



Harmonizing and Uniting the Key Technical Disciplines for Risk Management of Cyber Security

Y. Y. Haimes
B. M. Horowitz
J. H. Lambert
J. R. Santos
K. G. Crowther

April, 2008

Abstract

This paper addresses the need to bridge the cultural, educational, and technical divides that are impeding professionals and organizations engaged in system and software development and associated security problems. In particular, harmonizing and uniting several key technical disciplines (software engineering, computer science, systems engineering) are critical for a sustainable risk management process incorporating the best practices of cyber security and information assurance. We identify the foundations for raising the level of shared culture and technical knowledge for system and software development, and suggest steps toward closing the cultural and technical gaps that divide the disciplines, including the development of joint curricula and other educational initiatives. A fundamental issue is how to link “mission assurance” with “information assurance,” recognizing that more and more missions are being provided with automation support and therefore are more and more information dependent. The paper is in six parts: Introduction, Six Principal Deficiencies of Cyber Security and Information Assurance, The Steps of a Systemic Risk-based Methodological Approach, The Roles of Universities and Other Institutions of Higher Education and Research, The Roles of the Institute for Information Infrastructure Protection, and Epilogue.

Preface and Definitions

Three commonly used terms are defined here in the context of this paper. *Information assurance* (IA) is characterized through five attributes: accessibility/availability, authenticity, integrity, confidentiality, and non-repudiation [George 2008]. *Cyber security*, which is a normative, subjective term, is not dissimilar to the term *safety*. Lowrance [1976] defines risk as a quantitative measure of the probability and severity of adverse effects, and safety as the level of risk that is acceptable. Thus, the term risk, according to Lowrance, is imbedded in the term safety. Similarly, the same connotation seems to apply to cyber security; namely, the lack of IA (the level of information that is deemed at risk) is imbedded in the term cyber security. Finally, *information technology* (IT) is a generic term that connotes all interrelated and interconnected elements that enable cyber communications, including hardware (computing electronic devices, sensors, and networks), software, modeling, analysis, and human interaction. In this paper, following the same logic, we will consider IA as a subset and a by-product of IT. Finally, the flattened world caused by the interdependencies through IT has added an element of fuzziness to the boundaries that characterize and define a system, primarily because the physical dimension of a classical system has become more illusive. Indeed, one cannot separate the would-be intruders around the world from the IT system itself, its owners, users, and operators. Therefore, the term *IT system* (used subsequently in this paper) ought to be understood in this broader, almost unbounded cyber domain.

A. Introduction

The adage “if it isn’t broken don’t fix it” does not apply to information assurance in general or to cyber security in particular. Our IT systems are fundamentally broken with dire ramifications that span financial, sociopolitical, national and international security, and even world stability, particularly given the threats of emerging state-sponsored network intrusion, spying, disruption, and denial of service. Myriad incidents of network intrusions have become increasingly prevalent, including attacks at the Pentagon and other U.S. defense offices, commercial enterprises, a variety of banking institutions, and corporate intellectual properties. For example, Traynor [2007] reports in the *Guardian* that Russia is accused of unleashing cyber war to disable Estonia. McDougall [2008] reports in *InformationWeek* that the hacker-trader in Societe Generale who is accused of operating a multibillion-dollar fraudulent trading scheme had only basic computing and programming skills. The estimated \$60-80 billion of the yearly corporate financial losses in the United States due to intrusions demonstrate a pattern that is grounded on the disconnectedness that has been the hallmark of the architectural design, development, deployment, and maintenance of hardware and software engineering and systems integration. Goodman and Lin [2007], in a publication of the National Academies, *Toward a Safer and More Secure Cyberspace*, suggest one reason an in-depth defense secured by appropriate design of software architectures has not been more widely implemented: the lack of attention and compliance by software engineers to the functions and processes generated by the organization. In reality, large information systems emerge from incremental additions to foundational software systems in ways that are generally unanticipated by the designers of the original systems, rather than from organizational and cultural forces [Goodman and Lin 2007]. The result is a need for software adaptations that constantly respond to evolving and emergent organizational and

institutional needs and changes. Moreover, it is the organizational functions that benefit from information system productivity that generate loss when the information system functionality is absent. Such organizational consequences and vulnerabilities are typically not the domain of the software engineers and computer scientists. It is perhaps for these and related reasons that Goodman and Lin [2007] identify only one of six recommended research focus categories that is purely in the domain of traditional computer science and software engineering. The solution must, therefore, rely not only on broadening the disciplines involved in the cyber security problem, but also—and most importantly—on integrating their works and contributions.

The immediate past president of the National Academy of Engineering, observed:

Although the nation is at great risk from cyberterrorism, we have virtually no research base on which to build secure systems . . . The truth is that we don't know how to build secure information systems. The only model widely used for cybersecurity is the *perimeter defense* model, which is demonstrably fragile. [Wulf and Jones 2004]

Richard Kendall, a visiting scientist with the Software Engineering Institute and former chief information officer of Los Alamos National Laboratory, said:

A long-standing cyber problem that will be exasperated over time is the security unworthiness of software engineering. . . . The complexity of the software environment has created something akin to emergent effects of which the developers are unaware, but hackers can uncover and exploit. Another aspect of this problem is that operating system developers do not enforce their own conventions. Furthermore, to counter this reality with encryption, which has become more pervasive, contributes both strength and weaknesses to our cyber security. . . . we may know that something is being infiltrated, but we don't know what it is or how the exploit works. [Chittister and Haines 2008]

Boehm [2006] provides details concerning ten trends in the development and evolution of our information infrastructure that are expected to continue in the next decade. Some of the more obvious trends include the increases in: computation power; availability of commercial off-the-shelf (COTS) solutions; the need for complex system integration of COTS, legacy, and customized systems; the size and complexity of systems; global and international connectivity; the need for software dependability and criticality; the focus on end-user needs; and the need to integrate systems and software engineering functions. Indeed, these trends apply both to end-users who seek to benefit from the productivity gains that our information infrastructure can provide and to those who seek to exploit vulnerabilities of systems (i.e., hackers). As such, hackers will have at their disposal increases in computation power, standardization of vulnerabilities through reliance on COTS, connectivity across the globe, opportunities to disrupt and exploit critical and dependable systems, and so forth. In a final project report by Boehm and Lane [2007], they evaluate relevant Department of defense guidance, education, and training and provide recommendations, including: (i) Further experience on synchronization of hardware and software increments; (ii) Further experience on synchronization with externally-evolving system components; (iii) More usage experience in application to systems of systems; (iv) Shortfalls in available educational materials and courses.

