Software & Supply Chain Assurance:

A Historical Perspective of Community Collaboration

Joe Jarzombek, PMP, CSSLP
Director for Software & Supply Chain Assurance
Stakeholder Engagement & Cyber Infrastructure Resilience
Cyber Security & Communications

Enabling Enterprise Resilience through Security Automation, Software Assurance, and Supply Chain Risk Management
DOD SOFTWARE ASSURANCE INITIATIVE: Mitigating Risks Attributable to Software

Countering Threats that Target Software in Systems and Networks

Workshop Out-Briefs

Sep 1, 2004

The first public meeting
Action Required to Address Material Weakness in Software

SOFTWARE ASSURANCE INITIATIVE PROVIDES:

- Proactive transformation of processes to mitigate risks attributable to software
- Sustainable capability to counter threats to software-enabled DoD systems and networks

Increasing Dependence on Software
Increase in Threats Targeting Software
Increasing Ease of Exploitation
Increase in Software Vulnerabilities

Dramatic Increase in Mission Risk
SW Assurance is managed as part of: the DoD Information Assurance (IA) Strategy and the DHS National Cyber Security Strategy

- WG1 - Security Process Capability (improvement & evaluation),
- WG2 - Software Product Evaluation (product focused),
- WG3 - Counter Intelligence (CI) Threat Assessment Support
- WG4 - Acquisition/Procurement and Industrial Security, and
- WG5 - User Identification & Prioritization of Protected Assets
- WG6 - Workforce Education and Training
PITAC* Findings Relative to Needs for Secure Software Engineering & Software Assurance

- Commercial software engineering today lacks the scientific underpinnings and rigorous controls needed to produce high-quality, secure products at acceptable cost.

- Commonly used software engineering practices permit dangerous errors, such as improper handling of buffer overflows, which enable hundreds of attack programs to compromise millions of computers every year.

- In the future, the Nation may face even more challenging problems as adversaries – both foreign and domestic – become increasingly sophisticated in their ability to insert malicious code into critical software.

- Recommendations for increasing investment in cyber security provided to NITRD Interagency Working Group for Cyber Security & Information Assurance R&D

* President’s Information Technology Advisory Committee (PITAC) Report to the President, “Cyber Security: A Crisis of Prioritization,” February 2005 identified top 10 areas in need of increased support, including: ‘secure software engineering and software assurance’ and ‘metrics, benchmarks, and best practices’ [Note: PITAC is now a part of PCAST]
Software Assurance Addresses Exploitable Software: Outcomes of non-secure practices and/or malicious intent

Exploitation potential of vulnerability is independent of “intent”

High quality’ can reduce security flaws attributable to defects; yet traditional S/W quality assurance does not address intentional malicious behavior in software

*Intentional vulnerabilities: spyware & malicious logic deliberately imbedded (might not be considered defects)

Software Assurance (SwA) is the level of confidence that software functions as intended and is free of vulnerabilities, either intentionally or unintentionally designed or inserted as part of the software throughout the life cycle.*

From CNSS Instruction 4009 “National Information Assurance Glossary” (26APR2010)
DHS Software Assurance Program Overview

Program established in response to the National Strategy to Secure Cyberspace - Action/Recommendation 2-14:

“DHS will facilitate a national public-private effort to promulgate best practices and methodologies that promote integrity, security, and reliability in software code development, including processes and procedures that diminish the possibilities of erroneous code, malicious code, or trap doors that could be introduced during development.”

DHS Program goals promote the security and resilience of software across the development, acquisition, and operational life cycle.

DHS Software Assurance (SwA) program is scoped to address:

- **Trustworthiness** - No exploitable vulnerabilities or malicious logic exist in the software, either intentionally or unintentionally inserted,
- ** Dependability (Correct and Predictable Execution)** - Justifiable confidence that software, when executed, functions as intended,
- **Survivability** - If compromised, damage to the software will be minimized, and it will recover quickly to an acceptable level of operating capacity;
- **Conformance** – Planned, systematic set of multi-disciplinary activities that ensure processes/products conform to requirements, standards/procedures.

See Wikipedia.org for “Software Assurance” - CNSS Instruction No. 4009, "National Information Assurance Glossary," Revised 2006, defines Software Assurance as: "the level of confidence that software is free from vulnerabilities, either intentionally designed into the software or accidentally inserted at anytime during its lifecycle, and that the software functions in the intended manner".
In Education and Training, Software Assurance could be addressed as:

• A “knowledge area” extension within each of the contributing disciplines;
• A stand-alone CBK drawing upon contributing disciplines;
• A set of functional roles, drawing upon a common body of knowledge; allowing more in-depth coverage dependent upon the specific roles.

Intent is to provide framework for curriculum development and evolution of contributing BOKs.

* See ‘Notes Page’ view for contributing BOK URLs and relevant links
Assurance relative to Trust

Managing Effects of Unintentional Defects in Component or System Integrity

Managing Consequences of Unintentional Defects

Quality

Safety

Security

TRUST

Managing Consequences of Attempted/Intentional Actions Targeting Exploitable Constructs, Processes & Behaviors
“Supply chain introduces risks to American society that relies on Federal Government for essential information and services.”

30 Sep 2005 changes to Federal Acquisition Regulation (FAR) focus on IT Security

Focuses on the role of contractors in security as Federal agencies outsource various IT functions.

Risk Management (Enterprise <=> Project): Shared Processes & Practices // Different Focuses

- **Enterprise-Level:**
  - Regulatory compliance
  - Changing threat environment
  - Business Case

- **Program/Project-Level:**
  - Cost
  - Schedule
  - Performance

Software Supply Chain Risk Management traverses enterprise and program/project interests
Software Assurance “End State” Objectives…

► Government, in collaboration with industry / academia, raised expectations for product assurance with requisite levels of integrity and security:
  ▪ Helped advance more comprehensive software assurance diagnostic capabilities to mitigate risks stemming from exploitable vulnerabilities and weaknesses;
  ▪ Collaboratively advanced use of software security measurement & benchmarking schemes
  ▪ Promoted use of methodologies and tools that enabled security to be part of normal business.

► Acquisition managers & users factored risks posed by the software supply chain as part of the trade-space in risk mitigation efforts:
  ▪ Information on suppliers’ process capabilities (business practices) would be used to determine security risks posed by the suppliers’ products and services to the acquisition project and to the operations enabled by the software.
  ▪ Information about evaluated products would be available, along with responsive provisions for discovering exploitable vulnerabilities, and products would be securely configured in use.

