



Supporting *Secure Software Operations*

Robert A. Martin
Sean Barnum

May 2011

UNCLASSIFIED

MITRE

© 2010 The MITRE Corporation. All rights reserved.

Report Documentation Page				Form Approved OMB No. 0704-0188	
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	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
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:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 32	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			



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

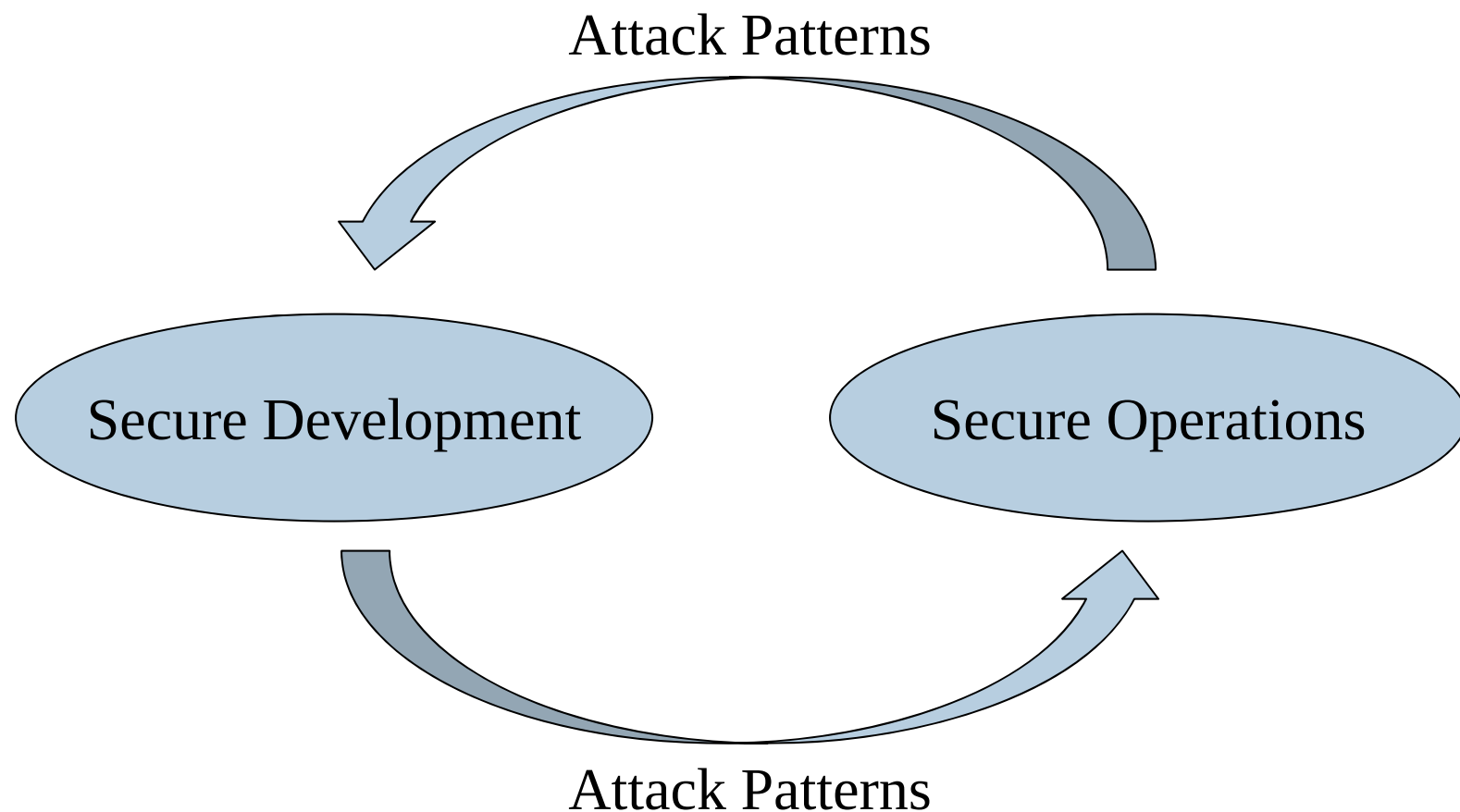


Bridge the secure development and secure operations domains

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 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 pattern abstractions to the low-level operational world?