Longstaff et al. [2002] describe how the increased risk in COTS wireless communication technologies can have a critical impact on information assurance and trustworthiness of critical infrastructures. However, although the technological solutions are available, they are frequently not implemented due to organizational or institutional misalignment with technological needs. Chittister and Haines [2008] describe how conceptual mistakes made during system development, integration, and engineering result from an educational and technical disconnect among the several parties that contribute to the planning and development process. Rosenbloom [2004] aptly captures the need to rethink the traditional and current academic orientation that partitions computer science and engineering disciplines. One approach that would optimally cater to the growing call for harmonization and unification of computer science with engineering is the creation of academic programs that are explicitly interdisciplinary and geared toward holistic understanding of large scale and complex systems.

B. Six Principal Deficiencies of Cyber Security and Information Assurance

The challenges associated with cyber security and cyber intrusion cannot be isolated from the deficiencies in knowledge management and systems integration, including the high organizational walls that separate the disciplines (see, e.g., [Ashkenas et al. 1998]). This section identifies the six principal deficiencies of cyber *insecurity* that must be overcome in order to harmonize and bridge the divided several technical disciplines engaged in system and software development, toward the ultimate realization of a sustainable cyber security. Recognizing the six deficiencies should shape the priorities of the entire multidisciplinary development team

1. *compartmentalized development* in software engineering, which compromises cyber security throughout the system's life cycle;
2. *educational, training, and experience gaps* among software engineers, systems engineers, and hardware engineers and scientists;
3. *insufficient organizational interfaces and shared cyber security*, which are motivating the need to pursue research on harmonizing and uniting complementary disciplines involved in cyber security research;
- (iv) *insufficient funds allocated for cyber security*, which constitutes a gross oversight by the leadership in the private and public sectors;
- (v) *inattention to extreme events*, which results in the consequent misallocation of resources in addressing sources of risk to software and system development; and
- (vi) *lack of what Collins[2001] calls "the Hedgehog Concept"*; namely, focus on cyber security during software engineering development and the overall system integration.

1. Compartmentalized software engineering development

The various phases of the software development process define the *software lifecycle*. Those phases include function/domain analysis, scope determination, requirements specification, architecture specification, coding, testing, implementation, integration, documentation, training, maintenance, patching, and decommissioning. *Compartmentalization* of the software development process refers to the fact that some phases of the process are frequently performed by groups of experts and disciplines with a large degree of independence from those who manage other phases of the lifecycle.

Jackson [2006] writes: “Almost all grave software problems can be traced to conceptual mistakes made before programming started” (p. 72). Preventive risk management of software engineering development is essential to ensure reliability and cyber security of software systems, and a holistic lifecycle analysis is imperative to ensure that comprehensive planning is made prior to development and acquisition of new systems [Longstaff and Haines 2002]. Effective management of technological systems ought to incorporate considerations that include the hierarchical nature of such systems, coordination of various echelons of the organizational and decisionmaking structure, involvement of multiple players and stakeholders, and planning for various temporal and socioeconomic conditions.

Methods for a formal specification and verification process for early elimination of errors have been under development since at least 1975 [Boehm and Hoare 1975]. More recently, people- and value-oriented processes have emerged, such as Team Software Process [Humphrey 2000], risk-based processes for achieving dependability objectives [Gerrard and Thompson 2002; Huang 2005], systems engineering processes such as Lean Development [Womack and Jones 1996], and software abstraction methodologies such as alloy: A lightweight objective model notation [Jackson 2002].

A growing focus on software engineering indicates that, indeed, software systems are at the heart of integrated technological systems. The development of software-targeted lifecycle analysis models is an evident acknowledgment of the criticality of software systems, and, at the same time, an important recognition of the need for seamless integration of all elements of a large-scale and complex system (LSC). Several literature and historical incident accounts attribute system failures to disconnect among multiple players (e.g., different contractors for various system components), stakeholders, and managers. Chittister and Haines [2008] argue that “one of the challenges facing the professional software and systems engineering communities is how to develop software using disciplined engineering methods and processes to significantly improve the performance, security, and efficiency of high performance computing (HPC) technology in the presence of LSC.”

Risk of cyber intrusion is introduced throughout the developmental lifecycle of systems and software. Thus, effective software management framework necessitates analysis of at least four perspectives [Haines et al. 1997]: technical performance—to ensure reliability, schedule—to prevent delays, cost—to operate within financial constraints, and process—to maintain a seamless coordination and decision flow among various developers, contractors, managers, and other players in all lifecycle stages. Integration of technological systems necessitates a lifecycle framework wherein risks are identified prior to the development stages and adaptively managed at all stages of system operation, maintenance, upgrading, and retirement. Two models of systems engineering lifecycle analysis are the waterfall model [Boehm 1981] and the spiral model [Boehm 1988]. However, Boehm [2006] suggests that the waterfall model is largely inadequate with today’s software development needs, and suggests that adaptations of the spiral model will ultimately help fulfill the information system development needs of emerging and future systems. Software engineering is increasingly performed in an iterative/adaptive manner consistent with the spiral model, which has provided improvements to the software development process by bridging some of the compartmentalization gaps. Large compartmentalization gaps still remain, however, due to educational, training, and experiential gaps.