► Suppliers delivered quality products with requisite integrity and made assurance claims about the IT/software safety, security and dependability:
  ▪ Relevant standards would be used from which to base business practices & make claims;
  ▪ Qualified tools used in software lifecycle enabled developers/testers to mitigate security risks;
  ▪ Standards and qualified tools would be used to certify software by independent third parties;
  ▪ IT/software workforce had requisite knowledge/skills for developing secure, quality products.

...Enabling Software Supply Chain Transparency
DHS NCSD Software Assurance (SwA) Program

Through public-private collaboration promotes security and resilience of software throughout the lifecycle; focused on reducing exploitable software weaknesses and addressing means to improve capabilities that routinely develop, acquire, and deploy resilient software products. Collaboratively advancing software-relevant rating schemes

• Serves as a focal point for interagency public-private collaboration to enhance development and acquisition processes and capability benchmarking to address software security needs.
  – Hosts interagency Software Assurance Forums, Working Groups and training to provide public-private collaboration in advancing software security and providing publicly available resources.
  – Provides collaboratively developed, peer-reviewed information resources on Software Assurance, via journals, guides & on-line resources suitable for use in education, training, and process improvement.
  – Provides input and criteria for leveraging international standards and maturity models used for process improvement and capability benchmarking of software suppliers and acquisition organizations.

• Enables software security automation and measurement capabilities through use of common indexing and reporting capabilities for malware, exploitable software weaknesses, and common attacks which target software.
  – Collaborates with the National Institute of Standards and Technology, international standards organizations, and tool vendors to create standards, metrics and certification mechanisms from which tools can be qualified for software security verification.
  – Manages programs to facilitate the adoption of Malware Attribute Enumeration Classification (MAEC), Common Weakness Enumeration (CWE), and Common Attack Pattern Enumeration and Classification (CAPEC).
Challenges in Mitigating Risks Attributable to Exploitable ICT/Software & Supply Chains

Several needs arise:

- Need internationally recognized standards to support security automation and processes to provide transparency for informed decision-making in mitigating enterprise risks.

- Need comprehensive diagnostic capabilities to provide sufficient evidence that “code behavior” can be understood to not possess exploitable or malicious constructs.

- Need ‘Assurance’ to be explicitly addressed in standards & capability benchmarking models for organizations involved with security/safety-critical applications.

- Need rating schemes for ICT/software products and supplier capabilities.
Mitigating Risks Attributable to Exploitable Software and Supply Chains

Enterprises seek comprehensive capabilities to:

- Avoid accepting software with **MALWARE** pre-installed.  
- Determine that no publicly reported **VULNERABILITIES** remain in code prior to operational acceptance, and that future discoveries of common vulnerabilities and exposures can be quickly patched.
- Determine that exploitable software **WEAKNESSES** that put the users most at risk are mitigated prior to operational acceptance or after put into use.
As part of the DHS risk mitigation effort, the SwA Program seeks to reduce software vulnerabilities, minimize exploitation, and address ways to improve the routine development of trustworthy software products and tools to analyze systems for hidden vulnerabilities.

The SwA framework encourages the production, evaluation and acquisition of better quality and more secure software; leverages resources to target the following four areas:

- **People** – education and training for developers and users
- **Processes** – sound practices, standards, and practical guidelines for the development of secure software
- **Technology** – diagnostic tools, cyber security R&D and measurement
- **Acquisition** – due-diligence questionnaires, contract templates and guidelines for acquisition management and outsourcing

* July 28, 2006 statement of George Foresman, DHS UnderSecretary for Preparedness, before the U.S. Senate Committee on Homeland Security and Governmental Affairs, Subcommittee on Federal Financial Management, Government Information, and International Security
Software Assurance Forum & Working Groups*

... encourage the production, evaluation and acquisition of better quality and more secure software through targeting

<table>
<thead>
<tr>
<th>People</th>
<th>Processes</th>
<th>Technology</th>
<th>Acquisition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Developers and users education &amp; training</td>
<td>Sound practices, standards, &amp; practical guidelines for secure software development</td>
<td>Security test criteria, diagnostic tools, common enumerations, SwA R&amp;D, and SwA measurement</td>
<td>Software security improvements through due-diligence questions, specs and guidelines for acquisitions/outsourcing</td>
</tr>
</tbody>
</table>

**Products and Contributions**

- Build Security In - https://buildsecurityin.us-cert.gov and SwA community resources & info clearinghouse
- SwA Common Body of Knowledge (CBK) & Glossary
- Organization of SwSys Security Principles/Guidelines
- SwA Developers' Guide on Security-Enhancing SDLC
- Software Security Assurance State of the Art Report
- Systems Assurance Guide (via DoD and NDIA)
- Practical Measurement Framework for SwA/InfoSec
- Making the Business Case for Software Assurance
- SwA Metrics & Tool Evaluation (with NIST)
- SwA Ecosystem w/ DoD, NSA, NIST, OMG & TOG
- NIST Special Pub 500 Series on SwA Tools
- Common Weakness Enumeration (CWE) dictionary
- Common Attack Pattern Enumeration (CAPEC)
- SwA in Acquisition: Mitigating Risks to Enterprise
- Software Project Management for SwA SOAR

*SwA Forum is part of Cross-Sector Cyber Security Working Group (CSCSWG) established under auspices of the Critical Infrastructure Partnership Advisory Council (CIPAC) that provides legal framework for participation.
BSI https://buildsecurityin.us-cert.gov focuses on making Software Security a normal part of Software Engineering

SwA Community Resources and Information Clearinghouse (CRIC)

https://buildsecurityin.us-cert.gov/swa/ focuses on all contributing disciplines, practices and methodologies that advance risk mitigation efforts to enable greater resilience of software/cyber assets.

The SwA CRIC provides a primary resource for SwA Working Groups.