Cyber Observables

The Secret Sauce for Bridging the Abstract to the Concrete



Cyber Observables Overview

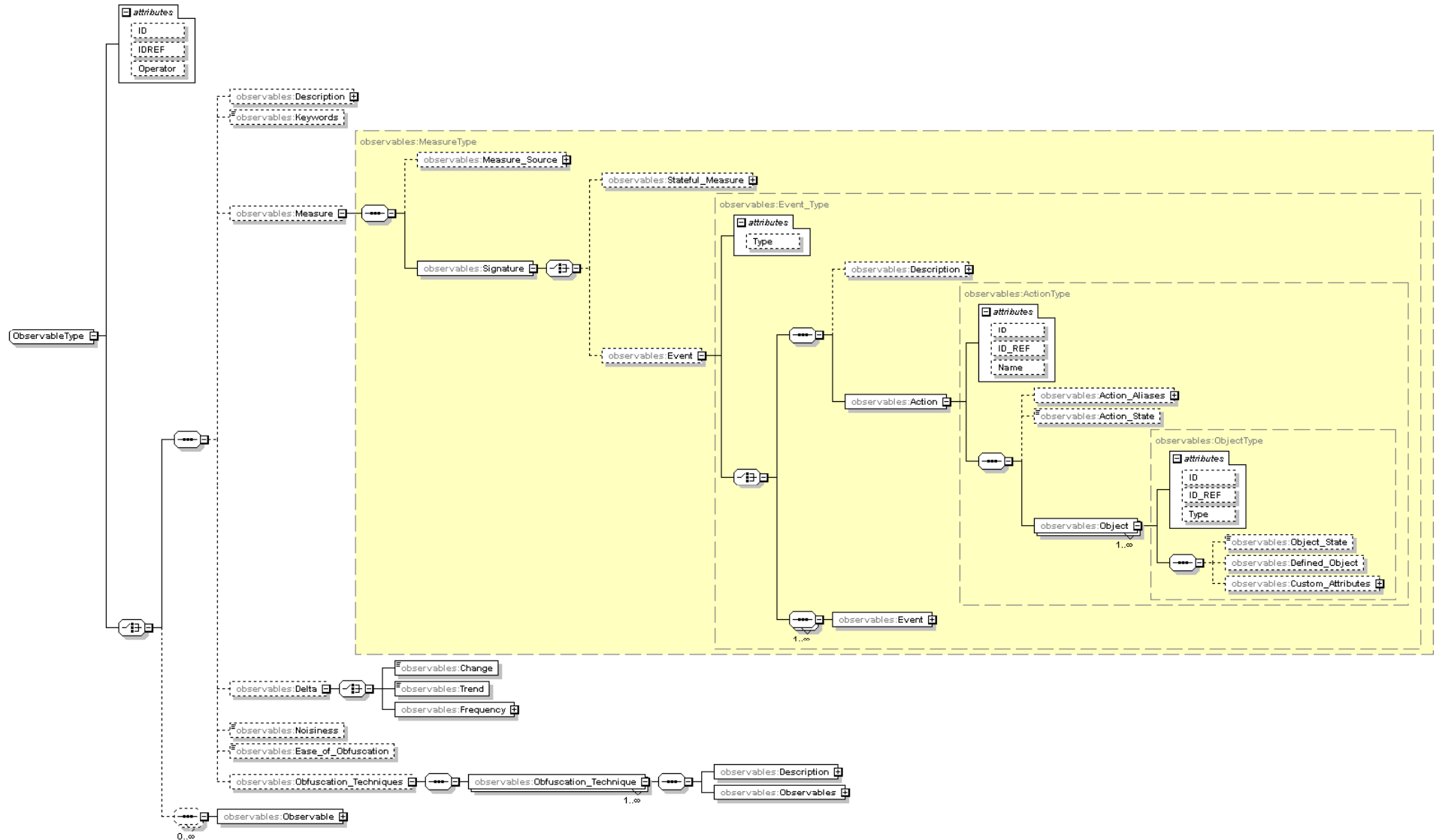
- The Cyber Observables construct is intended to capture and characterize events or properties that are observable in the operational domain.
- These observable events or properties can be used to adorn the appropriate portions of the attack patterns in order to tie the logical pattern constructs to 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