2. Educational, training, and experience gaps

Compartmentalization gaps that result in a disconnected software development process are compounded by the educational, training, and experiential gaps among software engineers, systems engineers, hardware engineers, scientists, and others who contribute throughout the IT system developmental processes. Current efforts to bridge these gaps typically focus on low-cost training for professionals in the field. Research is needed to understand the degree to which training by itself can bridge these gaps. Well-educated and -trained developers and maintainers of software-hardware engineering systems are the sine qua non for cyber security and information assurance. Indeed, to achieve a secure cyberspace will require a universal commitment (from all parties involved in the development of hardware-software systems) to technical knowledge; competence; dedication to quality and reliability; and appreciation of the role of risk-based decisionmaking in cyber security.

The advent of information technology paved the way for an evolutionary focus on process control from hardware to software engineering [Chittister and Haines 1996]. The integration of critical systems increasingly relies on software engineering, more than ever before, yet no major changes have been made in the curriculum of systems or software engineers. Effective software risk management is founded on comprehensive identification of sources of risk. Major categories of risk can be structured into hierarchical holographic models [Haines 1981, 2004] that comprise categories of perspectives such as software development, temporal perspective, leadership, the environment, the acquisition process, quality, and technology, thus providing a hierarchy of educational, training and experience demands that span the software lifecycle [Longstaff and Haines 2002].

The National Research Council [2002] initiated efforts to comprehensively identify and assess consequences of various threat scenarios to the United States. The Council argues that collaborative efforts by domain experts provide a highly leveraged opportunity to apply science and technology for formulating and deploying policies to maintain homeland security at all times. Goodman and Lin [2006] report requirements, principles, and categories that define an effective multidisciplinary research agenda to improve the security of our cyberspace. These clearly indicate the cross-domain nature of expertise necessary to accomplish cyber security research in the coming years. The existing pervasive divides among the domain experts in education, training, skills, orientation, and culture of systems engineers, software engineers, computer scientists, and others constitute significant sources of IT and cyber security risks. Boehm [2006] describes cross-cultural bridging required to unlock collaborative potential that could result from a global community to include a need for: cultural understanding (e.g., anthropologists), establishment of common shared visions (e.g., business and research leaders), and contracting mechanisms for incentives and trust (e.g., law, international law). Bridging the divides among the domain experts in education, training, skills, orientation, and culture of systems engineers, software engineers, computer scientists, and others is the sine qua non of effective systems integration, and hence cybersecurity [Chittister and Haines 2008]. Moreover, academic curricula must address the cultural divides among systems engineers, software engineers, and computer scientists, among others, through long-term and sustainable multidisciplinary research agendas pertinent to security of complex critical infrastructure

systems. A reasonable balance must be sought between specialization and cross-disciplinary training.

A clear advantage exists in creating a separate software engineering discipline that recognizes and understands the criticality of software systems and the vast array of associated industrial and organizational functions that should influence this process. Nevertheless, careful attention is required to ensure that development, acquisition, and operation of software systems follow a holistic life cycle-based management approach that addresses essential factors such as cost, technical performance, schedule, and decision flow process. Software systems exist in amorphous forms and are typically associated with hardware systems that are managed and controlled by operators and managers at different levels of the organizational hierarchy. Furthermore, elements of any technological system (software, hardware, human, organizational) will inevitably be integrated; therefore, academic and practical training of current and future systems engineers must not only focus on “independent” specialization, but also recognize, acknowledge, and respond to the natural connections of specialized systems.

3. Insufficient organizational interfaces and shared cyber security

Recognizing the proliferation of networked information systems and increasing trend in interconnectedness of critical infrastructure systems requires concerted and organized collaboration among software engineers, computer scientists, systems engineers, physicists, and other business and policy leaders. Wulf and Jones [2004] assert that most cyber security research efforts are based on “perimeter defense” models whose object is to protect what is “inside” the system or organization. Major flaws to such cybersecurity models include failure to protect the system from internal attacks, as well as attacks to external service providers that the system depends on highly for its complete operation. This assertion further implies that current research attention on developing models capable of describing interdependencies—particularly the connectivity of critical infrastructures to networked information systems—is inadequate. Goodman and Lin [2007] propose that one area of cyber security research should concentrate on mitigating the effects of cyber intrusion, not just the intrusion itself—implying that some degree of organization independence from information systems can still support the level of productivity demanded, but understanding this requires organizational interfaces between those planning the organizational structures and those planning cybersecurity. The NRC [2002] underscores the multiplicity of owners, operators, decisionmakers, and stakeholders in large-scale and complex systems (LSC). As such, cross-domain information sharing is imperative and should start in academic experience of future engineers through multidisciplinary studies and research projects focused on cyber security. Longstaff and Haimes [2002] describe how holistic development of survivable cyber-based infrastructure systems demands attention to organizational structure and its components including trust, knowledge management, organizational behavior, and other non-technology considerations.

The core values required by the organizations that develop software engineering and perform systems integration constitute important foundations for an ultimate sustainable cyber security. Consider the following statements by Collins and Porras [1994] in *Built to Last*:

Core ideology provides the bonding glue that holds an organization together as it grows, decentralizes, diversifies, expands globally, and attains diversity within . . . Core

values are the organization's essential and enduring tenets—a small set of timeless guiding principles that require no external justification; they have intrinsic value and importance to those inside the organization.

Collins [2001], in his book *Good to Great*, addresses the importance of the culture of discipline, transcending disciplined people, disciplined thought, and disciplined actions. Furthermore, there is no substitute for well-educated, well-trained engineers and scientists in charge of cyber security. Indeed, academic and practical training of software engineers should include exposure to system hacking and propagation dynamics of malwares. Risk assessment (particularly the process of identifying risk scenarios, their likelihood, and their consequences) ought to constitute a concrete foundation for developing strategies to counter the adverse outcomes that can arise from security breaches and their ripple effects. Removing the invisible high walls that separate the multiple disciplines involved in information technology is essential. Reevaluating and restructuring the curricula for students in computer science, systems engineering, electrical engineering, and other involved disciplines in the development of IT-based systems is another dimension of the core values that call for a fundamental organizational change in institutions of higher education. Longstaff et al. [2000] see the mission of government and private-sector organizations to:

- Apply their educational, technological, management, and policy expertise to the vital task of ensuring the security and survivability of their information infrastructure systems, both present and future.
- Ensure resilient and robust information infrastructures that continuously provide reliable, high-integrity services while protecting the privacy of everyone who uses them.
- Incorporate trust, science, technology, technology-transfer and education, and organizational behavior, not as separate entities, but as one indivisible Gestalt for information assurance and survivable dependable systems.

