The DHS Master of Software Assurance Curriculum Project

Nancy Mead, Rick Linger, Carol Sledge
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA
nrm, rlinger @sei.cmu.edu

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

Agenda

Project Background

Overview of Curriculum Project

Outcomes

Prerequisites**

Curriculum Architecture

Core Body of Knowledge**

Next Steps

Curriculum Resources

Acronyms/Abbreviations; Glossary; References

** Information for review – see also notes pages and MSwA Reference Curriculum report.

Project Background

Sponsorship and Goal

Sponsored by Department of Homeland Security (DHS) National Cyber Security Division (NCSD)

Activity led by Software Engineering Institute (SEI) at Carnegie Mellon University

Goal

- develop a curriculum for a Master in Software Assurance degree program and
- define transition strategies for future implementation

Curriculum Context

Discipline of Software Assurance (SwA) targeted specifically to the

- security and correct functioning of software systems
  - whatever their origins,
  - subject matter, or
  - operational environments

Audiences for Curriculum

Faculty responsible for *[Primary]*

- design, development, and maintenance of

graduate software engineering programs focusing on

- software assurance knowledge and practices

Those in development & acquisition organizations responsible for

- staffing positions in software assurance, or
- providing current software engineers with increased software assurance capabilities

Purpose of MSwA Curriculum Project

Develop and present a core body of knowledge that can be drawn from to create

- standalone Software Assurance degree program
- track within existing Master's degree programs
  - Software Engineering; Computer Science

Foundational material includes (but not limited to)

- Graduate Software Engineering 2009 *[GSwE 2009]* Curriculum Guidelines for Graduate Degree Programs in Software Engineering [iSSEc 2009]
- work done by SEI in support of U.S. DHS Build Security In (BSI) website [DHS 2010a]
- Software Assurance Curriculum Body of Knowledge (SwACBK) [DHS 2010b]

Overview of Master of Software Assurance Curriculum Project

MSwA Project Primary Objectives

Improve the state of the Software Assurance Education

Develop a Master of Software Assurance Reference Curriculum

Identify educational offerings at other levels

- undergraduate
- community colleges

Definition: Software Assurance [CNSS]

Committee on National Security Systems definition:

> Software assurance is the level of confidence that software is free from vulnerabilities, either intentionally designed into the software or accidentally inserted at any time during its life cycle, and that the software functions in the intended manner.

Started with this definition and modified it for the curriculum project

[CNSS 2009]

Definition: Software Assurance [MSwA]

Master of Software Assurance Curriculum Project definition:

Application of technologies and processes to achieve a required level of confidence that software systems and services

function in the intended manner,

are free from accidental or intentional vulnerabilities,

provide security capabilities appropriate to the threat environment, and

recover from intrusions and failures.

[MSwA 2010]

Implied Differences: MSwA Curriculum

Areas of special emphasis and unique properties that distinguish the MSwA Curriculum from traditional software engineering and computer science programs include a focus on

- software *and services*
- development *and acquisition*
- security and *correct functionality*
- *software analytics*
- *system operations*
- *auditable evidence*

Process Used to Develop Curriculum -1

1. Develop project objectives
2. Identify and review sources
3. Define topics
   - software security practices that span the Software Development Life Cycle (SDLC) (considered in all life-cycle phases)
   - requirements engineering practices
   - architecture and design practices
   - Coding, verification, and testing practices
   - analysis of software and services in static and operational contexts
   - assembly, evolution, and deployment
   - risk mitigation strategies for complex systems
   - governance and management practices

Process Used to Develop Curriculum -2

4. Define software development life cycle practices and categories

   Four conceptual categories to help understand & assess the practices

   - security assurance
   - functionality assurance
   - operations assurance
   - assurance processes and management

5. Solicit external feedback
6. Develop outcomes and core Body of Knowledge (BoK)
7. Compare knowledge units to practices

Body of Knowledge (BoK)

Organized as BoK knowledge areas →knowledge units →knowledge topics, with associated Bloom cognitive levels.