Generated by XMLSpy

www.altova.com



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....

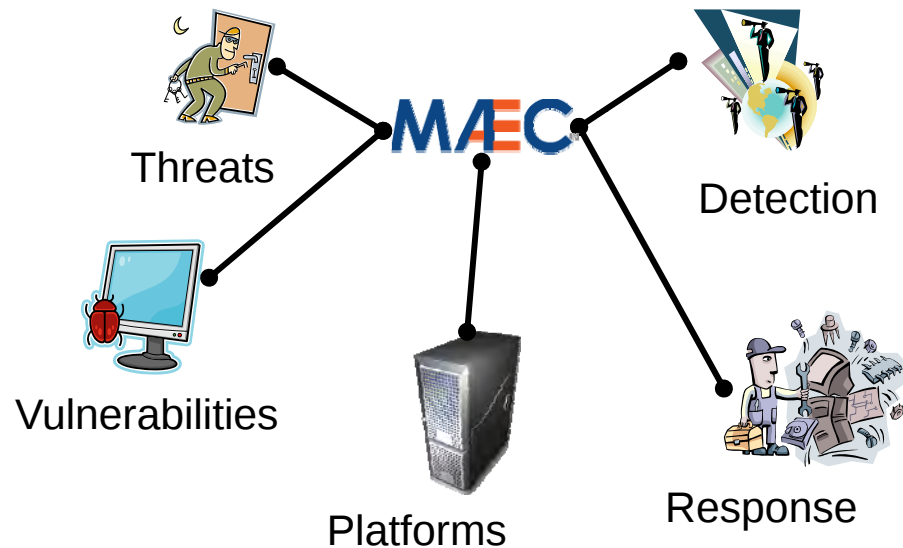


**Improve the analysis, characterization,
collection, discovery & knowledge
sharing of malware**

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



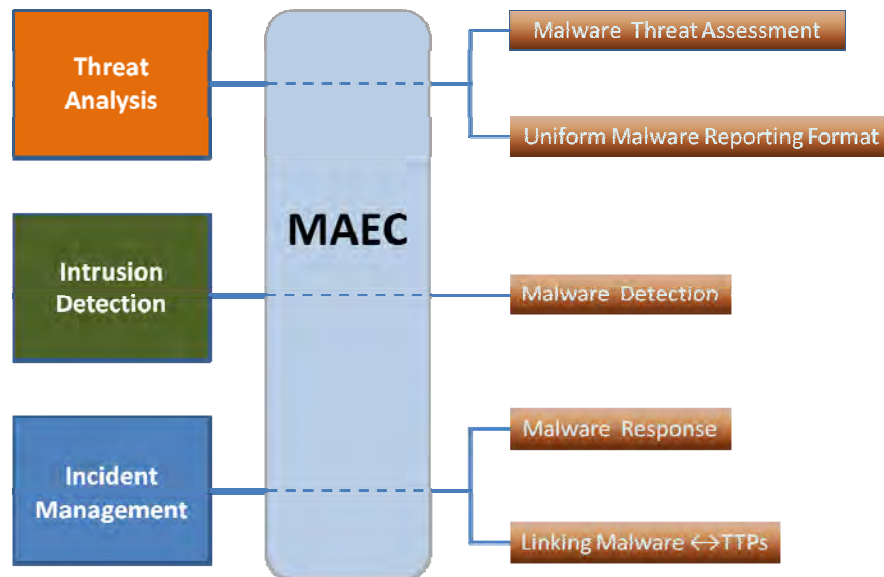
Malware Attribute Enumeration and Characterization (MAEC)



