Joint Technology, Tools and Product Evaluation /Malware Working Group Outbrief

Michael Kass, TT&PE Co-Chair
Larry Wagoner, TT&PE Co-Chair
Penny Chase, Malware WG Co-Chair
June 23, 2010

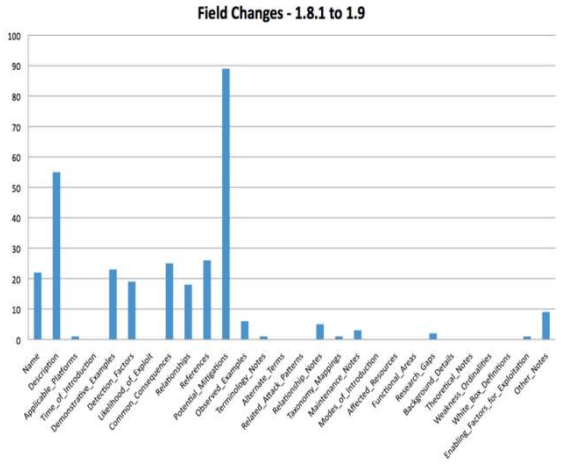


- Workshop Goals
  - Provide updates on CWE, CAPEC, MAEC, CWSS
  - Give stakeholders a quick overview and update of other enabling technologies that can be brought to bear in a software assurance automation protocol for enterprise assessment and reporting, including:
    - OMG Structured Assurance Case and Ecosystem Standards, ISO15026, and SCAP
  - Panel: Give stakeholders a high-level view of issues/obstacles to implementing such a broadly integrated protocol

Homeland Security

CWE Update – Conor Harris, MITRE (cwe.mitre.org)

- Version 1.9 released on Monday, June 21
- Graphical view of deltas between versions
  - V1.81 – 1.9 (***mitigations***, descriptions)
- Normalize common mitigations across Top 25
  - Analyze all Top 25 mits, merge overlap
  - Why is this good?
    - Higher quality
    - Easier to fill in (new entries and gaps)
    - Single point of maintenance
- Also working on:
  - Top 25 Mitigations
  - Top 25 Consequences
  - Vocabulary (reduced usage of “overloaded” words)

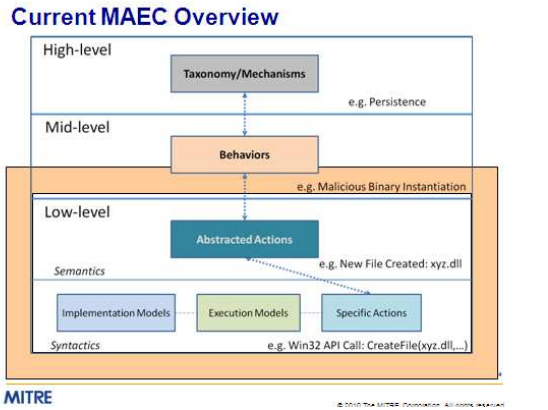


- Where is CAPEC today?
- V1.4
  - Massive schema changes
  - Some new content
  - Added initial set of network attack patterns
- V1.5
  - Added ~25 new network attack patterns
  - Added enhanced material to ~35 patterns
  - New View added for WASC Threat Taxonomy 2.0
  - Added ~65 mappings to CWE and several within CAPEC
- Currently 311 patterns, stubs, named attacks; 67 categories and 6 views

- CAPEC Future Plans
  - V1.6
    - At least 30 more patterns fleshed out from hooks & stubs to full
    - Some network attack patterns with fleshed out attack surface info
    - Initial set of supply chain and social engineering attack patterns
    - Potentially refined “observables” schema
  - Strategic focus for the near to mid-term will be on utilizing CAPEC as a bridge between secure development and secure operations
  - Continue expanding and refining content
  - Continue expanding outreach and supporting CAPEC use
  - Establish initial CAPEC compatibility program