4. Insufficient funding for cyber security

The grossly insufficient funds allocated for cyber security constitutes a gross oversight by the leadership in the private and public sectors. Currently, there are no significant market-based drivers for improving cyber security, often leaving security executives to fight internal battles to make a business case for security investments. We must explore the potential for creating a common reference to characterize cyber security risk that could lead to investment drivers in the marketplace for corporations that can assure a higher level of security. There should be recognition of two important factors that determine cyber risk: the security technology that reduces attack likelihoods and the security business processes that contain the consequences when an attack is successful (e.g., limiting the technology use or enforcing restoration processes if the consequences are too extreme). The quality of a cyber defense system is the integrated result of these two factors, and any cyber security assessment method to be used for market purposes must account for both of them.

Funding for cyber security is not yet based on sound priorities. Consider the common vulnerability scoring system (CVSS), which is a multifactor scoring system that integrates vendor-measured and user-measured factors for vulnerability into a single combined CVSS score. The CVSS score, which provides a transparent measure of specific software

vulnerabilities, is used in company purchase decisions; however, it is not designed for rating the security of an entire IT system of a company. A financial scoring system, such as bond ratings and credit scoring, is another important example. Bond ratings (e.g., Dun & Bradstreet), where the capacity of a company to meet debt offerings is judged, are based on sound analytical methods. In cyber security, the trend is to hypothesize that the capacity of a corporation to defend against, mitigate, and recover from attacks can be assessed quantitatively, but the required data are not easily accessible, and dissemination to a risk-bearing market requires the implementation of new reporting structures.

5. Extreme events in software and systems development

Taleb [2007], in his highly provocative book *The Black Swan: The Impact of the Highly Improbable*, says that we are typically only capable of understanding the causes of extreme events after they happen. However, he further states that the hindsight understanding produced by assessing causes after an event is usually not general enough to yield insight, missing causal understanding, and often lacks meaningful impact on decisions. “Clear” hindsight without inventive foresight is insufficient to cope with risks of improbable and elusive events to the homeland. Indeed, it is the improbable, or phantom events, that result in the greatest impact (e.g., the failure of the financial information infrastructure in Estonia from organized hackers; the unimagined terrorist attacks with passenger planes; the implausible combinations of hurricane, infrastructure, and social demographics).

By their definition, disasters constitute extreme and catastrophic events (and disasters from cyber attacks are no exception); thus their probabilities and associated consequences defy any common expected value representation of risk. Many analysts and decision theorists are beginning to recognize a simple yet fundamental philosophical truth—in the face of such unforeseen calamities as bridges falling, national IT systems failing (due to terrorist or nation-sponsored attacks), airplanes crashing, tsunamis washing, and hurricanes landing with great force, we must acknowledge the importance of studying “extreme” events [Haimes 1991, 2004]. For Taleb [2007], an extreme event has three attributes:

- it is an outlier, as it lies outside the realm of regular expectation, because nothing in the past can convincingly point to its possibility;
- it carries an extreme impact; and
- in spite of its outlier status, human nature makes us concoct explanations for its occurrence after the fact, making it explainable and predictable.

6. Absence of a needed “hedgehog concept” with respect to cyber security

Collins [2001] argues that great companies organize themselves around a simple business idea or model—a *Hedgehog Concept*, based on a clear understanding of three key fundamentals: what they could excel at, what drives the economics of their business, and what they care about passionately. They focus tenaciously on those core concepts; they organize around them, and do not allow themselves to become distracted by other things. Indeed, the missing focus on cyber security during software engineering development and the overall IT systems integration constitutes a lack of a hedgehog in Collins’s approach to greatness and success. This deficiency is inextricably linked to the preceding five deficiencies.

C. The Steps of a Systemic Risk-based Methodological Approach

The multifaceted problem of cyber insecurity, the multiple parties involved in the delivery of information technology, and the myriad sources of risk to this system of systems—including inside intruders and outside hackers—defy a business-as-usual approach in addressing it. Building on a recognition of the six principal deficiencies of cyber insecurity described above, this section introduces steps for a systemic risk-based approach to bridging the multidisciplinary technical disciplines. Because hackers will eventually find exploitation opportunities for any significant point of vulnerability, such a coordinated and integrated approach must evolve over time throughout the software and system developmental life cycle. It is imperative that a systemic risk-based methodological approach address the multiple aspects and dimensions of the cyber security problem, in the following process steps:

- *Risk assessment*
- *Technology best practices*
- *Knowledge management mechanisms*
- *Risk management*
- *Coordinating roles of all stakeholders*

We expand on each of the five process steps as follows:

C.1. Step 1. Risk Assessment

This step entails performing life cycle-based risk assessment to identify vulnerabilities in the integrated IT system encompassing hardware-software integration, organizational interfaces, shared security, and network links. The step requires documentation and analysis of all sources of risk to cyber security, with a focus on all initiating events that are germane to the development of software-hardware engineering.

Lowrance [1976] defines risk as “a measure of the probability and severity of adverse effects.” Kaplan and Garrick [1981] were the first to formalize a theory of scenario structuring in the context of quantitative risk assessment with a trio of questions: What can go wrong? What is the likelihood? What are the consequences? Current risk assessment methods decompose a system into non-interconnected subsystems that can be assessed with typical measures of risk, then combined to form system-level measures [Kaplan, Haimes, and Garrick 2001]. Information technology, which makes almost all systems interconnected in our era, requires the training in risk analysis of current and future software engineers, computer scientists, and systems engineers involved in IT. Such training and added knowledge about what can go wrong and what failures can be expected with its associated consequences ought to consider the cross-system interfaces and distributed cyber security. Consider, for example, the dire economic and social consequences from security breaches of Supervisory Control and Data Acquisition (SCADA) systems, used in distributed process and digital control. These SCADA systems remotely control the operations of railroads, electric power plants, and utilities, among other critical infrastructures. Indeed, SCADA systems have increasingly been useful in improving the efficiency of operating interconnected industrial processes situated in remote locations. It is well-known that dependence of SCADA on telecommunications technologies makes processes more productive

and efficient, but also increases the vulnerability of the overall systems to anomalies, whether accidental or willful.

