Evaluating Software Assurance Knowledge and Competency of Acquisition Professionals

Dan Shoemaker, University of Detroit Mercy
Nancy R. Mead, SEI

Abstract. As the potential for highly destructive cyberattacks grows, organizations must ensure that their procurement agents acquire high quality, secure software. ISO 12207 and the Software Assurance Competency Model, when used together, provide a clear view of the activities, knowledge, and competencies required to procure secure software.

The Benefit of Standardized Acquisition

Gartner forecasts that the worldwide dollar-valued IT spending forecast will grow 3.1% in 2014, reaching $3.8 trillion [1]. Considering the magnitude of this investment, organizations should work hard to ensure the effective acquisition of systems and software. This task is a complicated one; the success or failure of any acquisition effort depends on the capability of the individuals who do the work, and those individuals' capability depends on knowledge and experience. While an experienced and knowledgeable procurement agent may deliver the desired result, one who is inexperienced or incapable may bring about a disaster. Establishing capability requirements for every person involved in the acquisition process is vital for organizations to preserve their investment in technology.

It is essential to use standard criteria to judge the performance of any task; standard criteria allow actions to be judged objectively. Having a standard set of criteria also ensures coordinated management of the process. Though the benefits of coordinated management are manifold, the primary advantage is that defining a process enables repeatability. Looking back, the entire decade of the 1990s seems to have been devoted to detailing the benefits of repeatable processes. That thinking was probably best expressed in A Discipline for Software Engineering [2]. The justification for a well-defined, documented, and systematically executed process is that it can be more effectively managed and continuously improved [2]. A single, comprehensive set of standard criteria to guide the work also ensures efficient communication between participants, which, in turn, ensures a more suitable final product.

Repeatability requires consistent execution of the fundamental activities of a process. According to conventional wisdom within the software industry, standards convey those requisite activities. Standards define the fundamental requirements for the performance of a given process. A properly written and administered standard will ensure that every participant in the process knows and follows principles and practices that have track records of success. Concerning this discussion, there are several official and quasi-official standards for acquisition. The standards for acquisition include IEEE 1062-1998, an eight-page collection of high-level recommendations for ensuring quality in software acquisition [3]. There are guidelines that provide recommendations for the security testing of government off-the-shelf (GOTS) and commercial off-the-shelf (COTS) products [4]. However, these recommendations in no way constitute a complete process. The Common Body of Knowledge to Produce Acquire and Sustain Secure Software itemizes a complete set of principles and practices for secure acquisition [5]. However, this white paper does not provide general guidance [6].

The almost total absence of comprehensive lifecycle recommendations for acquisition might be explained by the dominant role of ISO 12207-2008, both internationally and in the United States [7]. That standard documents a comprehensive set of activities and supporting tasks to establish effective lifecycle acquisition of system and software products. The standard dictates a complete set of highly interdependent lifecycle activities for proper execution of the supply and reuse process, in addition to explicit acquisition recommendations. The standard also provides comprehensive advice about to how to carry out the ancillary activities that are necessary to support those processes, such as documentation, software quality assurance, and configuration management.

Factoring the 21st Century into the Equation

All of the existing standards for acquisition could serve as a basis for structuring a repeatable lifecycle acquisition function. However, with the exception of the Common Body of Knowledge to Produce Acquire and Sustain Secure Software, they are all oriented toward assurance of product quality. Though most of the standard activities associated with product quality (e.g., planning, testing, reviews, audits) still have currency in this discussion, the ever-increasing threats in cyberspace have added a new dimension to the requirements for a capable and successful procurement process. Thus, it is critical that acquirers adopt and follow assurance practices to ensure that products not only operate as intended, but also have sufficient integrity to withstand attack.

The need for secure products makes the problems associated with ensuring the quality of the purchased product almost nostalgically simple. A recent report summarizes the security issues facing all acquirers; the report uses five categories—each with a different implication for acquirers—to classify these concerns [8]:

- **installation of malicious logic on hardware or software**
- **installation of counterfeit hardware or software**
- **failure or disruption in the production or distribution of critical products or services**
- **reliance upon a malicious or unqualified service provider for the performance of technical service**
- **installation of unintentional vulnerabilities on software or hardware**

These categories highlight a central question: "Do acquisition personnel have the capability to ensure that purchased system and software products are free of these threats?"

