Software Assurance

Public/Private Collaboration Efforts for Enterprise Security Automation

Sept 27, 2010

Joe Jarzombek, PMP, CSSLP
Director for Software Assurance
National Cyber Security Division
Office of the Assistant Secretary for Cybersecurity and Communications
Software Assurance (SwA) – Security Automation

10:45am - SwA Panel: Use Cases, Standards and Roadmap for Enterprise Security Automation
11:45am - Knowing Your Weaknesses (CWE)
1:30pm - Ranking Your Weaknesses (CWSS)
2:30pm - Understanding How They Attack Your Weaknesses (CAPEC)
3:45pm - Sharing Understanding of Malware (MAEC)
4:45pm - Panel on SwA Automation Protocol
SwA Panel: Use Cases, Standards and Roadmap for Enterprise Security Automation

• Panel Facilitator – Joe Jarzombek, DHS NCSD
• Relevant International Standards – Don Davidson, DoD
• Enterprise Security Automation – Bob Martin, MITRE
• Use Cases for Security Automation – Dan Schmidt, NSA and Tim Grance, NIST
Software Assurance (SwA) – Security Automation

- Security Content Automation Protocol (SCAP)
- Software Assurance Automation Protocol (SwAAP)
- Enterprise System Information Protocol (ESIP)
- Enterprise Remediation Automation Protocol (ERAP)
- Enterprise Compliance Automation Protocol (ECAP)
- Event Management Automation Protocol (EMAP)
- Incident Tracking and Assessment Protocol (ITAP)
- Threat Analysis Automation Protocol (TAAP)

Use Cases for Enterprise IT Security
<table>
<thead>
<tr>
<th></th>
<th>SCAP</th>
<th>SwAAP</th>
<th>ESIP</th>
<th>ERAP</th>
<th>ECAP</th>
<th>EMAP</th>
<th>ITAP</th>
<th>TAAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>CVE</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>OVAL</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>XCCDF</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CVRF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OCIL</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>CPE</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>CCE</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CWE</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CAPEC</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>MAEC</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>SCAP</td>
<td>SwA</td>
<td>ESIP</td>
<td>ERAP</td>
<td>ECAP</td>
<td>EMAP</td>
<td>ITAP</td>
<td>TAAP</td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
<td>-----</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>CEE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>CRE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>ERI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>ARF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>OCRL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>IODEF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>NIEM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CYBEX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>
Software Assurance (SwA) – Security Automation

Panel on Software Assurance Automation Protocol
Facilitator: Joe Jarzombek, DHS NCSD
Steve Quinn, NIST
Dan Schmidt, NSA
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
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
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
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.

---

**Process Docs & Artifacts**
- Requirements/Design Docs & Artifacts

---

**Hardware Environment**

---

**Software System Artifacts**
SCAP 1.1 uses the following specifications:

- Extensible Configuration Checklist Description Format (XCCDF) 1.1.4, a language for authoring security checklists/benchmarks and for reporting results of checklist evaluation [QUI08]
- Open Vulnerability and Assessment Language (OVAL) 5.6, a language for representing system configuration information, assessing machine state, and reporting assessment results
- Open Checklist Interactive Language (OCIL) 2.0, a language for representing security checks that requires human feedback
- Common Platform Enumeration (CPE) 2.2, a nomenclature and dictionary of hardware, operating systems, and applications [BUT09]
- Common Configuration Enumeration (CCE) 5, a nomenclature and dictionary of system configurations
- Common Vulnerabilities and Exposures (CVE), a nomenclature and dictionary of software flaws
- Common Vulnerability Scoring System (CVSS) 2.0, an open specification for scoring the severity of software flaw vulnerabilities [MEL07].
4. **SCAP General Requirements and Conventions**