- Current Content (12 “major” categories)
  - 1000 - Mechanism of Attack
  - Data Leakage Attacks - (118)
  - Resource Depletion - (119)
  - Injection (Injecting Control Plane content through the Data Plane) - (152)
  - Spoofing - (156)
  - Time and State Attacks - (172)
  - Abuse of Functionality - (210)
  - Exploitation of Authentication - (225)
  - Probabilistic Techniques - (223)
  - Exploitation of Privilege/Trust - (232)
  - Data Structure Attacks - (255)
  - Resource Manipulation - (262)
  - Network Reconnaissance - *(286)*

- MAEC is being developed as a formal language for characterizing attributes and behaviors of all types of malware
- Described:
  - Vocabulary, Grammar and Output Format
  - Action and Behavior Model
  - Schema
  - Dynamic Malware Analysis
    - Example case
  - Collaboration
    - Engaged with IEEE Industry Connections Security Group (ICSG) Malware Group
    - Community Discussion List: http://maec.mitre.org/community/discussionlist.html
    - MAEC Development Group
      - Handshake (social networking)



Homeland Security

- Future plans:
  - Develop additional translators for dynamic analysis tools into MAEC XML
  - Begin creation of process for cataloguing malware behaviors
  - Expand and revise schema, particularly with regards to action attributes
  - Collaborate with CAPEC & CWE teams in order to develop consensus approach on object/observable development
  - Encourage and invite more participation in the development process
    - MAEC Website: http://maec.mitre.org (contains MAEC Discussion list sign-up)
    - MAEC Handshake Group

Homeland Security

- Anyone who's doing application security will have to prioritize the reported weaknesses
  - SATE revealed different prioritization by tools
  - Hundreds/thousands of bug reports per package is typical
- Analytical methods will vary - whitebox/blackbox, manual/automatic,
- Want to prioritize results in a consistent way
- Need to balance general guidance with specific findings
- Want to address the needs of multiple stakeholders
- Where possible, borrow from other work such as CVSS

Homeland Security

- *"I ran a tool that gave me hundreds of results. Which issues should I address, based on:"*
    - my company's goals
    - compliance requirements
    - customer expectations
    - best current practices
    - amount of time before next release

- Software developers/programmers
- Software project managers
- SW acquirers
  - “The purchased software shall not have any outstanding weaknesses greater than CWSS score 7.0, as determined by methods X and Y.”
  - Large enterprise organizations
- Code analysis vendors - tools and services
- Vulnerability researchers
- Secure development advocates
- CIO’s
- System administrators
- Application users

- Inherent Risk
  - Known-exploitable vs. unlikely? (e.g. buffer overflows)
- Inherent Attack Complexity
  - Level of authentication required
  - Local/remote
  - Amount of victim interaction required
- Execution Environment
  - Specific to some or all configurations? Default?
- Threat Environment
  - Script kiddies or nation-state? Financial or physical?
- Business/Mission Priority
  - "Top 25" list conformance, customer needs, software vendor training/education

Known to tools

Known to developers

Known to customers

- Incomplete information is the norm
- Environmental/business/mission must be considered
- Support for multiple scoring modes
  - General (Prevention): score weaknesses based on their general occurrence in software
  - Specific(Reaction)score a weakness based on its occurrence within a specific software package
- Stakeholder analysis needed
- Metric complexity is a risk

- Upcoming white paper
- Build community
  - Software vendors
  - Automated code analysis vendors
  - Current CVSS community
  - Academic community
  - Researchers/consultants

- A quick overview of all elements that may comprise a software assurance automation protocol SwAAP

- Provide a basic level of understanding for our stakeholders in preparation for the follow-on panel discussion

- **Software Assurance Automation Protocol (SwAAP)**
  - For measuring & enumerating software weaknesses and the assurance cases.