Risks are dynamic and will assuredly evolve across the IT system's various life cycle phases. Hence, a sound risk assessment process constitutes determining the extent to which risks can evolve during an IT system's life cycle, such as during hardware-software integration; organizational interfaces, shared security, network links, and processes for operating, maintaining, upgrading, or replacing the system. The capability maturity model (CMM) proposed by Sherer and Cooper [1994]¹ provides a mechanism with which to assess the vulnerability of an IT system to risk, and is comprised of five levels: initial, organized, integrated project management, quantitative management, and process optimization. Software CMM (SW-CMM) was the first formal CMM and has been adopted by companies engaged in software development worldwide. The SW-CMM is being replaced by a more encompassing CMM Integration (or CMMI).² The five levels are similar to the earlier versions but take on new and extended definitions (see CMMI overview)²: initial, managed, defined, quantitatively managed, and optimized.

The identification of all conceivable risk scenarios is at the heart of the risk assessment process. The deployment of hierarchical holographic modeling (HHM) and its derivatives, including the adaptive multiplayer HHM (AMP-HHM) game as an effective systemic process for risk identification, has been widely demonstrated [Haimes 2008, 2004, 1981; Haimes and Horowitz 2004]. The term *holographic* refers to the desire to have a multiview image of a system when identifying vulnerabilities (as opposed to a single view, or a flat image of the system). Views of risk can include, but are not limited to: economic, organizational, technical, political, and social. In addition, risks can be geography related and time related. In order to capture a holographic outcome, the team that performs the analysis must provide a broad array of experience and knowledge. The term *hierarchical* refers to the desire to understand what can go wrong at many different levels of the system hierarchy. HHM recognizes that for the risk assessment to be complete, one must realize that the macroscopic risks that are understood at the upper management level of an organization are very different from the microscopic risks observed at lower levels. In a particular situation, a microscopic risk can become a critical factor in making things go wrong. In order to carry out a complete HHM analysis, the team that performs it must include people who bring knowledge from up and down the hierarchy.

Perhaps one of the most valuable and critical aspects of HHM is its ability to facilitate the evaluation of the subsystem risks and their corresponding contributions to the risks in the total system. In the planning, design, or operational mode, the ability to model and quantify the risks contributed by each hardware-software subsystem markedly facilitates identifying, quantifying, and evaluating risk. In particular, HHM has the ability to model the intricate relationships among the various IT subsystems and to account for all relevant and important elements of risk and

1. Scherer, S.W., and J. Cooper, September 1994. *Software acquisition maturity*. Pittsburgh, PA, Carnegie Mellon University, Software Engineering Institute.

2. Capability Maturity Model, <http://www.sei.cmu.edu/cmm/>, accessed December 4, 2007.

2. Capability Maturity Model, accessed December 4, 2007.

<http://www128.ibm.com/developerworks/rational/library/content/RationalEdge/feb02/ConventionalToModernFeb02.pdf>,

uncertainty. This makes for a more tractable modeling process and results in a more representative and encompassing risk assessment process.

The AMP-HHM game is based not only on the actions of the multiple player teams and their consequences, but also on an explicit understanding of the inherent characteristics of the players that necessarily lead to the observed actions and consequences. For example, the strategies and actions of the homeland defenders (“Blue Team”) in the AMP-HHM game respond to the states of their own system as well as to those of the attackers (“Red Team”). Intelligence analyses for countering terrorism will be far more effective if they are driven not only by the symptoms (i.e., the actions of the terrorist networks), but also by the root causes (i.e., the states that characterize the terrorist networks). To this end, the AMP-HHM game also offers a roadmap for scenario tracking that accounts for the characteristics of both the root causes and the target (see Haimes [2002]; Horowitz and Haimes [2004]). A sample of risk scenarios that would be identified through the AMP-HHM game includes the sources of risks during the design, development, systems integration, deployment, operations, and maintenance of each sub-system that constitutes the integrated IT system; the likelihood of those sources of risks to happen; the multiscale economic and other consequences associated with those risks; the time domain associated with those risks.

C.2. Step 2. Technology best practices

This step identifies **current** modeling paradigms and states of practices in IT systems integration encompassing hardware technology, software engineering, and systems engineering. This step requires comprehensive understanding of:

- the major sub-systems that constitute the integrated IT system;
- the major model building blocks that can best represent each subsystem (e.g., state, decision, random, and exogenous variables; and inputs and outputs); and
- the processes and methodologies that are being used in the integration of hardware technology, software engineering, and systems engineering, and the responses of the integrated IT system to inevitable successful attack.

The elements and processes associated with IT system integration (hardware technology, software engineering, and systems engineering) are decomposable functionally and temporally. Functional decomposition enables analysis of a large-scale system into its component subsystems. For each subsystem, domain experts can identify the model building blocks (e.g., state, decision, random, and exogenous variables; and inputs and outputs) that subsequently describe the as-planned state of functionality and the events that can potentially lead to failure states. Reliability tools such as event trees and fault trees can help analyze the combination of subsystem failure events that can lead to system inoperability.

Temporal decomposition enables the analysis of system performance by examining the changes in model building blocks over time. Because IT systems are emergent, it is imperative to address the dynamic nature of systems functionality, technology development, reliance on COTS and their associated dynamic changes that often leave the users high and dry, software development, stakeholders objectives, hackers learning process, and so forth. For example, the Internet and Internet-exploiting systems, are emergent systems whose functions, requirements, organizational

structure, stakeholders, and even their architecture and configuration are not under the control of a design organization. Furthermore, such systems are dynamically changing at high speed, which creates a situation where the required software architecture and software development for the high performance computing technology, at least for some of the functionality of large-scale and complex systems, remain in limbo. Flexibility, robustness, and resilience in the software engineering design and implementation are critical to accommodate changes leading to effective functionality and enhanced cybersecurity.