Where applicable, SwA CRIC & BSI provide relevant links to each other.
Process Agnostic Lifecycle

**Touch Points & Artifacts**

**Architecture & Design**
- Architectural risk analysis
- Threat modeling
- Principles
- Guidelines
- Historical risks
- Modeling tools
- Resources

**Code**
- Code analysis
- Assembly, integration & evolution
- Coding practices
- Coding rules
- Code analysis
- Resources

**Test**
- Security testing
- White box testing
- Attack patterns
- Historical risks
- Resources

**Requirements**
- Requirements engineering
- Attack patterns
- Resources

**System**
- Penetration testing
- Incident management
- Deployment & operations
- Black box testing
- Resources

**Fundamentals**
- Risk management
- Project management
- Training & awareness
- Measurement
- SDLC process
- Business relevance
- Resources

[https://buildsecurityin.us-cert.gov](https://buildsecurityin.us-cert.gov)

**Key**
- Best (sound) practices
- Foundational knowledge
- Tools
- Resources

Since 3 Oct 2005
Software assurance (SwA) 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 (from CNSS 4009 IA Glossary - see Wikipedia for definitions and descriptions).

As part of DHS risk mitigation efforts to enable greater resilience of cyber assets, the Software Assurance Program seeks to reduce software vulnerabilities, minimize exploitation, and address ways to routinely acquire, develop and deploy reliable and trustworthy software products with predictable execution, and to improve diagnostic capabilities to analyze systems for exploitable weaknesses.

The Software Assurance Forum and several working groups, composed of stakeholders in government, industry, and academia, are contributing to efforts focused on advancing software assurance objectives. The next Software Assurance Forum is in November 2009. Registration information is available on the Forums page.

Focused efforts for advancing software assurance are addressed in the working groups listed below. Click on any working group’s name to see Recent Releases and Updates, current activities, and other information for that working group.

- Workforce Education & Training
- Processes & Practices
- Technology, Tools & Product Evaluation
- Acquisition & Outsourcing
- Measurement
- Business Case
- Malware Attribution

WHY IS SOFTWARE ASSURANCE CRITICAL?

The nation’s critical infrastructure (energy, transportation, telecommunications, etc.), businesses, and services are extensively and increasingly controlled and enabled by software. Vulnerabilities in that software put those resources at risk. The risk is increased by the large number of security vulnerabilities found in a large portion of software and the ease with which they can be exploited.

See https://buildsecurityin.us-cert.gov/swa/ for information
Security-Enhanced Capabilities: Mitigating Risks to the Enterprise

With today’s global software supply chain, Software Engineering, Quality Assurance, Testing and Project Management must explicitly address security risks posed by exploitable software.

- Traditional processes do not explicitly address software-related security risks that can be passed from projects to using organizations.

Mitigating Supply Chain Risks requires an understanding and management of Suppliers’ Capabilities, Products and Services

- Enterprise risks stemming from supply chain are influenced by suppliers and acquisition projects (including procurement, SwEng, QA, & testing).
- IT/Software Assurance processes/practices span development/acquisition.
- Derived (non-explicit) security requirements should be elicited/considered.

More comprehensive diagnostic capabilities and standards are needed to support processes and provide transparency for more informed decision-making for mitigating risks to the enterprise.

Free resources are available to assist personnel in security-enhancing contracting, outsourcing and development activities (see https://buildsecurityin.us-cert.gov)
Need for Rating Schemes

- Rating of Software products:
  - Supported by automation
  - Standards-based
  - Rules for aggregation and scaling
  - Verifiable by independent third parties
  - Labeling to support various needs (e.g., security, dependability, etc)
  - Meaningful and economical for consumers and suppliers

- Rating of Suppliers providing software products and services
  - Standards-based or model-based frameworks to support process improvement and enable benchmarking of organizational capabilities
  - Credential programs for professionals involved in software lifecycle activities and decisions


Both collaboratively developed through the Software Assurance Working Group on Workforce Education and Training

0. INTRODUCTION
0.1/0.2 PURPOSE / SCOPE
0.3 REASONING UNDERLYING THE ORGANIZATION
0.4 ORGANIZATION OF REMAINDER OF DOCUMENT

1. THE ADVERSE
1.1. LIMIT, REDUCE, OR MANAGE VIOLATORS
1.2. LIMIT, REDUCE, OR MANAGE BENEFITS TO VIOLATORS OR ATTACKERS
1.3. INCREASE ATTACKER LOSSES
1.4. INCREASE ATTACKER UNCERTAINTY

2. THE SYSTEM
2.1. LIMIT, REDUCE, OR MANAGE VIOLATIONS
2.2. IMPROVE BENEFITS OR AVOID ADVERSE EFFECTS ON SYSTEM BENEFITS
2.3. LIMIT, REDUCE, OR MANAGE SECURITY-RELATED COSTS
2.4. LIMIT, REDUCE, OR MANAGE SECURITY-RELATED UNCERTAINTIES

3. THE ENVIRONMENT
3.1. NATURE OF ENVIRONMENT
3.2. BENEFITS TO AND FROM ENVIRONMENT
3.3. LIMIT, REDUCE, OR MANAGE ENVIRONMENT-RELATED LOSSES
3.4. LIMIT, REDUCE, OR MANAGE ENVIRONMENT-RELATED UNCERTAINTIES

4. CONCLUSION
5. APPENDIX A: PRINCIPLES OF WAR
6. APPENDIX B: PURPOSE-CONDITION-ACTION-RESULT MATRIX
7/8. BIBLIOGRAPHY / ACKNOWLEDGEMENTS
BSI https://buildsecurityin.us-cert.gov focuses on making Software Security a normal part of Software Engineering.

SwA Community Resources and Information Clearinghouse (CRIC)

https://buildsecurityin.us-cert.gov/swa/ focuses on all contributing disciplines, practices and methodologies that advance risk mitigation efforts to enable greater resilience of software/cyber assets.

The SwA CRIC provides a primary resource for SwA Working Groups. Where applicable, SwA CRIC & BSI provide relevant links to each other.

Organized for Project Managers

- Derives material from DHS SwA “Build Security In” web site
  - https://buildsecurityin.us-cert.gov
- Provides a process focus for projects delivering software-intensive products and systems

Published in May 2008
July 2007 FREE publicly available resource provides a comprehensive look at efforts to improve the state of Software Security Assurance:

- describes the threats and common vulnerabilities to which software is subject;
- presents the many ways in which the S/W Security Assurance problem is being framed and understood across government, industry, and academia;
- describes numerous methodologies, best practices, technologies, and tools currently being used to specify, design, and implement software that will be less vulnerable to attack, and to verify its attack-resistance, attack-tolerance, and attack-resilience;
- offers a large number of available resources from which to learn more about principles and practices that constitute Software Security Assurance;
- provides observations about potentials for success, remaining shortcomings, and emerging trends across the S/W Security Assurance landscape.