Common Weakness Enumeration (CWE),

Common Attack Pattern Enumeration & Classification (CAPEC),

Malware Attribute Enumeration & Characterization (MAEC),

Common Weakness Scoring System (CWSS),

OMG Software Assurance Evidence Metamodel (OMG SAEM),

OMG Argumentation Metatmodel (OMG ARG),

Software Assurance Findings Expression Schema (SAFES),

NIST SAMATE's "Food Label",

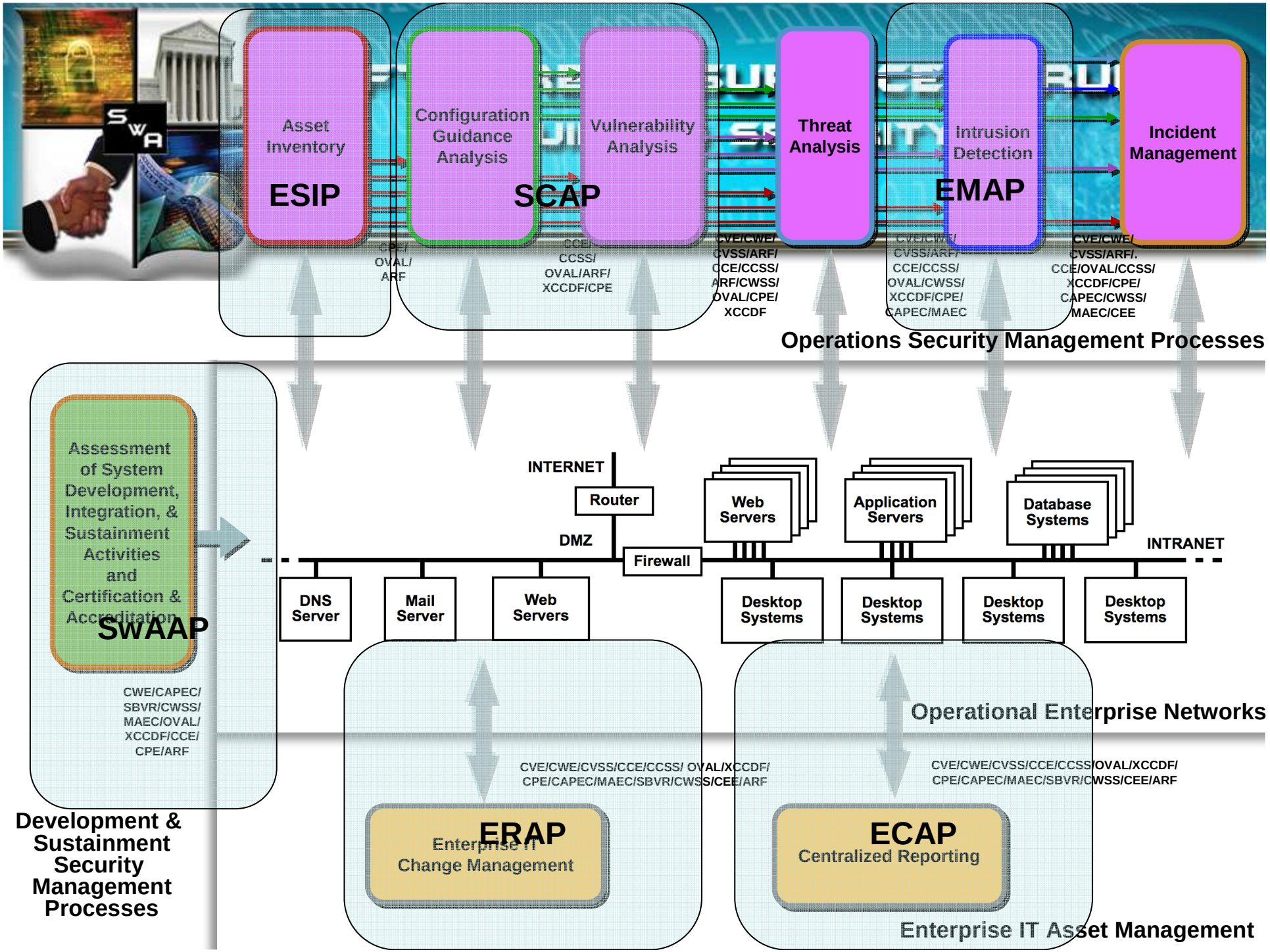
OMG Structured Metrics Metamodel (OMG SMM),