Assurance Process and Management

- Assurance Across Life Cycles
- Risk Management
- Assurance Assessment
- System Operational Assurance

Assurance Product and Technology

- System Security Assurance
- System Functionality Assurance
- System Operational Assurance

Outcomes

Outcomes: Graduating Student

Outcomes

- Specify knowledge, skills, and capabilities that graduates of an MSwA program can expect upon completion of a master’s degree program
- Represent minimum capabilities expected of a professional in the area of software assurance upon completion of the program
- provide a model for curriculum content, organization, and expected curriculum outcomes
- Support those who assess software assurance programs

Outcomes: APM -1

Assurance Process and Management (APM)

Assurance Across Life Cycles

- ability to incorporate assurance technologies and methods into life-cycle processes and development models for new or evolutionary system development, and for system or service acquisition

Risk Management

- ability to perform risk analysis, trade-off assessment, and prioritization of security measures

Outcomes: APM -2

Assurance Process and Management (APM)

Assurance Assessment

- ability to analyze effectiveness of assurance operations and create auditable evidence of security measures

System Operational Assurance

- ability to make a business case for software assurance, lead assurance efforts, understand standards, comply with regulations, plan for business continuity, and keep current in security technologies

Outcomes: APT

Assurance Product and Technology (APT)

System Security Assurance

- ability to incorporate effective security technologies and methods into new and existing systems

System Functionality Assurance

- ability to verify new and existing system functionality for conformance to requirements and absence of malicious content

System Operational Assurance

- ability to monitor and assess system operational security and respond to new threats

Prerequisites

Prerequisites for Entering Students

[MSwA 2010] defines prerequisites foundations from a security perspective, with areas such as reliability and safety for other analyses

Undergraduate degree disciplines of entrants

- computer science, software engineering, electrical and computer engineering, math, or information systems

Prerequisites satisfied through combination of

- undergraduate courses, work experience, and possibly remedial education

Organized into 3 categories

- computing foundations, software engineering, security engineering

Prereq: Computing Foundations -1

Discrete Mathematics

- sets, functions, and relations; graphs and trees; propositional and predicate logic; number systems; modular arithmetic; proof techniques
- fundamentals of probability and statistics

Computer Fundamentals

- computer hardware function and organization; assembly and microcode organization; memory organization and access; communication interfaces; multiprocessing

Prereq: Computing Foundations -2

Networks and Communications

- network-centric computing and communication; network topologies and protocols; addressing and routing
- web applications and multimedia; wireless and mobile computing
- network configuration, monitoring, performance, and management

Prereq: Computing Foundations -3

Programming Environments

- operating systems, including scheduling, memory management, concurrency, security, file system and device management
- real-time and embedded system concepts
- implementation environments, including programming languages, compilers, assemblers, loaders, and libraries
- support tools for analysis of programs and systems

Prereq: Computing Foundations -4

Program Development

- object-oriented programming using a language such as Java or C++
- input/output streams and process threads; pointers; memory allocation and deallocation
- fundamental data structures including arrays, lists, queues, and stacks
- analysis and implementation of basic algorithms
- semantic foundations of programming languages

Prereq: Software Engineering -1

Software Development Life Cycle

- requirements elicitation, analysis, and validation; use case and flow analysis; system and software specification; design and implementation; verification, inspection, and testing; maintenance and evolution
- software quality, dependability, and reliability
- project management concepts, including work breakdown, scheduling, budgeting, tracking, and risk management
- system development models, including incremental, spiral, and agile methods; project management frameworks including the CMMI®

Prereq: Software Engineering -2

Software Analysis

- software analysis tools and methods, including static and dynamic analyzers
- reverse engineering methods for analysis of program behavior

Prereq: Security Engineering -1

Security Issues

- knowledge of security threats from criminal, nation-state, and insider adversaries; consequences of attacks on infrastructure, defense, and economic systems; security risks and requirements for application domains such as finance, energy, and transportation
- security issues in computing trends, including global networks, systems-of-systems, open-source, cloud computing, and social networking
- security properties, including privacy, confidentiality, authentication, authorization, availability, integrity, and non-repudiation
- security aspects of human behavior in interacting with systems

Prereq: Security Engineering -2

Security Analysis

- intruder motivations and methods; network, system, and software security vulnerabilities
- traditional security practices, including firewalls, intrusion detection systems, and network monitoring systems, and their limitations
- methods for achieving traditional security properties; encryption and decryption; secure coding techniques
  - Note: security testing is an important part of prerequisite knowledge, but current curriculums don’t cover this topic

Curriculum Architecture

MSwA Curriculum Architecture

Is the minimum content degree that programs should include.

Is compatible with software engineering master’s programs that are based GSwE2009 curriculum.

Is intended to provide a structural basis for programs that deliver the outcomes described earlier.