*The SOAR reflects output of efforts in the DoD-DHS Software Assurance Forum and Working Groups that provide collaborative venues for stakeholders to share and advance techniques and technologies relevant to software security.*
• Describes how to integrate security principles and practices in software development life cycle
• Addresses security requirements, secure design principles, secure coding, risk-based software security testing, and secure sustainment
• Provides guidance for selecting secure development methodologies, practices, and technologies
  – Collaboratively developed/updated via SwA Forum working groups
  – Released Oct 2008 by DACS
  – Free, available for download via DACS & DHS SwA Community Resources & Information Clearinghouse

https://www.thedacs.com/techs/enhanced_life_cycles/
Common security-related elements of software development methodologies
- Security requirements help drive design, code handling, programming, and testing activities

Secure Programming practices:
- Minimize unsafe function use
- Use the latest compiler toolset
- Use static and dynamic analysis tools
- Use manual code review on high-risk code
- Validate input and output
- Use anti-cross site scripting libraries
- Use canonical data formats
- Avoid string concatenation for dynamic SQL
- Eliminate weak cryptography
- Use logging and tracing

Test to validate robustness and security
- Fuzz testing
- Penetration testing & third party assessment
- Automated test tools (in all development stages)

Code Integrity and Handling
- Least privilege access, Separation of duties,
- Persistent protection, Compliance management; Chain of custody & supply chain integrity.

Documentation (about software security posture & secure configurations)
“Software Assurance in Acquisition: Mitigating Risks to the Enterprise“
Version 1.0, Oct 2008, available for community use
Executive Summary

1. Introduction
   1.1 Background
   1.2 Purpose and Scope
   1.3 Audience—Acquisition Official Defined
   1.4 Document Structure
   1.5 Risk-Managed Software Acquisition Process

2. Planning Phase
   2.1 Needs Determination, Risk Categorization, & Solution Alternatives
   2.2 SwA Requirements
   2.3 Acquisition Plan and/or Acquisition Strategy
   2.4 Evaluation Plan and Criteria
   2.5 SwA Due Diligence Questionnaires

3. Contracting Phase
   3.1 Request for Proposals
      3.1.1 Work Statement
      3.1.2 Terms and Conditions
      3.1.3 Instructions to Suppliers
      3.1.4 Certifications
      3.1.5 Prequalification
   3.2 Proposal Evaluation
   3.3 Contract Negotiation
   3.4 Contract Award

4. Implementation and Acceptance Phase
   4.1 Contract Work Schedule
   4.2 Change Control
   4.3 Risk Management Plan
   4.4 Assurance Case Management
   4.5 Independent Software Testing
   4.6 Software Acceptance

5. Follow-on Phase
   5.1 Support and Maintenance
      5.1.1 Risk Management
      5.1.2 Assurance Case Management—Transition to Ops
      5.1.3 Other Change Management Considerations
   5.2 Disposal or Decommissioning

Appendix A/B—Acronyms/Glossary

Appendix C—An Imperative for SwA in Acquisition

Appendix D—Software Due Diligence Questionnaires
   Table D-1. COTS Proprietary Software Questionnaire
   Table D-2. COTS Open-Source Software Questionnaire
   Table D-3. Custom Software Questionnaire
   Table D-4. GOTS Software Questionnaire
   Table D-5. Software Services

Appendix E—Other Examples of Due Diligence Questionnaires

Appendix F—Sample Language for the RFP and/or Contract
   F.1 Security Controls and Standards
   F.2 Securely Configuring Commercial Software
   F.3 Acceptance Criteria
   F.4 Certifications
   F.5 Sample Instructions to Offerors Sections
   F.6 Sample Work Statement Sections
   F.7 Open Web Application Security Project
   F.8 Certification of Originality

Appendix H—References
Software Assurance (SwA) Pocket Guide Series

SwA in Acquisition & Outsourcing
• Software Assurance in Acquisition and Contract Language
• Software Supply Chain Risk Management and Due-Diligence

SwA in Development *
• Risk-based Software Security Testing
• Requirements and Analysis for Secure Software
• Architecture and Design Considerations for Secure Software
• Secure Coding and Software Construction
• Key Practices for Mitigating the Most Egregious Exploitable Software Weaknesses

* All include questions to ask developers

SwA Life Cycle Support
• SwA in Education, Training and Certification

SwA Pocket Guides and SwA-related documents are collaboratively developed with peer review; they are subject to update and are freely available for download via the DHS Software Assurance Community Resources and Information Clearinghouse at https://buildsecurityin.us-cert.gov/swa (see SwA Resources)
Organizations that provide security engineering & risk-based analysis throughout the lifecycle will have more resilient software products / systems.

“Build Security In” throughout the lifecycle

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Security</td>
<td></td>
<td></td>
<td>Risk-based Test Plans</td>
<td></td>
<td>Risk Analysis</td>
<td>Security Ops &amp; Vulnerability Mgt</td>
</tr>
<tr>
<td>Requirements</td>
<td></td>
<td></td>
<td></td>
<td>Code and Testing</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Organizational Process Assets cover:
- governance, policies, standards, training, tailoring guidelines
- leverage Software Assurance resources (freely available) to incorporate in training & awareness
- modify SDLC to incorporate security processes and tools (should be done in phases by practitioners to determine best integration points)
- avoid drastic changes to existing development environment and allow for time to change culture and processes
- make the business case and balance the benefits
- retain upper management sponsorship and commitment to producing secure software.

* Adopted in part from “Software Assurance: Mitigating Supply Chain Risks” (DHS NCSD SwA); “What to Test from a Security Perspective for the QA Professional” (Cigital) and “Neutralizing the Threat: A Case Study in Enterprise-wide Application Security Deployments” (Fortify Software & Accenture Security Technology Consulting)
We are engaged with many parts of the Community for Software Assurance-related standardization.
ISO/IEC/IEEE 15026, System and Software Assurance

“System and software assurance focuses on the management of risk and assurance of safety, security, and dependability within the context of system and software life cycle.

Terms of Reference changed: ISO/IEC JTC1/SC7 WG7, previously “System and Software Integrity” SC7 WG9
ISO/IEC/IEEE 15026 Assurance Case

Set of structured assurance claims, supported by evidence and reasoning (arguments), that demonstrates how assurance needs have been satisfied.