ISO "Assurance Case" 15026 (ISO 15026),

OMG Knowledge Discovery Metamodel (OMG KDM),

OMG Abstract Syntax Tree Metamodel (OMG ASTM)

- **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?**


Asset Inventory
ESIP
Configuration Guidance Analysis
Vulnerability Analysis
SCAP
Threat Analysis
Intrusion Detection
EMAP
Incident Management
CPE/ OVAL/ ARF
CCE/ CCSS/ OVAL/ARF/ XCCDF/CPE
CVE/CWE/ CVSS/ARF/ CCE/CCSS/ ARF/CWSS/ OVAL/CPE/ XCCDF
CVE/CWE/ CVSS/ARF/ CCE/CCSS/ OVAL/CWSS/ XCCDF/CPE/ CAPEC/MAEC
CVE/CWE/ CVSS/ARF/. CCE/OVAL/CCSS/ XCCDF/CPE/ CAPEC/CWSS/ MAEC/CEE
Operations Security Management Processes
Assessment of System Development, Integration, & Sustainment Activities and Certification & Accreditation
SWAAP
CWE/CAPEC/ SBVR/CWSS/ MAEC/OVAL/ XCCDF/CCE/ CPE/ARF
INTERNET
Router
DMZ
Firewall
Web Servers
Application Servers
Database Systems
INTRANET
DNS Server
Mail Server
Web Servers
Desktop Systems
Desktop Systems
Desktop Systems
Desktop Systems
Operational Enterprise Networks
CVE/CWE/CVSS/CCE/CCSS/ OVAL/XCCDF/ CPE/CAPEC/MAEC/SBVR/CWSS/CEE/ARF
CVE/CWE/CVSS/CCE/CCSS/OVAL/XCCDF/ CPE/CAPEC/MAEC/SBVR/CWSS/CEE/ARF
ERAP
Enterprise IT Change Management
ECAP
Centralized Reporting
Development & Sustainment Security Management Processes
Enterprise IT Asset Management

- SQ:"SwAAP will take ~5 or 6 years of development, need to understand that. SCAP is just now maturing."
- Q: "*Do you have to try and understand all the technologies to keep yourself from going down the wrong path?"*
  - SQ: Need to look at how to represent events
    - None of these enumerations make any sense without context.
  - SB: All of these pieces have been created with a specific purpose
    - You don't have to use all of the technologies, but they can all be useful
    - Need to know what your organization needs
- Q: "*What parts of the SwAAP are low hanging fruit?"*
  - JJarzombek: CWE, CAPEC, MAEC exist already.
  - BMartin: Automation is not the only thing driving this. We need to know who and what was involved in the assuring of the system. This is the need of the Assurance Case
  - SQ: At NIST, automation doesn't mean take the human entirely, but how the human should collect the information.

- Q: “Should start creating use cases for the SwAAP to help new members of the community?”
  - SB: There are use cases for individual pieces; we just haven’t put them together.
  - DCampara:As a working group we need to build use cases for repeatable process against these enumerations
  - MDonaldson: Use case for multiple tools would be good….and operational environment use cases.. change environment and use these tools
- ASzakal:”We don’t want to be in a position where the government defines one protocol.”
- SQ:NIST established a development cycle that matures at its own rate… Once it’s submitted to NIST, NIST will evaluate. It can mature for comment from the time it becomes a part of program, with a minimum 18 months to be in a product.