Includes

- preparatory material
- core materials
- elective materials
- capstone experience

Architectural Structure of an MSwA2010 Degree Program

Preparatory Materials	Computing Foundations Software Engineering Security Engineering
MSwA Core	Assurance Across Life Cycles Risk Management Assurance Assessment Assurance Management System Security Assurance Assured Software Analytics System Operational Assurance
Electives	Courses Related to Assurance in Selected Domains
Capstone Experience	Project

MSwE with SwA Specialization

Preparatory Materials	Computing Foundations Software Engineering Security Engineering
GSwE Core	Ethics and Professional Conduct Systems Engineering Requirements Engineering Software Design Software Construction Software Testing Software Maintenance Configuration Management Software Engineering Management Software Engineering Processes Software Quality
MSwA Core	Assurance Across Life Cycles Risk Management Assurance Assessment Assurance Management System Security Assurance Assured Software Analytics System Operational Assurance
Capstone Experience	Project

Core Body of Knowledge

Core Body of Knowledge (BoK):APM -1

Assurance Process and Management (APM)

1. Assurance Across Life Cycles

- Software Life-Cycle Processes
  - New development: processes associated with the full development of a software product
  - Integration, assembly, and deployment: processes concerned with the final phases of the development of a new or modified software product
  - Operation and evolution: processes that guide the operation of the software product and its change over time
  - Acquisition, supply, and service: processes that support acquisition, supply, or service of a software product

Core BoK: APM -2

Assurance Process and Management (APM)

1. Assurance Across Life Cycles (continued)

- Software Assurance Processes and Practices
  - Process and practice assessment: methods, procedures, and tools used to assess assurance processes and practices
  - Software assurance integration into SDLC phases: integration of assurance practices into typical life-cycle phases (e.g., requirements engineering, architecture and design, coding, test, evolution, and acquisition)

Core BoK: APM -3

Assurance Process and Management (APM)

2. Risk Management

- Risk Management Concepts
  - Types and classification
  - Probability, impact, severity
  - Models, processes, metrics
- Risk Management Process
  - Identification
  - Analysis
  - Planning
  - Monitoring and management

Core BoK: APM -4

Assurance Process and Management (APM)

2. Risk Management (continued)

- Software Assurance Risk Management
  - Vulnerability and threat identification
  - Analysis of software assurance risks
  - Software assurance risk mitigation
  - Assessment of software assurance processes and practices

Core BoK: APM -5

Assurance Process and Management (APM)

3. Assurance Assessment

- Assurance Assessment Concepts
  - Baseline level of assurance
  - Assessment methods
- Measurement for Assessing Assurance
  - Product and process measures by life-cycle
  - Other performance indicators that test for the baseline, by life-cycle phase
  - Measurement processes and frameworks
  - Business survivability and operational continuity

Core BoK: APM -6

Assurance Process and Management (APM)

3. Assurance Assessment (continued)

- Assurance Assessment Processes
  - Comparison of measurements to the established baseline
  - Identification and response to variances

Core BoK: APM -7

Assurance Process and Management (APM)

4. Assurance Management

- Making the Business Case for Assurance
  - Valuation and cost/benefit models, cost and loss avoidance, return on investment
  - Risk analysis
  - Compliance justification
  - Business impact/needs analysis

Core BoK: APM -8

Assurance Process and Management (APM)

4. Assurance Management (continued)

- Managing Assurance
  - Project management across the life cycle
  - Integration of other knowledge units
- Compliance Considerations for Assurance
  - Laws and regulations
  - Standards
  - Policies

Core BoK: APT -1

Assurance Product and Technology (APT)

5. System Security Assurance

- For Newly Developed, Acquired, and Existing Software and Services
  - Security aspects of software and services
  - Attack objectives, strategies, and tactics
  - Analysis of threats to networks, systems, and applications
  - Security requirements and properties
  - Security engineering methods and technologies
- Ethics and Integrity in Security Engineering
  - Ethics and legal constraints
  - Computer attack case studies
-

Core BoK: APT -2

Assurance Product and Technology (APT)

6. System Functionality Assurance

- Assurance Technology
  - Technology evaluation
  - Technology improvement
- Assured Software Development
  - Development methods
  - Quality attributes
  - Maintenance methods

Core BoK: APT -3

Assurance Product and Technology (APT)

6. System Functionality Assurance (continued)

- Software Assurance Analytics
  - Systems analysis
  - Reverse engineering technologies
  - Structural and functional analysis
  - Analysis of analysis methods and tools
  - Testing for assurance
  - Assurance evidence
- Assurance in Acquisition
  - Assurance of acquired software
  - Assurance of software services

Core BoK: APT -4

Assurance Product and Technology (APT)

7. System Operational Assurance

- Operational Procedures
  - Business objectives
  - Assurance procedures
  - Assurance training
- Operational Monitoring
  - Monitoring technology
  - Operational evaluation
  - Operational maintenance
  - Malware analysis