- Shows compliance with assurance objectives
- Provides an argument for the safety and security of the product or service.
- Built, collected, and maintained throughout the life cycle
- Derived from multiple sources

Sub-parts

- A high level summary
- Justification that product or service is acceptably safe, secure, or dependable
- Rationale for claiming a specified level of safety and security
- Conformance with relevant standards & regulatory requirements
- The configuration baseline
- Identified hazards and threats and residual risk of each hazard / threat
- Operational & support assumptions

Attributes

- Clear
- Consistent
- Complete
- Comprehensible
- Defensible
- Bounded
- Addresses all life cycle stages
Many DHS sponsored efforts are key to changing how software-based systems are developed, deployed and operated securely.
## The Landscape of Cyber Security Standardization Efforts

<table>
<thead>
<tr>
<th>Standard Processes</th>
<th>Standard Formats &amp; Concepts</th>
<th>Common Collections/Reference Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT</td>
<td>Cyber Security</td>
<td>IT</td>
</tr>
<tr>
<td><strong>Pre-Deployment Phase</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12207: Life cycle processes for SW</td>
<td>Common Criteria</td>
<td>OMG KDM - Knowledge Discovery Metamodel</td>
</tr>
<tr>
<td>16326: Project Mgmt</td>
<td></td>
<td>OMG SBVR - Symantec Business Vocabulary and Rules</td>
</tr>
<tr>
<td>15939: Measurement</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16085: Risk Management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15288: Life cycle processes for systems</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Post-Deployment Operations Phase</strong></td>
<td>27000</td>
<td>SP800-117</td>
</tr>
<tr>
<td>ITIL</td>
<td>SP800-53 and 53a</td>
<td>SP800-126</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X.CVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X.CVSS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X.OVAL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X.XCCDF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X.CCE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X.CPE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X.CWE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X.CAPEC</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X.CEE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X.MAEC</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X.CYBIEF</td>
</tr>
</tbody>
</table>
## THE GOAL

Qualified system and SW engineers...  
... applying sound processes ...  
... using appropriate assurance tools ...  
... delivered and deployed securely ...  
... and operated securely ...  
... all based on a commonly understood nomenclature...  
... aware of emerging assurance issues...  
... adapted for assurance considerations ...

---

### Measuring Cyber Security

- **SWEBOK**
  - Security KA
- **SE2004 curriculum**
  - Curriculum proposals
- **ABET accreditation**
- **IEEE CSDP**
  - Assurance-related questions
- **ISSA CCLSP**
  - Assurance-related questions

---

### Measuring Cyber Security SOAR

- 24748 Guide to life cycle management
- 15026 SW and systems assurance
- 24772 Prog Language vulnerabilities

---

### SWA CBOK

- 15288 System LC processes
- 12207 SW LC processes
- 15289 Documentation
- 15939 Measurement
- 16085 Risk management
- 16326 Management

---

### General Models

- Process considerations
- Assurance case
- OMG Models for the assurance case

---

### Common Criteria

- Common Criteria
- OMB FDCC/SCAP
- Supply chain studies...

---

### Program Language vulnerabilities

- Programming language standards of SC22 and others

---

### Supply Chain

- NIST Checklists
  - Secure Configuration Guides
- X.CYBIEF

---

### NVD, CVE, OVAL, XCCDF, CVSS, CPE, CCE, CWE, CAPEC, CEE, MAEC
Many SwA Resources Focus On Development
Process Improvement Lifecycle - A Process for Achieving Assurance

Understand Your Business Requirements for Assurance

Understand Assurance-Related Process Capability Expectations

Measure Your Results

Build or Refine and Execute Your Assurance Processes

Information System

Look to Standards for Assurance Process Detail

Organization Support

Adapted from: Paul Croll, Computer Sciences Corporation, August 2007
The Assurance PRM Is A Holistic Framework

Define Business Goals
- Development Organization
  - DO 1 Establish the assurance resources to achieve key business objectives
  - DO 2 Establish the environment to sustain the assurance program within the organization

- Acquisition and Supplier Management
  - AM 1 Select, manage, and use effective suppliers and third party applications based upon their assurance capabilities.

- Development Engineering
  - DE 1 Establish assurance requirements
  - DE 2 Create IT solutions with integrated business objectives and assurance
  - DE 3 Verify and Validate an implementation for assurance

- Development Project
  - DP 1 Identify and manage risks due to vulnerabilities throughout the product and system lifecycle
  - DP 2 Establish and maintain assurance support from the project
  - DP 3 Protect project and organizational assets

Prioritize funds and manage risks

Enterprise Assurance Support
- ES 1 Establish and maintain organizational culture where assurance is an integral part of achieving the mission
- ES 2 Establish and maintain the ability to support continued delivery of assurance capabilities
- ES 3 Monitor and improve enterprise support to IT assets

Enable Resilient Technology
- Sustained environment to achieve business goals through technology

Created to facilitate Communication Across An Organization’s Multi-Disciplinary Stakeholders

 Courtesy of Michele Moss, BAH, SwA Processes & Practices

https://buildsecurityin.us-cert.gov/swa/proself_assm.html
The DHS SwA Processes and Practices Working Group has synthesized the contributions of leading government and industry experts into a set of high-level goals and supporting practices (an evolution of the SwA community’s Assurance Process Reference Model)

The goals and practices are mapped to specific industry resources providing additional detail and real world implementation and supporting practices

- Assurance Focus for CMMI
- Building Security In Maturity Model
- Open Software Assurance Maturity Model
- CERT® Resilience Management Model
- CMMI for Acquisition
- CMMI for Development
- CMMI for Services
- SwA Community’s Assurance Process Reference Model – Initial Mappings
- SwA Community’s Assurance Process Reference Model - Self Assessment
- SwA Community’s Assurance Process Reference Model – Mapping to Assurance Models

Other valuable resources that are in the process of being mapped include

- NDIA System Assurance Guidebook
- Microsoft Security Development Lifecycle
- SAFECode
Our Assurance Capability Framework Enables Communication

Project leadership and team members need to know where and how to contribute

- Assurance PRM defines the goals and practices needed to achieve SwA
- Assurance for CMMI ® defines the Assurance Thread for Implementation and Improvement of Assurance Practices that are assumed when using the CMMI-DEV

Understanding gaps helps suppliers and acquirers prioritize organizational efforts and funding to implement improvement actions

https://buildsecurityin.us-cert.gov/swa/procresrc.html
DHS Software & Supply Chain Assurance Outreach