4.1 Support for Legacy SCAP Versions ........................................... 4-1
4.2 XCCDF Conventions and Requirements ..................................... 4-1
   4.2.1 Metadata Elements ....................................................... 4-1
   4.2.2 Use of CPE Names ....................................................... 4-2
   4.2.3 The `<xccdf:Benchmark>` Element .................................... 4-3
   4.2.4 The `<xccdf:Profile>` Element ......................................... 4-3
   4.2.5 The `<xccdf:Rule>` Element ........................................... 4-4
   4.2.6 Allowed Check System Usage ......................................... 4-5
   4.2.7 XCCDF Test Results ..................................................... 4-10
4.3 OVAL Conventions and Requirements ....................................... 4-12
   4.3.1 Supported Previous Versions of OVAL (5.3, 5.4, and 5.5) ........ 4-13
   4.3.2 Support for Deprecated Constructs in OVAL ..................... 4-14
   4.3.3 OVAL Schema Specification ........................................... 4-14
   4.3.4 OVAL Results ............................................................ 4-14
4.4 OCIL Conventions .................................................................. 4-15
4.5 CPE Conventions .................................................................. 4-15
4.6 CCE Conventions .................................................................. 4-15
4.7 CVE Conventions .................................................................. 4-15
4.8 CVSS Conventions ................................................................ 4-15

---


Recommendations of the National Institute of Standards and Technology

Stephen Quinn
David Waltermire
Christopher Johnson
Karen Scarfone
John Banghart
5. SCAP Use Case Requirements

5.1 SCAP Data Streams

5.2 SCAP Configuration Verification

5.3 SCAP Vulnerability Assessment

5.3.1 SCAP Vulnerability Assessment Using XCCDF and OVAL

5.3.2 SCAP Vulnerability Assessment Using Standalone OVAL

5.3.3 OVAL Definitions and Vulnerability Assessment

5.4 Patch Validation

5.4.1 Using OVAL Definitions for Patch Validation

5.4.2 Referencing an OVAL Patch Data Stream

5.5 SCAP Inventory Collection
Software Assurance Automation Protocol (SwAAP) - For measuring & enumerating software weaknesses and the assurance cases.


• plus SCAP to capture “accredited” system CPEs and CCE settings?
• OVAL checks for capturing “finger print” of software applications to address supply-chain risk measurement?
“Other” Automation Protocols ("O"AP)

- Event Management Automation Protocol (EMAP)
  - For reporting of security events.
  - Uses Common Event Expression (CEE), Malware Attribute Enumeration & Characterization (MAEC), CAPEC, etc.

- Enterprise Remediation Automation Protocol (ERAP)
  - For automated remediation of mis-configuration & missing patches.
  - Uses Common Remediation Enumeration (CRE) and Extended Remediation Information (ERI).

- Enterprise Compliance Automation Protocol (ECAP)
  - For reporting configuration compliance.
  - Uses Asset Reporting Format (ARF), Open Checklist Reporting Language (OCRL), etc.

- Enterprise System Information Protocol (ESIP)
  - For reporting of asset inventory information.
  - Uses ..... 

- Threat Analysis Automation Protocol (TAAP)
  - For analyzing threats and security risks.
  - Uses....

- Incident Management Automation Protocol (IMAP)
  - For supporting incident management and response.
  - Uses IODEF, etc
Enterprise IT Asset Management

Asset Inventory

Configuration Guidance Analysis

Vulnerability Analysis

Threat Analysis

Intrusion Detection

Incident Management

Operational Enterprise Networks

Centralized Reporting

Enterprise IT Change Management

Operational Security Management Processes

Assessment of System Development, Integration, & Sustainment Activities and Certification & Accreditation

Scanning, Configuration, and Analysis (SCAP)

Threat Assessment and Analysis (TAAP)

Enterprise IT Asset Management (IMAP)

SwAAP

Development & Sustainment Security Management Processes

ERAP

ECAP
Next SwA Forum 27 Sep – 1 Oct 2010 at NIST, Gaithersburg, MD

SOFTWARE ASSURANCE FORUM

“Building Security In”

https://buildsecurityin.us-cert.gov/swa

Joe Jarzombek, PMP, CSSLP
Director for Software Assurance
National Cyber Security Division
Department of Homeland Security
Joe.Jarzombek@dhs.gov
(703) 235-5126
LinkedIn SwA Mega-Community
Next SwA Forum 27 Sep – 1 Oct 2010 at NIST, Gaithersburg, MD