Technology best practices call for a cultural, educational, organizational, and technical paradigm shift in hardware and software development that focuses on the risk to cyberspace through myriad malicious intruders.

C.3. Step 3. Knowledge management mechanisms

This step entails studying the prevailing educational, cultural, institutional, and organizational nature of the underpinning knowledge management that determines the effectiveness of systems integration.

The challenges associated with the prevailing educational, cultural, institutional, and organizational changes are neither trivial nor new, knowing that if evolution were to happen, it would be slow and not without pain and procrastination. For example, Longstaff and Haines [2002] argued six years ago that “very few institutions of higher education, if any, have responded so far to this need by offering relevant courses, revising their curricula, or by introducing undergraduate and graduate degree programs in this area. The knowledge that this specialized professional must acquire transcends traditional disciplines. Managing this needed knowledge is at the core of the vision, objectives, and programmatic plans of successful government and private-sector organizations”

To overcome this seemingly unmanageable challenge, lessons learned from institutions of higher education and the private sector ought to be shared and multiplied to be embraced by other organizations. For example, the School of Engineering and Applied Science at the University of Virginia has established a new joint educational degree program, Electrical and Computer Engineering, shared by two departments—Computer Science and Electrical and Computer Engineering. O’Neal [2004] comments on the prevailing compartmentalization of course offerings among major universities, leading to a lack of an overall understanding of the integrated software-hardware modeling, control, and integration. Further, he asserts a need for restructuring computer science degree curricula to incorporate interdisciplinary programs, which will expand employment opportunities for students and increase the competitiveness of faculty to secure national research funding. For example, national funding agencies such as the National Science Foundation favor collaborative research proposals.³

To generalize the findings from this effort, we must collect and build on various sources of information such as samples of academic curricula from various universities on software

3. “For example, by establishing collaborations with researchers in industry and government laboratories, developing partnerships with international academic institutions and organizations, and building networks of U.S. colleges and universities.” Source: <http://www.nsf.gov/pubs/2007/nsf07046/nsf07046.jsp>, accessed (insert date).

engineering, computer science, computational science and engineering (CSE), and other related disciplines. Such sources of information can be supplemented with a survey instrument yet to be designed and developed that can be distributed initially to the consortium members of the Institute for Information Infrastructure Protection (I3P) and later to a broader audience. Gap analysis will address cultural divides; hence the communications and ultimate systems integration challenges. Subsequently, the ingredients of an interdisciplinary program will be developed that are the by-product of the academic educational system that graduates most software engineers without formal courses in systems engineering, process control, and systems integration.

C.4. Step 4. Risk management

This step entails formulating risk management policy options to enhance the reliability of the integrated IT system against critical sources of risks. The step requires identification and analysis of:

- the options that are available to address multiscale risks;
- the trade-offs in terms of the associated costs, benefits, and risks; and
- the impacts of current decisions on future options.

Understanding the nature of risk to interdependent and complex systems and having the capability to model can result in improved risk management. Haimes [1991] presents a second trio of questions for quantitative analysis of risk management: What can be done and what options are available? What are the trade-offs in terms of costs, benefits, and risks? What are the impacts of current decisions on future options? Answers to these questions will enhance a decisionmaker's capacity to make appropriate decisions about acceptable levels of risk that are likely to evolve over time in increasingly complex and interdependent systems.

In order to model the trade-offs that exist among risk-reducing options, we need to understand how and to what degree the phases of the development process are interdependent. It is likely that phases exhibit multiple interdependencies. For any given analysis, a subset of particularly relevant interdependencies will tend to dominate the modeling activity, depending on the questions that have been asked and the decisionmaker who will ultimately use the analytical results for policy formulation. The role of the modeler is to isolate the relevant interdependencies and build analytical and other tools to address the questions asked by decisionmakers to aid in formulating policy. All other couplings are considered only to the extent that they shape or constrain processes between and among the software development phases. Indeed, understanding the complex, interdependent nature of relationships among various phases of the software development process is central to the risk analysis process.

To manage the potential breaches of IT security, we need a principled approach of risk management to justify investments in cyber security research [Haimes and Chittister 2005]. Justifying those investments is not limited to organizations, because every computer user, whether individual or institutional, has to make difficult choices about investments in cyber security, balancing risk with other spending priorities. *However, from the broader "social compact perspectives," the initial and most important investment in cyber security, and thus in IA, resides within the development of reliable software engineering and the proper subsequent*

systems integration. This also means reversing the practice of “circling the wagons” in the quest to achieve cybersecurity and considering cybersecurity during software engineering development as par with cost and performance. Such an approach also requires changing the organizational culture that often separates the education, training, and experience of software engineers, systems engineers, and hardware engineers and scientists.

C.5. Step 5. Coordinating roles of all stakeholders

This step entails identifying the role of professional and nonprofit organizations, such as the I3P, in contributing to coordinating educational and training roles of all stakeholders, focusing on the development of integrated curriculum in response to the cyber security and information assurance needs and procedures.

This step requires identifying, mapping, and evaluating research contributions, curriculum innovations, and outreach activities common to three groups engaged in cybersecurity risk management:

- developers of large-scale software-based systems, specifically those arising in computational science and engineering (CSE);
- software engineers; and
- system integrators and systems engineers, who bridge technical and cultural divides between the two groups.

This approach characterizes innovations in several areas, the most important being the education and cultural background, knowledge, expertise, and experience of the three groups.

D. The Roles of Universities and Other Institutions of Higher Education and Research

Very few engineering and applied science schools in the United States and around the world have acknowledged in their curricula the imperative need to respond to the complexity and the cross-disciplinary nature of the integrated development of hardware-software systems. Consequently, even fewer schools have considered extending, expanding, and integrating courses that span the professional disciplines involved in the architectural design, development, and maintenance of hardware-software systems. Consider, for example, the subject of modeling. Because software today assumes the role of systems integration (increasingly displacing hardware in that role), basic knowledge in systems modeling seems to be an imperative topic of learning for all software engineers who oversee systems integration.