- Co-sponsor SSCA Forum & WGs for government, academia, and industry to facilitate ongoing public-private collaboration.
- Provide SwA presentations, workshops, and tracks at conferences
- Co-sponsor issues of CROSSTALK to “spread the word”
  - Sep/Oct 2009 issue on “Resilient Software”
  - Mar/Apr 2010 issue on “System Assurance”
  - Sep/Oct 2010 issue on “Game Changing Tools & Practices”
  - Mar/Apr 2011 issue on “Rugged Software”
  - Sep/Oct 2011 issue on “Protecting against Predatory Practices”
  - Mar/Apr 2012 issue on “Securing a Mobile World”
  - Sep/Oct 2012 issue on “Resilient Cyber Ecosystem”
  - Mar/Apr 2013 issue on “Supply Chain Risk Management”
  - Sep/Oct 2013 issue on “Securing the Cloud”
  - Mar/Apr 2014 issue on “Mitigating Risks from Counterfeit & Tainted Products”
- Collaborate with standards organizations, consortiums, professional societies, education/training initiatives in promoting SwA
- Provide free SwA resources via “BuildSecurityIn” website to promote secure development methodologies (since Oct 05)
- Host SSCA Community Resources & Information Clearinghouse via https://buildsecurityin.us-cert.gov/SwA
April 2009 SwA Report provides background, context and examples:

• Motivators
• Cost/Benefit Models Overview
• Measurement
• Risk
• Prioritization
• Process Improvement & Secure Software
• Globalization
• Organizational Development
• Case Studies and Examples
The Center for Internet Security

Practical Measurement Framework for Software Assurance and Information Security

February 9, 2009

Consensus Metric Definitions

Organizations struggle to make cost-effective security investments; information security professionals lack widely accepted and unambiguous metrics for decision support. CIS established a consensus team of over 100 industry experts to address this need. The result is a set of standard metrics and data definitions that can be used across organizations to collect and analyze data on security preventive performance and outcomes.

This document contains over sixty (60) metrics definitions for six (6) important business functions: Incident Management, Vulnerability Management, Patch Management, Application Security, Configuration Management and Financial Metrics. Additional consensus metrics are currently being defined for these and additional business functions.
Measurement Guidance: Purpose

- To provide a practical framework for measuring software assurance achievement of SwA goals and objectives within the context of individual projects, programs, or enterprises.
  - Making informed decisions in the software development lifecycle related to information security compliance, performance, and functional requirements/controls
  - Facilitate adoption of secure software design practices
  - Mitigate risks throughout the System Development Lifecycle (SDLC) and ultimately reduce the numbers of vulnerabilities introduced into software code during development
  - Determining if security/performance/trade-offs have been defined and accepted
  - Assessing the trustworthiness of a system.
- Can be applied beyond SwA to a variety of security-related measurement efforts to help facilitate risk-based decision making through providing quantitative information on a variety of aspects of organization’s security related performance.
Software Assurance Ecosystem: The Formal Framework

The value of formalization extends beyond software systems to include related software system process, people and documentation.

---

**Process, People & Documentation Evaluation Environment**
- Some point tools to assist evaluators but mainly manual work
- Claims in Formal SBVR vocabulary
- Evidence in Formal SBVR vocabulary
- Large scope requires large effort

**Software System / Architecture Evaluation**
- Many integrated & highly automated tools to assist evaluators
- Claims and Evidence in Formal vocabulary
- Combination of tools and ISO/OMG standards
- Standardized SW System Representation In KDM
- Large scope capable (system of systems)
- Iterative extraction and analysis for rules

---

**Claims, Arguments and Evidence Repository**
- Formalized in SBVR vocabulary
- Automated verification of claims against evidence
- Highly automated and sophisticated risk assessments using transitive inter-evidence point relationships

---

**Reports Risk Analysis, etc)**

---

**IA Controls**

**CWE**

---

**Hardware Environment**

**Software System Artifacts**

**Process Docs & Artifacts**

**Requirements/Design Docs & Artifacts**

---

**Process, People, documentation Evidence**

**Formalized Specifications**

**Executable Specifications**

**Software system Technical Evidence**

---
Software Assurance Curriculum Project

**Vol I: Master of Software Assurance Reference Curriculum**

In Dec 2010 the IEEE Computer Society and the ACM recognized the Master of Software Assurance (MSwA) Reference Curriculum as a certified master’s degree program in SwA—the first curriculum to focus on assuring the functionality, dependability, and security of software and systems.

**Vol II: SwA Undergraduate Course Outlines**

see [www.sei.cmu.edu/library/abstracts/reports/10tr019.cfm](http://www.sei.cmu.edu/library/abstracts/reports/10tr019.cfm) to download the PDF version of the report CMU/SEI-2010-TR-019

**Vol III: Master of SwA Course Syllabi**

**Vol IV: Community College Education**

- Report on “Integrating the MSwA Reference Curriculum into Model Curriculum and Guidelines for Graduate Degree Programs in Information Systems” provides reference and guidance material.
- To facilitate implementation, the MSwA project team is offering assistance, free of charge, to educational institutions looking to launch an MSwA degree program.
- For more information, go to [https://buildsecurityin.us-cert.gov/bsi/1165-BSI.html](https://buildsecurityin.us-cert.gov/bsi/1165-BSI.html).
Software & Supply Chain Assurance Strategy

Security Enumerations & Languages, Information Sharing and Analysis

Enable Automation

Support Response
Outreach & Collaborate

Manage Supply Chain Risks For Software, Hardware and Services

Influence Policy

Influence Standards

Standards are a foundation of good policy

Policy decisions influence tool design
Tools allow standards and policies to work

Standards are implemented by tools
Tools allow standards and policies to work

Policies implement standards

CRR with ExtDep Contract Reviews Business Due-Diligence CFIUS Reviews Etc.

