Supporting Secure Software Operations

Robert A. Martin
Sean Barnum

May 2011

UNCLASSIFIED
Report Documentation Page

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE
MAY 2011

2. REPORT TYPE

3. DATES COVERED
00-00-2011 to 00-00-2011

4. TITLE AND SUBTITLE
Supporting Secure Software Operations

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

6. AUTHOR(S)

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
MITRE Corporation, 202 Burlington Road, Bedford, MA, 01730-1420

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR’S ACRONYM(S)

11. SPONSOR/MONITOR’S REPORT NUMBER(S)

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

13. SUPPLEMENTARY NOTES
Presented at the 23rd Systems and Software Technology Conference (SSTC), 16-19 May 2011, Salt Lake City, UT. Sponsored in part by the USAF. U.S. Government or Federal Rights License

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:
   a. REPORT
      unclassified
   b. ABSTRACT
      unclassified
   c. THIS PAGE
      unclassified

17. LIMITATION OF ABSTRACT
   Same as Report (SAR)

18. NUMBER OF PAGES
   32

19a. NAME OF RESPONSIBLE PERSON

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18
Agenda

8:00-8:45am  Software Security Knowledge about Applications Weaknesses

9:00-9:45am  Software Security Knowledge about Attack Patterns Against Applications

Training in Software Security

10:15-11:00am  Software Security Practice

11:15-12:00am  Supporting Capabilities

Assurance Cases

Secure Development & Secure Operations
Secure Software Operations

- Where secure development use cases required foundational knowledge and ways to package it and understand it within a static context, Secure Software Operations requires situational awareness & interpretation of foundational knowledge within a dynamic context.

- Considering that secure operations is a key element of overall software assurance we need ways to:
  - Bridge the secure development and secure operations domains
  - Improve the analysis, characterization, collection, discovery & knowledge sharing of malware
  - Combine elements of the ecosystem as practical applications to support secure software operations

- This portion of the tutorial will focus on resources/efforts focused at addressing these three needs.
Secure Software Operations

- Bridge the secure development and secure operations domains
- Improve the analysis, characterization, collection, discovery & knowledge sharing of malware
- Combine elements of the ecosystem as practical applications to support secure software operations

Cyber Observable eXpression (CybOX)

Malware Attribute Enumeration & Characterization (MAEC)

Security Content Automation Protocol (SCAP) and other Automation Protocols
Bridge the secure development and secure operations domains

Cyber Observable eXpression (CybOX)

The topic and content covered in this presentation was published as an article in the Sep/Oct 2010 issue of CrossTalk: The Journal of Defense Software Engineering
Attack Patterns Bridge Secure Development and Operations

Secure Development

Secure Operations

Attack Patterns

Attack Patterns
Secure Operations Knowledge Offers Unique Value to Secure Development

Using attack patterns makes it possible for the secure development domain to leverage significant value from secure operations knowledge, enabling them to:

- Understand the real-world frequency and success of various types of attacks.
- Identify and prioritize relevant attack patterns.
- Identify and prioritize the most critical weaknesses to avoid.
- Identify new patterns and variations of attack.

Secure Development Knowledge Offers Unique Value to Secure Operations

Attack patterns enable those in the secure operations domain to provide appropriate context to the massive amounts of data analyzed to help answer the foundational secure operations questions.
So, this all sounds great but how do we map these high-level attack patterns to the low-level operational world?

**Cyber Observables**

The Secret Sauce for Bridging the Abstract to the Concrete
Cyber Observables Overview

- The Cyber Observables construct is intended to capture and characterize events or properties that are observable in the operational domain.

- These observable events or properties can be used to adorn the appropriate portions of the attack patterns in order to tie the logical pattern constructs to real-world evidence of their occurrence or presence.

- This construct has the potential for being the most important bridge between the two domains, as it enables the alignment of the low-level aggregate mapping of observables that occurs in the operations domain to the higher-level abstractions of attacker methodology, motivation, and capability that exist in the development domain.

- By capturing them in a structured fashion, the intent is to enable future potential for detailed automatable mapping and analysis heuristics.
A Brief History of Cyber Observables

- September 2009: Concept introduced to CAPEC in Version 1.4 as future envisioned adornment to the structured Attack Execution Flow
- June 2010: Broader relevance to MSM recognized leading to CAPEC, MAEC & CEE teams collaborating to define one common structure to serve the common needs
- August 2010: Discussed with US-CERT at GFIRST 2010
- December 2010: Cyber Observables schema draft v0.4 completed
- December 2010: Discussions with Mandiant for collaboration and alignment between Cyber Observables and Mandiant OpenIOC
- January 2011: Discussed & briefed with MITRE CSOC
- February 2011: Discussed & briefed with NIST – EMAP and US-CERT who also have a need for this construct and had begun to work on parallel solutions
Simplified Overview of Current Schema
Cyber Observable Broader Use Cases