Though the past decade has produced a number of acceptable methods for assuring the security of the product [9, 10], ensuring the ability of the individual worker to apply these approaches

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
OCT 2014		**00-00-2014 to 00-00-2014**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Evaluating Software Assurance Knowledge and Competency of Acquisition Professionals	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Carnegie Mellon University,Software Engineering Institute,Pittsburgh,PA,15213**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
As the potential for highly destructive cyberattacks grows, organizations must ensure that their procurement agents acquire high quality, secure software. ISO 12207 and the Software Assurance Competency Model, when used together, provide a clear view of the activities, knowledge, and competencies required to procure secure software.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**Same as Report (SAR)**	**4**	

is difficult. A new model from the Carnegie Mellon University Software Engineering Institute (SEI), the Software Assurance Competency Model, establishes a foundation for assessing the capability of software assurance professionals [11]. The model can be used by individuals to assess their own capabilities and professional goals, and by organizations to assist in staffing and building teams with appropriate competencies. At present, there is not a competency exam associated with the model, which is intended to be instantiated by organizations for their own use.

This model, which has been endorsed by the IEEE Computer Society, portrays the requisite competencies for software assurance work across a range of knowledge areas [11]. The competency areas captured in this model are 1) Assurance Across the Lifecycle, 2) Risk Management, 3) Assurance Management, 4) Assurance Assessment, 5) System Security Assurance, 6) System Functionality Assurance, and 7) System Operational Assurance. The model is further decomposed into individual units based on knowledge and skills. Those knowledge and skill units can be ranked at competency levels 1 through 5 [9]. The Software Assurance Competency Model provides a common definition of the activities required to ensure a secure product, and it uses a competency-based evaluation scheme. The model's knowledge and competency stipulations can be combined with the acquisition process recommendations from ISO 12207 to define a set of standard, competency-based acquisition processes for any organization. This amalgamation can then be used to judge whether a given acquisition process is being performed at a sufficient level of capability.

A Competency-Based Model for Secure Acquisition Practice

The SEI Software Assurance Competency Model comprises seven competency areas, which are decomposed into 20 knowledge units. Some of these knowledge units are devoted to elements of software work that do not involve acquisition. However, 13 of those 20 units can apply to ensuring a secure acquisition: Software Lifecycle Processes, Software Assurance Processes and Practices, Risk Management Concepts, Risk Management Processes, Software Assurance Risk Management, Assurance Assessment Concepts, Measurement for Assessing Assurance, Making the Business Case for Assurance, Managing Assurance Compliance Considerations, Assurance Ethics and Integrity in Creation, Acquisition, and Operation of Software, Systems Assurance Technology, Assurance in Acquisition, Operational Monitoring, System Control, and Operational Procedures [11].

Each of these knowledge units is tied to a staged set of competencies. Table 1 provides the general definition of these requisite abilities [11].

Integrating Standard Acquisition Practices with Competency Requirements

The areas in the SEI Software Assurance Competency Model cover the entire software and system assurance process. Though the SEI model does not specifically designate competencies for acquisition, ISO 12207 does specify an end-to-end set of acquisition practices. These practices have been standardized since 1995 [7]. Table 2 summarizes required practices for ISO 12207.

L1 – Technician	Possesses technical knowledge and skills, typically gained through a certificate or an associate degree program, or equivalent knowledge and experience
L2 – Professional Entry Level	Possesses application-based knowledge and skills and entry-level professional effectiveness; may also manage a small internal project, supervise and assign L1 personnel, supervise and assess system operations, and implement accepted assurance practices
L3 – Practitioner	Possesses breadth and depth of knowledge, skills, and effectiveness; may also set plans, tasks, and schedules for in-house projects and may define and manage such projects and supervise teams on the enterprise level
L4 – Senior Practitioner	Possesses breadth and depth of knowledge, skills, and effectiveness and a variety of work experiences; has 5 to 10 years of experience and professional development; identifies and explores software assurance practices, manages large projects, interacts with clients
L5 – Expert	Advances the field by developing, modifying, and creating methods, practices, and principles at the organizational level or higher; has peer/industry recognition

Table 1: Staged Competencies for Each Knowledge Unit

1. Initiation • Prepare a concept or a need to acquire, develop, or enhance a product or service • Prepare a set of requirements, including relevant design, testing, and compliance standards • Prepare a risk and cost-benefit analysis for acquisition • Prepare a set of acceptance criteria and criteria for evaluation Prepare acquisition plan based on requirements, analyses, and criteria defined in prior steps
2. Request for Proposals • Document acquisition requirements depending on acquisition option selected • Define contract milestones • Specifically delegate implementation of requirements to responsible organizational entity
3. Contract Preparation and Update • Establish plans for supplier selection • Institute and carry out a negotiation process including contract preparation • Institute a process for change control
4. Supplier Monitoring • Prepare a plan for supplier review • Systematically review supplier during product preparation period
5. Acceptance and Completion • Perform acceptance reviews and testing • Institute systematic configuration management

Table 2: Standard Acquisition Steps for ISO 12207

Together, ISO 12207 and the SEI Software Assurance Competency Model describe the skills and competencies required to execute a software acquisition process. The complete set of acquisition practices specified in ISO 12207 can be combined with the knowledge units and competencies from the SEI Software Assurance Competency Model to provide an assurance knowledge and competency-based description for the standard activities of software and system acquisition.