- Language for sharing structured information about malware
 - Grammar (Schema)
 - Vocabulary (Enumerations)
 - Collection Format (Bundle)
- Focus on attributes and behaviors
- Enable correlation, integration, and automation

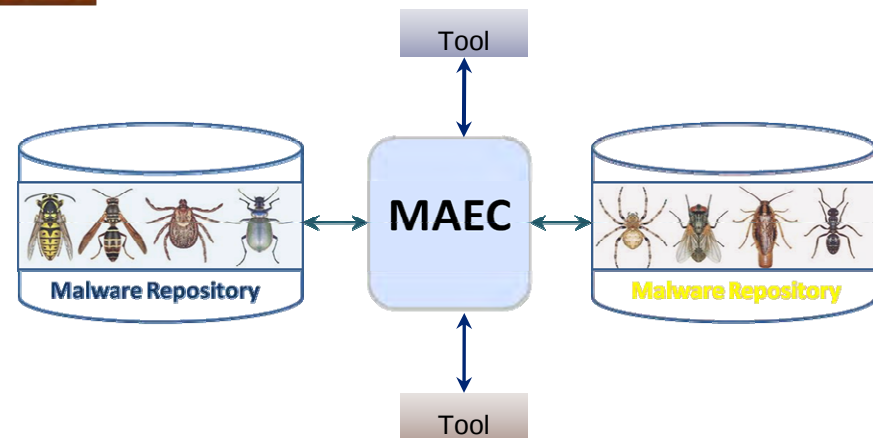
MAEC Use Cases

■ Operational

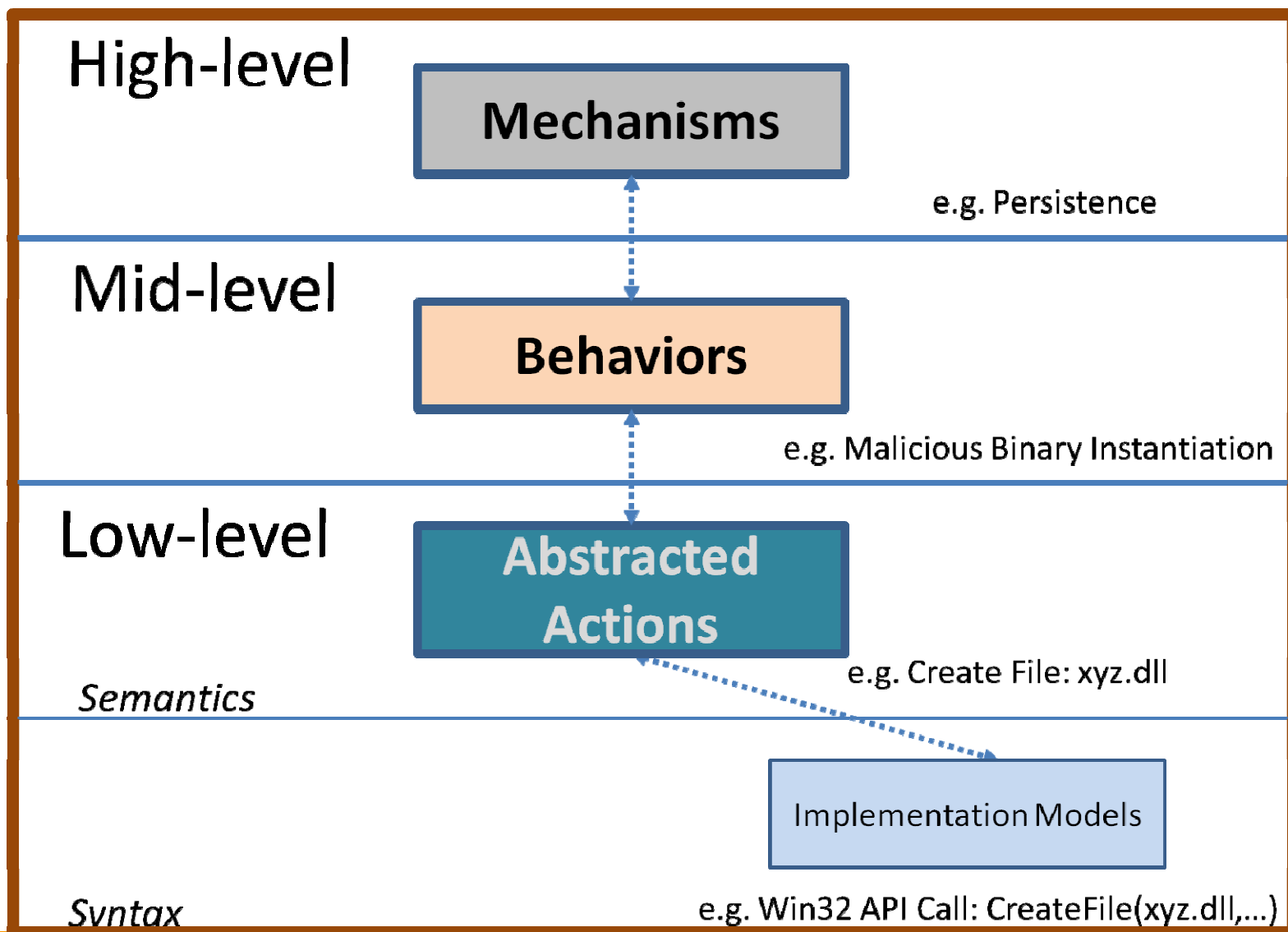


■ Analysis

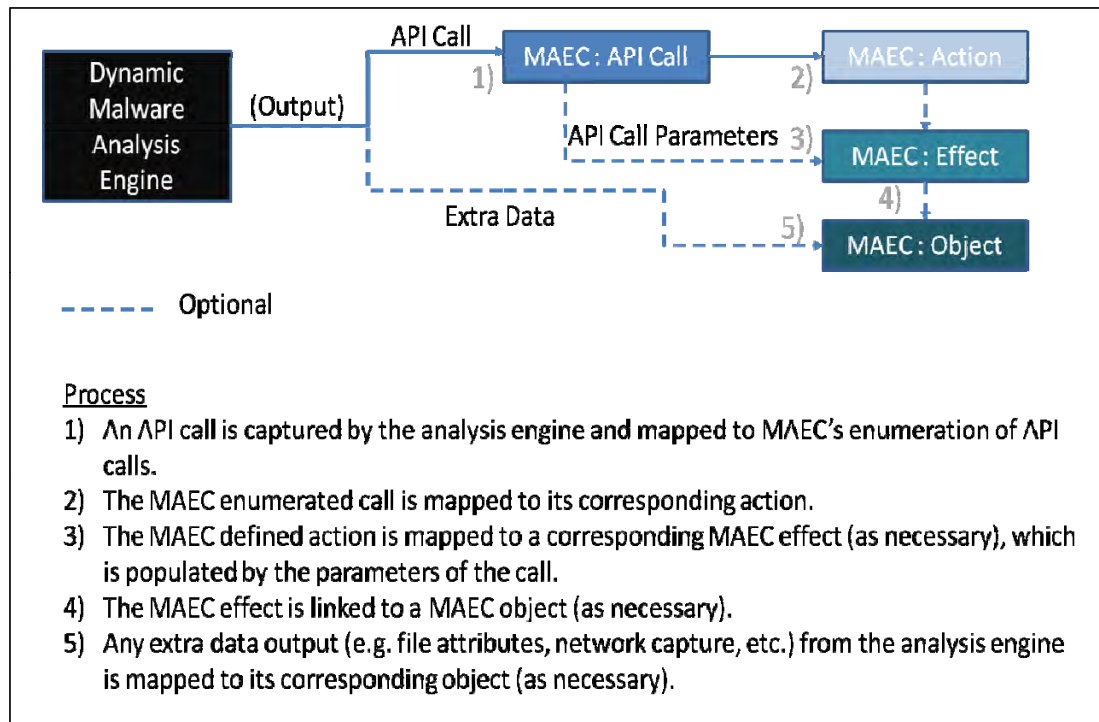
- Help Guide Analysis Process
- Standardized Tool Output
- Malware Repositories