Publications, Websites, FedVTE, Training & Education Working Groups, Forums

ISO/IEC JTC1 The Open Group OMG ITU-T CYBEX NIST Spec Pubs SP800-161, 160 & 53

US Fed Dept Mgmt & IT Acquisition

Stakeholder Engagement and Cyber Infrastructure Resilience
Exploitable Software Weaknesses (CWEs) are exploit targets/vectors for future Zero-Day Attacks
Software Assurance

Software Assurance (SwA) is the level of confidence that software functions as intended and is free of vulnerabilities, either intentionally or unintentionally designed or inserted as part of the software throughout the life cycle.*

* Derived From: CNSSI-4009

Automation

Languages, enumerations, registries, tools, and repositories throughout the Lifecycle

Including design, coding, testing, deployment, configuration and operation
Cyber Threats Emerged Over Time

1980's
- Password guessing
- Exploiting known vulnerabilities
- Burglaries
- Packet spoofing
- Hijacking sessions
- Back doors
- Disabling audits
- Internet social engineering attacks
- GUI intruder tools
- Network mgmt. diagnostics
- Sniffers

1990's
- Automated probes/scans
- Executable code attacks (against browsers)
- Automated widespread attacks
- Widespread attacks on DNS infrastructure
- Email propagation of malicious code
- “Stealth”/advanced scanning techniques

2000's
- Distributed attack tools
- Increase in wide-scale Trojan horse distribution
- Windows-based remote controllable Trojans (Back Orifice)
- Sophisticated command & control
- Anti-forensic techniques
- Diffuse spyware
- Increase in tailored worms
- DDoS attacks
- Binary encryption
- Techniques to analyze code for vulnerabilities without source code
- Widespread denial-of-service attacks

2010's
- WWW attacks
- Techniques to analyze code for vulnerabilities without source code

Attack Sophistication
Solutions Also Emerged Over Time

- **1980’s**
  - GUI intruder tools
  - Password guessing
  - Hijacking sessions
  - Password cracking
  - Exploiting known vulnerabilities
  - Packet spoofing
  - Burglaries

- **1990’s**
  - Automated widespread attacks
  - Automated widespread attacks on DNS infrastructure
  - Executable code attacks (against browsers)
  - Automated probes/scans
  - Network management diagnostics
  - Sniffers
  - Backdoors
  - Internet social engineering
  - Internet social engineering attacks

- **2000’s**
  - Email propagation of malicious code
  - Widespread attacks using NNTP to distribute attack
  - DDoS attacks
  - Binary encryption
  - Sophisticated command & control
  - Anti-forensic techniques
  - Home users targeted
  - Increase in wide-scale Trojan horse distribution
  - Increase in tailored worms
  - Sophisticated distributed attack tools
  - Increase in widespread denial-of-service attacks
  - Techniques to analyze code for vulnerabilities without source code

- **2010’s**
  - Diffuse spyware
  - Widespread denial-of-service attacks
  - Sophisticated command & control
  - Anti-forensic techniques
  - Increase in tailored worms
  - Sophisticated distributed attack tools
  - Increase in widespread denial-of-service attacks
  - Techniques to analyze code for vulnerabilities without source code

---

**Attack Sophistication**
Architecting Security with Information Standards for COIs

- Asset Management
- Vulnerability Management
- Configuration Management
- Threat Management
- System Development
- System Certification
- Intrusion Detection
- Incident Management
- Change Management
- Trust Management
- Identity Management
- Central Reporting

1980's - 2010's
### Cyber Ecosystem Standardization Efforts

<table>
<thead>
<tr>
<th>Question</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>What IT systems do I have in my enterprise?</td>
<td>CPE (Platforms)</td>
</tr>
<tr>
<td>What known vulnerabilities do I need to worry about?</td>
<td>CVE (Vulnerabilities)</td>
</tr>
<tr>
<td>What vulnerabilities do I need to worry about right now?</td>
<td>CVSS (Scoring System)</td>
</tr>
<tr>
<td>How can I configure my systems more securely?</td>
<td>CCE (Configurations)</td>
</tr>
<tr>
<td>How do I define a policy of secure configurations?</td>
<td>XCCDF (Configuration Checklists)</td>
</tr>
<tr>
<td>How can I be sure my systems conform to policy?</td>
<td>OVAL (Assessment Language)</td>
</tr>
<tr>
<td>How can I ensure operation of my systems conforms to policy?</td>
<td>OCIL (Interactive Language)</td>
</tr>
<tr>
<td>What weaknesses in my software could be exploited?</td>
<td>CWE (Weaknesses)</td>
</tr>
<tr>
<td>What attacks can exploit which weaknesses?</td>
<td>CAPEC (Attack Patterns)</td>
</tr>
<tr>
<td>How can we recognize malware &amp; share that info?</td>
<td>MAEC (Malware Attributes)</td>
</tr>
<tr>
<td>What observable behavior might put my enterprise at risk?</td>
<td>CybOX (Cyber Observables)</td>
</tr>
<tr>
<td>How can I share threat information?</td>
<td>STIX (Structure Threat Information)</td>
</tr>
<tr>
<td>What events should be logged, and how?</td>
<td>CEE (Events)</td>
</tr>
<tr>
<td>How can I aggregate assessment results?</td>
<td>ARF (Assessment Results)</td>
</tr>
</tbody>
</table>

- Many standards are XML-based; enabling automation of information exchange
- Several standards support multiple enterprise cybersecurity functions
The Structured Threat Information eXpression

- A framework/data model to standardize the representation of cyber threat intelligence
- Builds on existing languages/models where possible
- Provides a structure that enables:
  - Consistent semantics
  - Automated interpretation
  - Advanced analysis
- Enables the expression of relationships between entities within the framework:
  - E.g. threat actor $A$ uses TTP $B$ which can be detected via indicator $C$
STIX: Primary Components

- What activity are we seeing?
- What threats should I look for?
- Where has this threat been seen?
- What does it do?
- What weaknesses does it exploit?
- Why does it do this?
- Who is responsible for this threat?
- What can I do about it?
What could/should have been done to harden the attack surface/vector to prevent the target from being exploitable?
Leverage Common Weakness Enumeration (CWE) to mitigate risks to mission/business domains

CWE is a formal list of software weakness types created to:
- Serve as a common language for describing software security weaknesses in architecture, design, or code.
- Serve as a standard measuring stick for software security tools targeting these weaknesses.
- Provide a common baseline standard for weakness identification, mitigation, and prevention efforts.