- Detect malicious activity from attack patterns
- Empower & guide incident management
- Identify new attack patterns
- Prioritize existing attack patterns based on tactical reality

- Potential ability to analyze data from all types of tools and all vendors
- Improved sharing among all cyber observable stakeholders
- Ability to metatag cyber observables for implicit sharing controls
- Enable automated signature rule generation
- Enable new levels of meta-analysis on operational cyber observables
- Potential ability to automatically apply mitigations specified in attack patterns

- Etc....
Improve the analysis, characterization, collection, discovery & knowledge sharing of malware

Malware Attribute Enumeration & Characterization (MAEC)
Malware Attribute Enumeration and Characterization (MAEC)

- **Language for sharing structured information about malware**
  - Grammar (Schema)
  - Vocabulary (Enumerations)
  - Collection Format (Bundle)

- **Focus on attributes and behaviors**

- **Enable correlation, integration, and automation**
MAEC Use Cases

■ Operational

■ Analysis
  – Help Guide Analysis Process
  – Standardized Tool Output
  – Malware Repositories
MAEC Overview

High-level

Mechanisms

e.g. Persistence

Mid-level

Behaviors

e.g. Malicious Binary Instantiation

Low-level

Abstracted Actions

e.g. Create File: xyz.dll

Implementation Models

e.g. Win32 API Call: CreateFile(xyz.dll,...)

Semantics

Syntax
Dynamic Malware Analysis → MAEC

- Demonstrate the ability to generate MAEC XML descriptions from dynamic analysis tools
- Developed proof-of-concept translators for:
  - CW Sandbox (Sunbelt)
  - ASAT (MITRE)
  - Anubis
  - ThreatExpert

**Process**

1. An API call is captured by the analysis engine and mapped to MAEC’s enumeration of API calls.
2. The MAEC enumerated call is mapped to its corresponding action.
3. The MAEC defined action is mapped to a corresponding MAEC effect (as necessary), which is populated by the parameters of the call.
4. The MAEC effect is linked to a MAEC object (as necessary).
5. Any extra data output (e.g. file attributes, network capture, etc.) from the analysis engine is mapped to its corresponding object (as necessary).
Test Case: CWSandbox Output -> MAEC

Raw CWSandbox Output

Python

XSD Bindings

MAEC XML

MAEC Actions

MAEC Objects

MAEC Behaviors

© 2010 The MITRE Corporation. All rights reserved.
Collaboration

- Related Making Security Measurable Efforts
  - There is significant overlap between MAEC, CAPEC, and CEE in describing observed actions, objects, and states.
  - As such, we’re working on developing a common schematic structure of observables for use in these efforts:
MAEC Community: Discussion List

- Request to join: [http://maec.mitre.org/community/discussionlist.html](http://maec.mitre.org/community/discussionlist.html)
- Archives available
MAEC Community: MAEC Development Group on Handshake

- MITRE hosts a social networking collaboration environment: [https://handshake.mitre.org](https://handshake.mitre.org)
- Supplement to mailing list to facilitate collaborative schema development
- Malware Ontologies SIG Subgroup
Combine elements of the ecosystem as practical applications to support secure software operations

Security Content Automation Protocol (SCAP) and other Automation Protocols
# Remembering the Acronyms

<table>
<thead>
<tr>
<th>Question</th>
<th>Acronym</th>
</tr>
</thead>
<tbody>
<tr>
<td>What IT systems do I have in my enterprise?</td>
<td>CPE (Platforms)</td>
</tr>
<tr>
<td>What 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 be sure the 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>What should be logged, and how?</td>
<td>CEE (Events)</td>
</tr>
<tr>
<td>How can I aggregate assessment results?</td>
<td>ARF (Results)</td>
</tr>
<tr>
<td>How can we recognize malware?</td>
<td>MAEC (Malware Attributes)</td>
</tr>
</tbody>
</table>
Standardization Efforts leveraged by the Security Content Automation Protocol (SCAP)

<table>
<thead>
<tr>
<th>Question</th>
<th>SCAP Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>What IT systems do I have in my enterprise?</td>
<td>• CPE (Platforms)</td>
</tr>
<tr>
<td>What 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 be sure the 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>What should be logged, and how?</td>
<td>• CEE (Events)</td>
</tr>
<tr>
<td>How can I aggregate assessment results?</td>
<td>• ARF (Results)</td>
</tr>
<tr>
<td>How can we recognize malware?</td>
<td>• MAEC (Malware Attributes)</td>
</tr>
</tbody>
</table>
SCAP – FDCC and USGCB

MEMORANDUM TO: Karen S. Evans, Administrator, E-Government and Information Technology
FROM:

SUBJECT: Guidance on the Federal Desktop Core Configuration (FDCC)

In March 2007, OMB Memorandum M-07-11 announced "Implementation of Commonly Accepted Security Configurations for Windows Operating Systems," directing agencies with Windows XP SP2 deployed and/or plans to upgrade to the Vista SP1 operating system to adopt the Federal Desktop Core Configuration (FDCC) security configurations developed by the National Institute of Standards and Technology (NIST), the Department of Defense (DOD) and the Department of Homeland Security (DHS).