If the number of universities and other institutions of higher education that incorporate such courses in their curricula is very limited, the questions are: Why is there lack of leadership and incentives? and What are the long-term implications for cyber security and information assurance? Could one not expect a less chaotic state of affairs in cyber security for government and industry in dealing with cyber intrusion, spying, and denial of service if universities and other institutions of higher education were to treat this subject more seriously, methodically, and systemically? Is it too utopian to suggest that universities initiate such a change in our approach to a more holistic framework in the development of integrated hardware-software systems? Indeed, although universities seem to be the most logical and natural places for educating a cadre of cross-disciplinary professionals, present trends and evidence do not support this premise. Today, most university faculty members are prisoners of their own disciplines and institutional

department setups. The present university system of rewards, including promotion and tenure, provides strong disincentives for change, especially if initiated by a young, developing faculty.

On the other hand, universities seem more likely to respond to government initiatives, especially when adequately funded, or to follow the strong, emerging leadership of professional organizations such as the U.S. National Academies and the numerous professional societies, than to initiate and assume the leadership role themselves. Universities today seem to have lost some of their ability to identify urgent educational needs and the pedagogical framework that is conducive to cross-disciplinary learning and growth; with this, they have also lost some of their ability to pioneer responses to emerging societal challenges.

Who should take the initiative in a more holistic pedagogical approach to integrated hardware-software systems—universities, industry, government agencies, or professional societies and organizations? The need is urgent for one or more of these parties to take such an initiative, support it, and provide the political will to ensure its realization. However, only with long-term investment in the education of our new engineering and science students will we be able to expect an appropriate incorporation of the needed cross-disciplinary educational and training programs within a holistic approach to achieve a more harmonious interplay and collaboration among the various science and engineering disciplines on this subject.

E. The Roles of the Institute for Information Infrastructure Protection

Because universities and other institutions of higher education and advanced research are the source of the cyber insecurity and lack of information assurance problem and, at the same time, they are also the leading candidates to provide enduring solutions, we need a third party to energize the change. However, this third party cannot be too divorced from the university culture, structure, and mode of operation. The Institute for Information Infrastructure Protection (I3P) is a natural intermediary through which the actions for change introduced in this paper are more likely to be realized.

The I3P is a consortium of leading national cyber security institutions, including academic research centers, government laboratories, and nonprofit organizations. It was founded in September 2001 to help meet a well-documented need for improved research and development (R&D) to protect the nation's information infrastructure against catastrophic failures. A main role of the I3P is to coordinate a national cyber security R&D program and help build bridges among academia, industry, and government. The I3P is working to identify and address critical research problems in information infrastructure protection and open information channels among researchers, policymakers, and infrastructure operators. Although historical, legal, and cultural problems prevent some research organizations from working together, the I3P has overcome many of those obstacles to reach new levels of cooperation. The I3P consortium has assumed a major role in helping to untangle such cyber security issues as infrastructure interdependencies, or systems of systems, and it is beginning to investigate cyber security related policy, risk, and economic issues.

Given its unique composition and role, the I3P may consider assuming leadership in the development of educational and training resources to bridge the technical disciplines engaged in systems and software development. Such initiatives could address traditional university curricula of electrical and computer engineering, computer science, systems engineering, and engineering management, as well as the certification requirements of professional societies that play a role in

systems and software development. The I3P may develop a monitoring and measurement plan for realizing progress in harmonizing the technical disciplines that are engaged in systems and software engineering development.

F. Epilogue

Generally, complex problems do not have simple solutions, especially when an important ingredient of the solution entails the need for change in organizational culture. Nevertheless, when the consequences of inaction become dire, as is the case with the worldwide cyber insecurity and lack of information assurance, such change may have to take place, albeit at the beginning it may be slow and piecemeal. As a starting point, a well-designed and structured national (or, preferably, international) survey of the relevant curricula, spanning the topics discussed in this paper, of all universities and institutions of higher education could achieve multiple purposes. The survey should be based on the principal deficiencies and methodological steps identified in this white paper. Conducting the survey, analyzing the responses, and widely disseminating its findings would:

- place appropriate emphasis on the urgency and centrality of the educational and training dimension to achieving markedly better cyber security and information assurance;
- initiate serious dialogue among the principal stakeholders on the need for and management of change;
- enable universities and other institutions of higher education and advanced research, and especially faculty, to marshal the necessary technical and political will to address this need for change;
- inform policymakers in Congress and the administration on the need to fund new more-integrated educational and continuing education programs in response to the needs discussed here;
- illuminate areas of success and areas of deficiency in addressing the gaps across technical disciplines; and
- encourage individuals and organizations to accept the fact that circling the wagons and responding to cyber intrusions by patching can provide a reprieve only until the next episode, and that a more effective defense against cyber intrusion is a cross-disciplinary team effort that spans the entire system's life cycle including development of: 1) the architectural specification, 2) designs, 3) software engineering, 4) systems integration plans, and 5) conduct of operations and maintenance.

Acknowledgments

The authors are grateful to their I3P colleagues for their many contributions to the ideas developed in this paper. The research that led to this paper was supported, in part, by the I3P.

References

Asbeck, E., and Y. Y. Haimes, 1984. The partitioned multiobjective risk method, *Large Scale Systems*, 6(1): 13-38.