Table 3 presents a suggested amalgamation of the ISO 12207 acquisition process requirements with the standard knowledge units of the SEI Software Assurance Competency Model (note: 12207 practices are in bold and SEI SwA Competency practices are in italics). The associated SwA Competency levels can be added to each of the individual SEI knowledge units based on the needs of the situation.

ISO 12207 Practice	SEI Software Assurance Competency Model Practice
Prepare a concept or a need to acquire, develop, or enhance a product or service	○ *Software Lifecycle Processes* ○ *Making the Business Case for Assurance* ○ *Ethics and Integrity in Creation, Acquisition, and Operation*
Prepare a set of requirements including relevant design, testing and compliance standards	○ *Software Assurance Processes and Practices* ○ *Risk Management Concepts* ○ *Risk Management Processes* ○ *Software Assurance* ○ *Risk Management Assurance Assessment Concepts* ○ *Measurement for Assessing Assurance*
Prepare a risk and cost-benefit analysis for acquisition	○ *Risk Management Concepts* ○ *Risk Management Assurance Assessment Concepts* ○ *Making the Business Case for Assurance* ○ *Assurance in Acquisition*
Prepare a set of acceptance criteria and criteria for evaluation Software Lifecycle Processes	○ *Software Assurance Processes and Practices* ○ *Risk Measurement for Assessing Assurance* ○ *Assurance in Acquisition* ○ *Operational Monitoring*
Prepare acquisition plan based on requirements, analyses, and criteria defined in prior steps	○ *Software Lifecycle Processes* ○ *Risk Management Concepts* ○ *Risk Management Processes* ○ *Software Assurance* ○ *Managing Assurance Compliance Considerations* ○ *Assurance in Acquisition* ○ *Operational Monitoring*
Document acquisition requirements depending on acquisition option selected	○ *Software Assurance Processes and Practices* ○ *Risk Management Processes* ○ *Software Assurance* ○ *Risk Management Assurance Assessment Concepts* ○ *Measurement for Assessing Assurance*
Define contract milestones	○ *Software Lifecycle Processes* ○ *Software Assurance Processes and Practices* ○ *Managing Assurance Compliance Considerations*
Specifically delegate implementation of requirements to responsible organizational entity	○ *Software Lifecycle Processes* ○ *Management Concepts* ○ *Risk Management Processes* ○ *Software Assurance* ○ *Risk Management Assurance Assessment Concepts*
Establish plans for supplier selection	○ *Making the Business Case for Assurance* ○ *Managing Assurance Compliance Considerations* ○ *Ethics and Integrity in Creation, Acquisition, and Operation* ○ *Assurance in Acquisition*
Institute and carry out a negotiation process including contract preparation	○ *Risk Management Processes* ○ *Software Assurance* ○ *Assurance in Acquisition*
Institute a process for change control	○ *Software Lifecycle Processes* ○ *System Control* ○ *Operational Procedures*
Prepare a plan for supplier review	○ *Software Assurance Processes and Practices* ○ *Risk Management Concepts* ○ *Risk Management Processes* ○ *Software Assurance* ○ *Risk Management Assurance Assessment Concepts* ○ *Measurement for Assessing Assurance* ○ *Systems Assurance Technology* ○ *Assurance in Acquisition* ○ *Operational Monitoring* ○ *System Control*
Systematically review supplier during product preparation period	○ *Management Processes Software Assurance* ○ *Measurement for Assessing Assurance* ○ *Managing Assurance Compliance Considerations* ○ *Systems Assurance Technology* ○ *Assurance in Acquisition* ○ *Operational Monitoring* ○ *System Control* ○ *Operational Procedures*
Perform acceptance reviews and testing	○ *Measurement for Assessing Assurance* ○ *Systems Assurance Technology* ○ *Assurance in Acquisition* ○ *System Control*
Institute systematic configuration management	○ *Systems Assurance Technology* ○ *Operational Monitoring* ○ *System Control* ○ *Operational Procedures*

Table 3: Creating a Competency-Based Model of Secure Acquisition Practice

Conclusion

The ability to guarantee a secure acquisition is far too important to the well-being of any organization to base its activities on individual virtuosity. Therefore, there is justification for a well-defined model of practice. ISO 12207 provides a commonly accepted statement of the complete set of practices necessary to conduct system and software acquisition. The acquisition activities and tasks specified in this standard have been accepted as correct for almost two decades [7]. The Software Engineering Institute has provided a model of the knowledge and competency levels needed to assure software and systems. Combining ISO 12207 and the Software Assurance Competency model to form a single description of the activities, knowledge, and competencies required to procure secure software and systems benefits the community as a whole.