On June 20, 2008, NIST published the updated Federal Desktop Core Configuration Major Version 1.0 settings release. Relative to the previous version of FDCC which was originally posted in July 2007, 40 settings have changed. Changes were derived from public comments during the April and May 2008 public comment periods, analysis of the March 31, 2008, Agency FDCC report and subject matter experts. FDCC Major Version 1.0 settings are available at http://www.antd.gov/fdcc/download_fdcc.cfm.

Federal Desktop Core Configuration Major Version 1.0

FDCC Major Version 1.0 is based on Microsoft Windows XP Service Pack (SP) 2 and Microsoft Windows Vista SP 1. Although Security Content Automation Protocol (SCAP) Content has been engineered so that it will also operate on Windows XP SP1, near-term Windows XP patch checking will be oriented toward Windows XP SP2. It is understood that many managed environments throughout the Federal government implement service packs shortly after their release. While near-term Windows XP patching is based on Windows XPSP2, we do not anticipate any significant misalignment issues for Windows XPSp1. NIST is currently working with IT product vendors to develop additional SCAP Content based on the FDCC settings for other platforms and applications.

To coincide with the release of FCEDC Major Version 1.1, new SCAP Content has also been made available. This SCAP Content is inclusive of the 40 FDCC settings changes. At this time, the FCEDC is comprised of settings located at http://fedio.fedgov that can be checked using the updated SCAP Content and SCAP-validated tools with FDCC Scanning capability as specified on the NIST website at http://www.nist.gov/scapproducts.cfm. Not all FDCC settings can be checked using automated scanning tools. NIST is coordinating the refinement of SCAP Content.
Other Automation Protocols Can Capture the Government Use Cases…

- **Enterprise System Information Protocol (ESIP)**
  - For reporting of asset inventory information. Common Platform Enumeration (CPE), etc.

- **Threat Analysis Automation Protocol (TAAP)**
  - For reporting and sharing structured threat information. Malware Attribute Enumeration & Characterization (MAEC), Common Attack Pattern Enumeration & Classification (CAPEC), Common Platform Enumeration (CPE), Common Weakness Enumeration (CWE), Open Vulnerability and Assessment Language (OVAL), Common Configuration Enumeration (CCE), and Common Vulnerabilities and Exposures (CVE).

- **Event Management Automation Protocol (EMAP)**
  - For reporting of security events. Common Event Expression (CEE), Malware Attribute Enumeration & Characterization (MAEC), and Common Attack Pattern Enumeration & Classification (CAPEC).
Other Automation Protocols Can Capture the Government Use Cases… (concluded)

- **Incident Tracking and Assessment Protocol (ITAP)**
  - For tracking, reporting, managing and sharing incident information. Open Vulnerability and Assessment Language (OVAL), Common Platform Enumeration (CPE), Common Configuration Enumeration (CCE), Common Vulnerabilities and Exposures (CVE), Common Vulnerability Scoring System (CVSS), Malware Attribute Enumeration & Characterization (MAEC), Common Attack Pattern Enumeration & Classification (CAPEC), Common Weakness Enumeration (CWE), Common Event Expression (CEE), Incident Object Description Exchange Format (IODEF), National Information Exchange Model (NIEM), and Cybersecurity Information Exchange Format (CYBEX).

- **Enterprise Remediation Automation Protocol (ERAP)**
  - For automated remediation of mis-configuration & missing patches. Common Remediation Enumeration (CRE), Extended Remediation Information (ERI), Open Vulnerability and Assessment Language (OVAL), Common Platform Enumeration (CPE), and Common Configuration Enumeration (CCE).

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

Asset Definition
Configuration Guidance
Vulnerability Alert
Threat Alert
Incident Report

Asset Inventory
Configuration Guidance Analysis
Vulnerability Analysis
Intrusion Detection
Incident Management

Operations Security Management Processes

System & Software Assurance Guidance/Requirements
Assessment of System Development, Integration, & Sustainment Activities and Certification & Accreditation

Knowledge Repositories

ESIP
SCAP
TAAP
EMAP
ITAP

SwAAP

Development & Sustainment Security Management Processes

Trust Management
Enterprise IT Change Management
Identity Management
Centralized Reporting

Operational Enterprise Networks

© 2010 The MITRE Corporation. All rights reserved.