- [Ashkenas](#), R.N., [D. Ulrich](#), [T. Jick](#), and [S. Kerr](#), 1998. *The boundaryless organization: Breaking the chains of organizational structure*. Hoboken, NJ: Jossey-Bass.
- Bier, V.M., Y.Y. Haimes, N.C. Matalas, J.H. Lambert, and R. Zimmerman, 1999. A survey of approaches for assessing and managing the risk of extremes. *Risk Analysis*, 19(1): 83-94.
- Boehm, B.W., 1981. *Software engineering economics*. Englewood Cliffs, NJ: Prentice Hall.
- , 1988. A spiral model for software development and enhancement. *Computer*, (May): 61-72.
- , 2006. Some future trends and implications for systems and software engineering processes. *Systems Engineering*, 9(1): 1-19.
- Boehm, B., and C.A.R. Hoare (eds.), 1975. *Proceedings 1975 International Conference on Reliable Software*. (April 1975). New York: ACM/IEEE.
- Boehm, B. and A. Lane, 2007. Final report: Integrating systems and software engineering project, Center for Systems and Software Engineering, University of Southern California, Los Angeles, CA..
- Brooks, F., 1995. *The mythical man-month*, 2nd ed. Reading, MA: Addison Wesley.
- Chittister, C.G., and Y.Y. Haimes, 1996. Systems integration via software risk management. *IEEE Transactions on Systems, Man, and Cybernetics*, 26(5): 521-532.
- Chittister, Clyde, and Y.Y. Haimes, 2008. Harmonizing high performance computing (HPC) with large-scale and complex (LSC) systems. Submitted to *Systems Engineering*. (date? forthcoming? Otherwise, it usually is not listed in references, or listed as unpublished.)
- Collins, Jim, 2001. *Good to Great*, HarperCollins Publishers, Inc. NY, NY.
- Collins, Jim and Jerry Porras, 1994. *Built to Last*, HarperBusiness NY, NY.
- George, Dickie, 2008. I3P Consortium Meeting, National Institute of Standards and technology (NIST), Gaithersburg, MD.
- Gerrard, P., and N. Thompson, 2002. *Risk-based e-business testing*. City: Artech House.
- Goodman, Seymour E., and Herbert S. Lin (eds.), 2007.(2006? See p. 6) *Toward a safer and more secure cyberspace*. Washington, DC: National Academy Press.
- Haimes, Y.Y., 1981. Hierarchical holographic modeling, *IEEE Transactions on Systems, Man, and Cybernetics*, 11(9): 606–617.
- , 1991. Total risk management, *Risk Analysis* **11**(2), 169–171.
- , 2002. Roadmap for modeling risks of terrorism to the homeland, *Journal of Infrastructure Systems*, 8(2): 35-41, June.

———, 2004. *Risk Modeling, Assessment, and Management*. 2nd ed. Hoboken, NJ: Wiley.

———, 2008. *Risk Modeling, Assessment, and Management*. 3rd ed. Hoboken, NJ: Wiley.

Haimes, Y.Y., and C.G. Chittister, 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., and B. Horowitz, 2004. Adaptive two-player hierarchical holographic modeling game for counterterrorism intelligence analysis. *Journal of Homeland Security and Emergency Management*: 1(3).

Haimes, Y.Y., R.M. Schooff, and C.G. Chittister, 1997. A holistic management framework for software acquisition. *Acquisition Review Quarterly*, Winter 1997: 55-85.

Horowitz, B., and Y. Y. Haimes, 2003, Risk-based methodology for scenario tracking for terrorism: a possible new approach for intelligence collection and analysis, *Systems Engineering* 6(3), 152–169.

Huang, L., 2005. A value-based process achieving software dependability. *Proceedings Software Process Workshop*. May. (in Boehm [2006]. (Does this second reference serve a purpose here?))

Humphrey, W., 2000. *Introduction to the team software process*. Reading, MA: Addison Wesley.

Jackson, Daniel, 2002. Alloy: A lightweight object modeling notation. *ACM Transactions on Software Engineering Methodology*. 11(2): 256-290.

———, 2006. Dependable software by design. *Scientific American*, 294(6): 69-75.

Kaplan, S., and B. J. Garrick, 1981, On the quantitative definition of risk, *Risk Analysis* **1**(1), 11–27.

Kaplan, S., Y. Y. Haimes, and B. J. Garrick, 2001, Fitting hierarchical holographic modeling into the theory of scenario structuring and a resulting refinement of the quantitative definition of risk, *Risk Analysis* **21**(5), 807–815.

Longstaff, Thomas A., Clyde Chittister, Richard Pethia, and Yacov Y. Haimes, 2000.(2002? See p. 3) Are we forgetting the risks of information technology? *Computer: Innovative Technology for Computer Professionals*, December: 43-51.

- Longstaff, T.A., and Y.Y. Haimes, 2002. A holistic roadmap for survivable infrastructure systems. *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans*, 32(2): 260-268.
- Longstaff, T.A., Y.Y. Haimes, and C. Sledge, 2002. Are we forgetting the risk of COTS products in wireless communications? *Risk Analysis*, 22(1): 1-6.
- Lowrance, W.W., 1976, *Of Acceptable Risk*, William Kaufmann, Los Altos, CA.
- McDougall, P., 2008. Societe Generale's 'hacker' trader had only limited computer skills, *InformationWeek*, January 25.
- National Research Council, 1996. *Shopping for safety: Providing consumer automotive safety information*, Special Report 248, Transportation Research Board. Washington, DC: National Academy Press.
- , 2002. *Making the nation safer: The role of science and technology in countering Terrorism*. Washington, DC: National Academy Press.
- National Research Council (NRC), Committee on Information System Trustworthiness, 2002. *Making the Nation Safer*. Washington, DC: National Academy Press. (is this the same as the previous reference?)
- O'Neal, M., 2004. Restructuring computing programs to meet employment challenges. *IEEE Computer*, November: 29-34.
- Rosenbloom, P.S., 2004. A new framework for computer science and engineering. *IEEE Computer*, November: 23-28.
- Taleb, N.N., 2007. *The black swan: The impact of the highly improbable*. New York: Random House. (please fix indent)
- Traynor, I., 2007. Russia accused of unleashing cyberwar to dusable Estonia, *The Guardian*, May 7.
- Womack, J.P., and D.T. Jones, 1996. *Lean thinking: Banish waste and create wealth in your corporation*. New York: Simon & Schuster.
- Wulf, Wm., and A. Jones, 2004. A perspective on cybersecurity research in the United States. In *Terrorism: Reducing Vulnerabilities and Improving Responses*. Washington, DC: National Research Council of the National Academies.