MAEC Overview



Dynamic Malware Analysis → MAEC



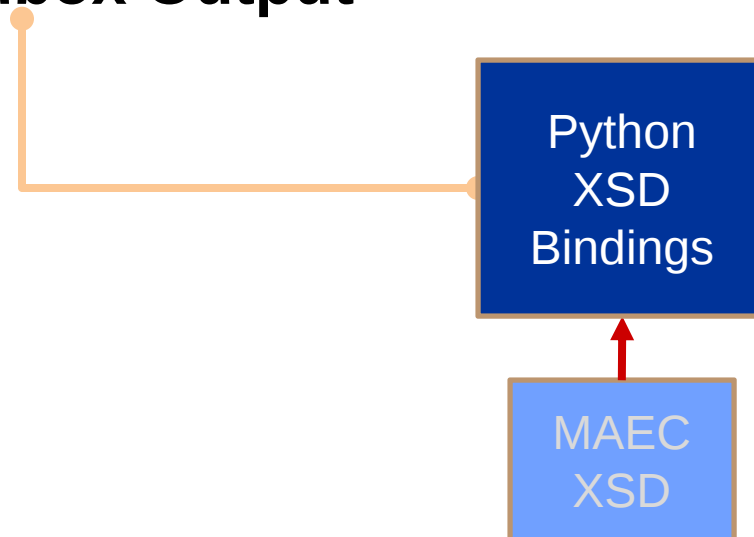
- Demonstrate the ability to generate MAEC XML descriptions from dynamic analysis tools
- Developed proof-of-concept translators for:
 - CW Sandbox (Sunbelt)
 - ASAT (MITRE)
 - Anubis
 - ThreatExpert