Core BoK: APT -5

Assurance Product and Technology (APT)

7. System Operational Assurance (continued)

- System Control and Recovery
  - Responses to adverse events
  - Business and system survivability

Next Steps

Next Steps and Dissemination

MswA 2010 Report: first of a set of activities needed to support Master of Software Assurance degree programs and tracks

To consider it a success, the curriculum model must be

- available
- understood by the targeted academic & industrial communities
- viewed as a key reference for SwA curriculum development, and
- actually used to develop and modify software assurance focused curricula

MSwA 2010 Curriculum Plans

2010:

- Release MSwA Reference Curriculum [MSwA 2010]
- Identify suitable software assurance undergraduate courses
- Identify sample MSwA curricula and course descriptions

2010 – 2011:

- Identify transition opportunities for the curriculum (e.g. papers, seminars, workshops, faculty development)
- Work with professional societies towards recognition of the curriculum

Curriculum Resources

Build Security In Website

Build Security In web site: https://buildsecurityin.us-cert.gov/

- For use by software developers and software development organizations
- Build Security In contains or links to information about best practices, tools, guidelines, rules, principles, resources, etc. to help organizations build secure and reliable software.

Contributing authors include Carnegie Mellon University Software Engineering Institute’s CERT Program, Cigital, and experts from the Software Assurance community

Expanding to include current doctoral research

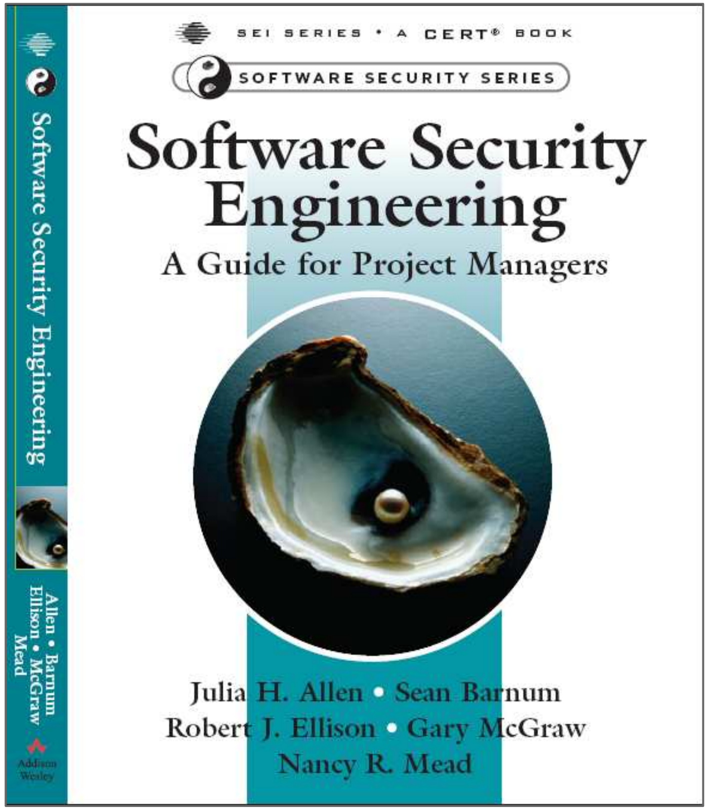
Sponsored by U.S. Department of Homeland Security, Software Assurance Program

Software Security Engineering: A Guide for Project Managers

Published May 2008

Contains an introduction to software security engineering and guidance for project managers

- Derives material from DHS SwA “Build Security In” web site
- Provides a process focus for projects delivering software-intensive products and systems



SEI SERIES • A CERT® BOOK

SOFTWARE SECURITY SERIES

Software Security Engineering

A Guide for Project Managers

Julia H. Allen • Sean Barnum

Robert J. Ellison • Gary McGraw

Nancy R. Mead

Acronyms/Abbreviations

Glossary

References

Acronyms/Abbreviations -1

ACM	Association for Computing Machinery
ASEE	American Society for Engineering Education
AN	Analysis *[Bloom cognitive level]*
AP	Application *[Bloom cognitive level]*
BoK	Body of Knowledge
BSI	Build Security In
C	Comprehension *[Bloom cognitive level]*
COTS	Commercial Off The Shelf
CMMI®	Capability Maturity Model Integration

Acronyms/Abbreviations -2

DHS	(U.S.) Department of Homeland Security
DOD	(U.S.) Department of Defense
IEEE	Institute of Electrical and Electronics Engineers
IEEE-CS	IEEE Computer Society
K	Knowledge *[Bloom cognitive level]*
MSE	Master of Software Engineering
MSwA	Master of Software Assurance
NCSD	National Cyber Security Division

Acronyms/Abbreviations -3

SDLC	Software Development Life Cycle
SEI	Software Engineering Institute
SwA	Software Assurance
SwACBK	Software Assurance Curriculum Body of Knowledge

Glossary -1 [Partial]

Acquisition Process of obtaining a system, software product, or software service. Software products may include commercial-off-the-shelf (COTS) products, modified-off-the-shelf products, open source products, or fully developed products.