The potential for highly destructive attacks directed through acquired software and system products is a reality in cyberspace. Whether the adversary is a nation state or a single hacker, it is presently far too easy to cause serious harm through the insertion of malicious and counterfeit objects into purchased software and systems. The inclusion of such tainted products in our national infrastructure could potentially threaten our way of life. Given the swiftness of technological change, it is excusable that organizations might not recognize the emerging importance of purchased software and systems. It is inexcusable, however, to know that threats exist and to stand idly by without doing anything about the situation. This paper suggests one approach organizations can take to better ensure the security of the products they buy.

Acknowledgments/Disclaimers:

Copyright 2014 Carnegie Mellon University

This material is based upon work funded and supported by the DoD under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

No warranty. This Carnegie Mellon University and Software Engineering Institute material is furnished on an "as-is" basis. Carnegie Mellon University makes no warranties of any kind, either expressed or implied, as to any matter including, but not limited to, warranty of fitness for purpose or merchantability, exclusivity, or results obtained from use of the material. Carnegie Mellon University does not make any warranty of any kind with respect to freedom from patent, trademark, or copyright infringement.

This material has been approved for public release and unlimited distribution.

DM-0001096

ABOUT THE AUTHORS

Daniel P. Shoemaker, Ph.D., is Principal Investigator and Senior Research Scientist at UDM's Center for Cyber Security and Intelligence Studies. Dan is also a full time Professor and former Department Chair at University of Detroit Mercy.

Phone: 313-680-1434
E-mail: dan.shoemaker@att.net

Nancy R. Mead is a Fellow and Principal Researcher at the Software Engineering Institute (SEI). Mead is also an Adjunct Professor of Software Engineering at Carnegie Mellon University. She is currently involved in the study of security requirements engineering and the development of software assurance curricula.

Mead has more than 150 publications and invited presentations, and has a biographical citation in Who's Who in America. She is a Fellow of the Institute of Electrical and Electronic Engineers, Inc. (IEEE) and a Distinguished Member of the Association for Computing Machinery (ACM). Dr. Mead received her PhD in mathematics from the Polytechnic Institute of New York, and received a BA and an MS in mathematics from New York University.

Phone: 412-818-3454
E-mail: nrm@sei.cmu.edu

REFERENCES

1. Gartner Worldwide IT Spending Forecast <http://www.gartner.com/technology/research/it-spending- forecast/>
2. Humphrey, Watts. A Discipline for Software Engineering. Reading, MA: Addison-Wesley, 1995.
3. Institute of Electrical and Electronic Engineers. IEEE Recommended Practice for Software Acquisition, (IEEE Std 1062, 1998 Edition [R2002]). New York: IEEE, 1998.
4. Roback, Edward A. NIST Special Publication 800-23: Guidelines to Federal Organizations on Security Assurance and Acquisition/Use of Tested/Evaluated Products. Gaithersburg, MD: National Institute of Standards and Technology, 2000.
5. Redwine, Sam. Software Assurance: A Curriculum Guide to the Common Body of Knowledge to Produce, Acquire and Sustain Secure Software. U.S. Department of Homeland Security, 2007.
6. Duncan, Scott. IEEE Software and Systems Standards Committee (S2ESC) Meeting Report. San Diego, CA, February 13-15, 2014. Web. 10 Mar. 2014. <http://asq.org/software/about/chairfeb06-software.html>
7. International Standards Organization. ISO/IEC 12207:2008, Systems and software engineering -- Software life cycle processes. Geneva: ISO, 2008.
8. United States Government Accountability Office. IT Supply Chain: National Security-Related Agencies Need to Better Address Risks (GAO Report to Congressional Requesters). United States Government Accountability Office, 2012.
9. Woody, Carol and Ellison, Robert J. Improving Software Assurance. Published 1 April, 2010, revised 5 July 2013. Web. 10 Mar. 2014. <https://buildsecurityin.us-cert.gov/articles/knowledge/assurance-cases/improving-software-assurance>
10. Davis, Noopur. Secure Software Development Life Cycle Processes. Published 5 July 2006, revised 31 July 2013. Web. 10 Mar. 2014 <https://buildsecurityin.us-cert.gov/articles/knowledge/sdlc-process/secure-software-development-life-cycle-processes>
11. Hilburn, Thomas; Ardis, Mark; Johnson, Glenn; Kornecki, Andrew; and Mead, Nancy R. Software Assurance Competency Model, Software Engineering Institute (CMU/SEI-2013-TN-004). Pittsburgh: Carnegie Mellon Software Engineering Institute, 2013.