i.e., reduce the most risks for the lowest cost).

CSES and its underpinning rationale are (1) consistent with the spirit of Value Based Software Engineering and (2) comprehend the different organizational mission needs for all stakeholders. For example, CSES identifies information assurance controls and mitigation costs as an investment toward assuring mission success.

A. Future Work

On the practical side, we need to find sample applications where deployment of the CSES with its associated MFC metric show usefulness and superiority, by providing a sound basis for analysis and decision-making. On the theoretical side, we need to develop the mathematical infrastructure that allows us to estimate or to approximate the MFC using failure costs and failure probabilities given (respectively) by stakeholders and engineers (V&V teams).

Acknowledgment

The authors would like to thank Beth Bidwell and Eric Cole at Lockheed Martin for the invaluable help in defining and developing this work.

References

[1] A. Mili and Sheldon, FT, "Measuring Reliability as a Mean Failure Cost," in Proceedings of the 10th IEEE High Assurance Systems Engineering Symposium. Dallas, TX: IEEE, 2007, pp. 403-404.

[2] Biffl, S, Aurum, A, Boehm, BW, Erdogmus, H and Gruenbacher, P, "Value Based Software Engineering," Springer Verlag, 2006.

[3] Boehm, BW and Huang, L. "Value Based Software Engineering: A Case Study," *IEEE Computer*, March 2003.; 36(3).

[4] Boehm, BW and Huang, L. "Value Based Software Engineering: Reinventing Earned Value Monitoring and Control," *ACM Software Engineering Notes*, March 2003.; 28(2).

[5] Boehm, B, "A View of 20th and 21st Century Software Engineering," in Proceedings of the 28th International Conference on Software Engineering. Shanghai, China: ACM, 2006.

[6] Omasreiter, H, "Balanced Decision Making in Software Engineering--General Thoughts and a Concrete Example from Industry," in Proceedings of the First International Workshop on The Economics of Software and Computation: IEEE Computer Society, 2007.

[7] Egyed, A, Biffl, S, Heindl, M and Paul, G, "A Value-Based Approach for Understanding Cost-benefit Trade-offs during Automated Software Traceability," in Proceedings of the 3rd International Workshop on Traceability in Emerging Forms of Software Engineering. Long Beach, California: ACM, 2005.

[8] Cleland-Huang, J and Denne, M, "Financially Informed Requirements Prioritization," in Proceedings of the 27th International Conference on Software Engineering. St. Louis, MO, USA: ACM, 2005.

[9] Mutschler, B, Bumiller, J and Reichert, M, "Designing an Economic-Driven Evaluation Framework for Process-Oriented Software Technologies," in Proceedings of the 28th International Conference on Software engineering. Shanghai, China: ACM, 2006.

[10] Heindl, M and Biffl, S, "A Case Study on Value-Based Requirements Tracing," in Proceedings of the 10th European Software Engineering Conference held jointly with 13th ACM SIGSOFT International Symposium on Foundations of Software Engineering. Lisbon, Portugal: ACM, 2005.

[11] Sobczak, A and Berry, DM. "Distributed Priority Ranking of Strategic Preliminary Requirements for Management Information Systems in Economic Organizations," *Information and Software Technology*, 2007; 49(9-10): 960-984.

[12] Boehm, B, Chulani, S, Verner, J and Wong, B, "Sixth Workshop on Software Quality," in Companion of the 30th International Conference on Software Engineering. Leipzig, Germany: ACM, 2008.

[13] Huang, L, Bai, X and Nair, S, "Developing a SSE-CMM-based Security Risk Assessment Process for Patient-Centered Healthcare Systems," in Proceedings of the 6th International Workshop on Software Quality. Leipzig, Germany: ACM, 2008.

[14] Dantu, R and Kolan, P. "Risk Management Using Behavior Based Bayesian Networks," *Proceedings of IEEE International Conference on Intelligence and Security Informatics*, 2005; LNCS 3495115-126.

[15] Sommestad, T, Ekstedt, M and Johnson, P, "Cyber Security Risks Assessment with Bayesian Defense Graphs and Architectural Models," in Proceedings of the 42nd Hawaii International Conference on System Sciences, vol. 42. Waikoloa, HI: IEEE Computer Society, 2009, pp. 10.

[16] Ekelhart, A, Fenz, S and Neubauer, T, "AURUM: A Framework for Information Security Risk Management," in Proceedings of the 42nd Hawaii International Conference on System Sciences, vol. 42. Waikoloa, HI: IEEE Computer Society, 2009, pp. 10.

[17] Haimes, YY. *Risk Modeling, Assessment, and Management*, 2nd ed. Wiley, 2004.

[18] Internet System Security, Xforce database. http://xforce.iss.net/

[19] McClure, S, Scambray, J and Kurtz, G. *Hacking Exposed, Network Security Secrets and Solutions*, 6th ed. McGraw-Hill Osborne Media: Columbus, OH, 2009; pp. 720.

[20] Department Of Defense Trusted Computer System Evaluation Criteria, December 1985, DOD 5200.28-Std.; http://www.radium.ncsc.mil/tpep/library/rainbow/5200.28-STD.html

[21] Debar, H, Dacier, M and Wespi, A, "Towards a taxonomy of intrusion-detection systems," in Computer Networks: The International Journal of Computer and Telecommunications Networking, vol. 31, H. Debar, M. Dacier, and A. Wespi,

Eds.: Elsevier North-Holland, Inc., New York, NY, USA, 1999, pp. 805-822.

[22] Rudd, A, McFarland, J and Olsen, S. "Managing Security Vulnerabilities in a Networked World," *J Digit Imaging*, 1998; 11(3 Suppl 1): pp. 216-8.

[23] Ritchey, RW and Ammann, P. "Using Model Checking to Analyze Network Vulnerabilities," Proceedings. 2000 IEEE Symposium on Security and Privacy, 2002-08-06, 2000, pp. 156-165.

[24] Dalkey, NC. *The Delphi Method, I : An Experimental Study of Group Opinion*. Rand Corp. ; Distributed by Clearinghouse for Federal Scientific and Technical Information, U.S. Dept. of Commerce, National Bureau of Standards, Institute for Applied Technology: Santa Monica, Calif., 1969.

[25] F. T. Sheldon, R. K. Abercrombie and Mili, A, "Evaluating Security Controls Based on Key Performance Indicators and Stakeholder Mission," in Proceedings of the 4th Annual Cyber Security and Information Intelligence Research Workshop. Oak Ridge, TN: ACM, 2008.

[26] F. T. Sheldon, R. K. Abercrombie and Mili, A, "Methodology for Evaluating Security Controls Based on Key Performance Indicators and Stakeholder Mission," in Proceedings of 42nd Annual Hawaii International Conference on System Sciences, vol. 42. Waikoloa, HI: IEEE, 2009, pp. 10.

[27] A. Mili and Sheldon, FT, "Challenging the Mean Time to Failure: Measuring Dependability as a Mean Failure Cost," in Proceedings of 42nd Hawaii International Conference on System Sciences, vol. 42. Waikoloa, HI: IEEE, 2009, pp. 10.