Some Common Types of Software Weaknesses:

| Buffer Overflows, Format Strings, Etc. | Errors |
| Structure and Validity Problems | Authentication Errors |
| Common Special Element Manipulations | Resource Management Errors |
| Channel and Path Errors | Insufficient Verification of Data |
| Handler Errors | Code Evaluation and Injection |
| User Interface Errors | Randomness and Predictability |
| Pathname Traversal and Equivalence | |

cwe.mitre.org
CVE – enabling reporting and patching of vulnerabilities

CWE – identifying and mitigating root cause exploitable weaknesses

CybOX – cyber observables and supply chain exploit indicators

CAPEC – schema attack patterns and software exploits
CWRAF/CWSS Provides Risk Prioritization for CWE throughout Software Life Cycle

• Enables education and training to provide specific practices for eliminating software fault patterns;
• Enables developers to mitigate top risks attributable to exploitable software;
• Enables testing organizations to use suite of test tools & methods (with CWE Coverage Claims Representation) that cover applicable concerns;
• Enables users and operation organizations to deploy and use software that is more resilient and secure;
• Enables procurement organizations to specify software security expectations through acquisition of software, hosted applications and services.
When should I focus on Weaknesses and Vulnerabilities?

Focus on Weaknesses
A type of defect that *may be* exploitable.

Keep Weaknesses from becoming vulnerabilities

Focus on Vulnerabilities
Something in code that *can actually* be exploited.

Concept  Design  Source code  Object code  Binaries  Deployed system
Software Assurance (SwA) Competency Model, March 2013

Developed to support the following uses:

• Provide employers of SwA personnel with a means to assess the SwA capabilities of current and potential employees.

• Offer guidance to academic or training organizations:
  – develop SwA courses to support the needs of organizations that are hiring and developing SwA professionals.
  – Enhance SwA curricula guidance by providing information about industry needs and expectations for competent SwA professionals.

• Provide direction and a progression for the development and career planning of SwA professionals.

• Provide support for professional certification and licensing.
ICT/software security risk landscape is a convergence between “defense in depth” and “defense in breadth”

Risk shifts to end-points; Enterprise Risk Management and Governance are security motivators

Acquisition could influence the lifecycle; more than development

Software & Supply Chain Assurance provides a focus for:
-- Resilient Software and ICT Components,
-- Security in the Component Life Cycle,
-- Software Security in Services, and
-- Supply Chain Risk Management (mitigating risks of counterfeit & tainted products)
What Are We Protecting?

Program Protection Planning

**Technology**
- What: Leading-edge research and technology
- Who Identifies: Technologists, System Engineers
- ID Process: CPI Identification
- Threat Assessment: Foreign collection threat informed by Intelligence and Counterintelligence assessments
- Countermeasures: AT, Classification, Export Controls, Security, Foreign Disclosure, and CI activities
- Focus: “Keep secret stuff in” by protecting any form of technology

**Components**
- What: Mission-critical elements and components
- Who Identifies: System Engineers, Logisticians
- ID Process: Criticality Analysis
- Threat Assessment: DIA SCRM TAC
- Countermeasures: SCRM, SSE, Anti-counterfeits, software assurance, Trusted Foundry, etc.
- Focus: “Keep malicious stuff out” by protecting key mission components

**Information**
- What: Information about applications, processes, capabilities and end-items
- Who Identifies: All
- ID Process: CPI identification, criticality analysis, and classification guidance
- Threat Assessment: Foreign collection threat informed by Intelligence and Counterintelligence assessments
- Countermeasures: Information Assurance, Classification, Export Controls, Security, etc.
- Focus: “Keep critical information from getting out” by protecting data

Protecting Warfighting Capability Throughout the Lifecycle

Note: Program Protection Planning Includes DoDI 8500 series
SSCA Focus on Tainted Components

*Mitigating risks attributable to exploitable non-conforming constructs in ICT*

“Tainted” products are those that are corrupted with malware, or exploitable weaknesses & vulnerabilities that put users at risk

- Enable ‘scalable’ detection and reporting of tainted ICT components
- Leverage/mature related existing standardization efforts
- Provide Taxonomies, schema & structured representations with defined observables & indicators for conveying information:
  - Tainted constructs:
    - Malicious logic/malware (MAEC),
    - Exploitable Weaknesses (CWE);
    - Vulnerabilities (CVE)
  - Attack Patterns (CAPEC)
- Catalogue Diagnostic Methods, Controls, Countermeasures, & Mitigation Practices
- Publicly reported weaknesses and vulnerabilities with patches accessible via National Vulnerability Database (NVD) sponsored by DHS & hosted by NIST

Components can become tainted intentionally or unintentionally throughout the supply chain, SDLC, and in Ops & sustainment

*Text demonstrates examples of overlap*
Scope of UL Cybersecurity Assurance Program

UL CAP focus on Network-Connectable Devices

- Addresses known vulnerabilities (CVSS+) at the time of certification (i.e. CVEs catalogued in the NVD)
- Performs baseline weakness assessment for potential “zero day” vulnerabilities (CWSS and rankings from other organizations)
- Addresses known malware at time of certification

Addressing the most relevant CWEs, establishes a baseline to mitigate weaknesses that, if otherwise exploited, could be vectors of attack; becoming zero-day vulnerabilities

* Example of programs using CVE, CWE, CWSS, CVSS, CAPEC, etc
ICT/Software & Supply Chain Assurance is a National Security & Economic Issue

- Adversaries can gain “intimate access” to target systems, especially in a global supply chain that offers limited transparency.

- Advances in science and technology will always outpace the ability of government and industry to react with new policies and standards.
  - National security policies must conform with international laws and agreements while preserving a nation’s rights and freedoms, and protecting a nation’s self interests and economic goals;
  - Forward-looking policies can adapt to the new world of global supply chains;
  - Standards for automation, processes, and products must mature to better address supply chain risk management, systems/software assurance, and the exchange of information and indicators for cyber security;
  - Assurance Rating Schemes for ICT/software products and suppliers are needed.

- ICT/software suppliers and buyers can take more deliberate actions to security-enhance their processes, practices and products to mitigate risks.
  - Government & Industry have significant leadership roles in solving this
  - Individuals can influence the way their organizations adopt security practices

Globalization will not be reversed; this is how we conduct business – To remain relevant, standards and capability benchmarking measures must address “assurance” mechanisms needed to manage IT/Software Supply Chain risks.
Public-Private Collaboration Efforts for Security Automation, Software Assurance, and Supply Chain Risk Management
Software & Supply Chain Assurance:

Enabling Enterprise Resilience through Security Automation, Software Assurance, and Supply Chain Risk Management

Joe Jarzombek, PMP, CSSLP
Director for Software & Supply Chain Assurance
Cyber Security & Communications
joe.jarzombek@hq.dhs.gov (until 31 Dec 2015)
sjoejazz@aol.com

Mitigating Risk Exposures Attributable to Exploitable ICT Products and Services