Correct functionality Software assurance seeks to provide a level of confidence that software functions in the intended manner as defined by requirements and specifications. Software should establish a secure computing environment and provide required functionality that is free from errors and known vulnerabilities. Software evolution should maintain these properties.

Glossary -2 [Partial]

Software analytics (1) Include reverse engineering technologies to transform arbitrary control logic into structured form and function abstraction to recover designs and specifications from implementations.

Software analytics (2) Specialized technologies and processes are necessary to analyze and assure functional and security properties of software. Analysis subject matter extends across the life cycle, and includes specification, design, code, inspection, and test artifacts. Analytic methods include reverse engineering to transform arbitrary control logic into structured form for improved understanding and function abstraction to recover designs and specifications from implementations.

References -1

[ACM 2008] ACM & IEEE-CS. “Computer Science Curriculum 2008: An Interim Revision of CS 2001.” *Computing Curriculum Series*. http://www.computer.org/portal/cms_docs_ieeecs/ieeecs/education/cc2001/ComputerScience2008.pdf (2008).

[Allen 2008] Allen, Julia H.; Barnum, Sean; Ellison, Robert J.; McGraw, Gary; & Mead, Nancy R. *Software Security Engineering: A Guide for Project Managers*. Addison-Wesley Professional, 2008.

References -2

[Bloom 1956] Bloom, B. S., ed. *Taxonomy of educational objectives: The classification of Educational goals: Handbook I, Cognitive Domain*. Longmans, 1956.

[CNSS 2009] Committee on National Security Systems. *Instruction No. 4009, National Information Assurance Glossary*. Revised June 2009.

[DHS 2010a] DHS SwA. *Build Security In*. https://buildsecurityin.us-cert.gov/daisy/adm-bsi/home.html (2010).

References -3

[DHS 2010b] Department of Homeland Security (DHS) Software Assurance (SwA)Workforce Education and Training Working Group. *Software Assurance CBK/Principles Organization.* https://buildsecurityin.us-cert.gov/bsi/dhs/927-BSI.html (2010).

[IEEE-CS 2004a] IEEE-CS & ACM. "Computer Engineering 2004: Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering." Computing Curriculum Series.

http://www.acm.org/education/education/curric_vols/CE-Final-Report.pdf (2004).

References -4

[IEEE-CS 2004b] IEEE-CS & ACM. “Software Engineering 2004: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering*.*” *Computing Curriculum Series*. http://www.computer.org/portal/cms_docs_ieeecs/ieeecs/education/cc2001/SE2004Volume.pdf (2004).

[iSSEc 2009] Integrated Software & Systems Engineering Curriculum (iSSEc) Project. *Graduate Software Engineering 2009 [GSwE2009] Curriculum Guidelines for Graduate Degree Programs in Software Engineering, Version 1.0*. Stevens Institute of Technology, 2009.

References -5

[Lethbridge 2006] Lethbridge, Timothy C., LeBlanc, Richard J., Jr., Kelley Sobel, Ann E., Hilburn, Thomas B., Diaz-Herrera, Jorge L. “SE2004: Recommendations for Undergraduate Software Engineering Curricula.” *IEEE Software 23*, 6 (Nov./Dec. 2006): 19-25.

[Mead 2010] Mead, Nancy R. & Ingalsbe, Jeff. “How to Get Started in Software Assurance Education.” Tutorial presented at the 23rd Annual IEEE-CS Conference on Software Engineering Education and Training (CSEET). Pittsburgh, PA, March 2010.

References -6

[MSwA 2010] Mead, Nancy R.; Allen, Julia H.; Ardis, Mark; Hilburn, Thomas B.; Kornecki, Andrew J.; Linger, Rick; & McDonald, James. *Master of Software Assurance Reference Curriculum* (CMU/SEI-2010-TR-005, ESC-TR-2010-005). Software Engineering Institute, Carnegie Mellon University, 2010.

[OPM 2010] U.S. Office of Personnel Management. *Federal Cyber Service: Scholarship for Service*. https://www.sfs.opm.gov/ (2010).