Test Case: CWSandbox Output -> MAEC

```
PID:1080,TID:1812,Caller:$00400000("KB823988.exe"),BEFORE,typFileSystem."FindFirstFile"
PID:1080,TID:1812,Caller:$00400000("KB823988.exe"),BEFORE,typFileSystem."SetFileAttrib"
PID:1080,TID:1812,Caller:$00400000("KB823988.exe"),BEFORE,typFileSystem."DeleteFileW"
PID:1080,TID:1812,Caller:$77A80000("CRYPT32.dll"),AFTER,typRegistry."RegOpenKeyExA"
PID:1080,TID:1812,Caller:$77A80000("CRYPT32.dll"),AFTER,typRegistry."RegEnumKeyA"
PID:1080,TID:1812,Caller:$77A80000("CRYPT32.dll"),AFTER,typRegistry."RegOpenKeyExA"
PID:1080,TID:1812,Caller:$77A80000("CRYPT32.dll"),AFTER,typRegistry."RegOpenKeyExA"
PID:1080,TID:1812,Caller:$77A80000("CRYPT32.dll"),AFTER,typRegistry."RegEnumKeyA"
PID:1080,TID:1812,Caller:$77A80000("CRYPT32.dll"),AFTER,typRegistry."RegOpenKeyExA"
PID:1080,TID:1812,Caller:$77A80000("CRYPT32.dll"),AFTER,typRegistry."RegEnumValueW"
PID:1080,TID:1812,Caller:$77A80000("CRYPT32.dll"),AFTER,typRegistry."RegOpenKeyExA"
PID:1080,TID:1812,Caller:$77A80000("CRYPT32.dll"),AFTER,typRegistry."RegOpenKeyExA"
PID:1080,TID:1812,Caller:$77A80000("CRYPT32.dll"),AFTER,typRegistry."RegOpenKeyExW"
PID:1080,TID:1812,Caller:$77A80000("CRYPT32.dll"),AFTER,typRegistry."RegOpenKeyExW"
PID:1080,TID:1812,Caller:$77A80000("CRYPT32.dll"),AFTER,typRegistry."RegCreateKeyExW"
```

```
<Action Successful="true" id="10" Action_Type="copy" Name="copy_file">
  <Description/>
  <Action_Initiator type="Process">
    <Initiator_Name>KB823988.exe</Initiator_Name>
    <Process_ID>1080</Process_ID>
    <Thread_ID>1812</Thread_ID>
  </Action_Initiator>
  <Action_Implementation>
    <API_Call>
      <Name>CopyFileW</Name>
      <API_Call_Parameter ordinal_position="1">
        <Name>filetype</Name>
        <Value>file</Value>
      </API_Call_Parameter>
      <API_Call_Parameter ordinal_position="2">
        <Name>srcfile</Name>
        <Value>c:\\KB823988.exe</Value>
      </API_Call_Parameter>
      <API_Call_Parameter ordinal_position="3">
        <Name>dstfile</Name>
        <Value>C:\\WINDOWS\\system32\\ntos.exe</Value>
      </API_Call_Parameter>
      <API_Call_Parameter ordinal_position="4">
        <Name>creationdistribution</Name>
        <Value>CREATE_ALWAYS</Value>
      </API_Call_Parameter>
      <API_Call_Parameter ordinal_position="5">
        <Name>desiredaccess</Name>
        <Value>FILE_ANY_ACCESS</Value>
      </API_Call_Parameter>
      <API_Call_Parameter ordinal_position="6">
        <Name>flags</Name>
        <Value>SECURITY_ANONYMOUS</Value>
      </API_Call_Parameter>
    </API_Call>
  </Action_Implementation>
</Action>
```

Raw CWSandbox Output



MAEC XML

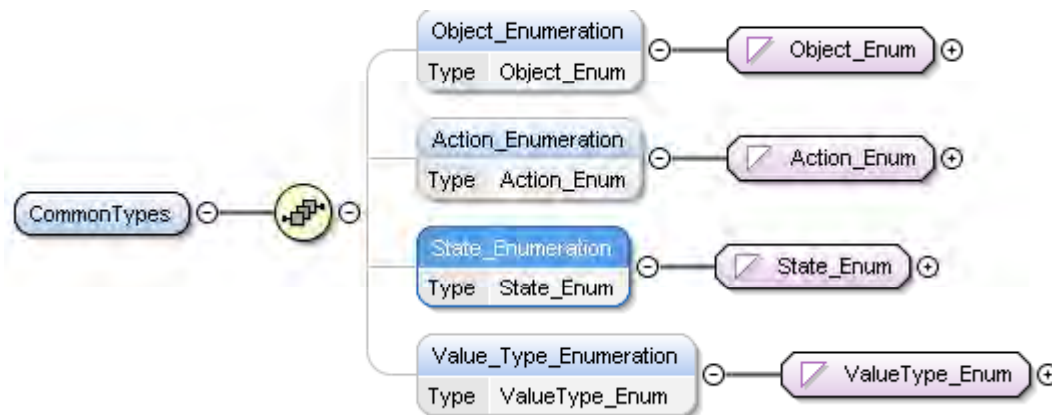
- MAEC Actions
- MAEC Objects
- MAEC Behaviors

Collaboration



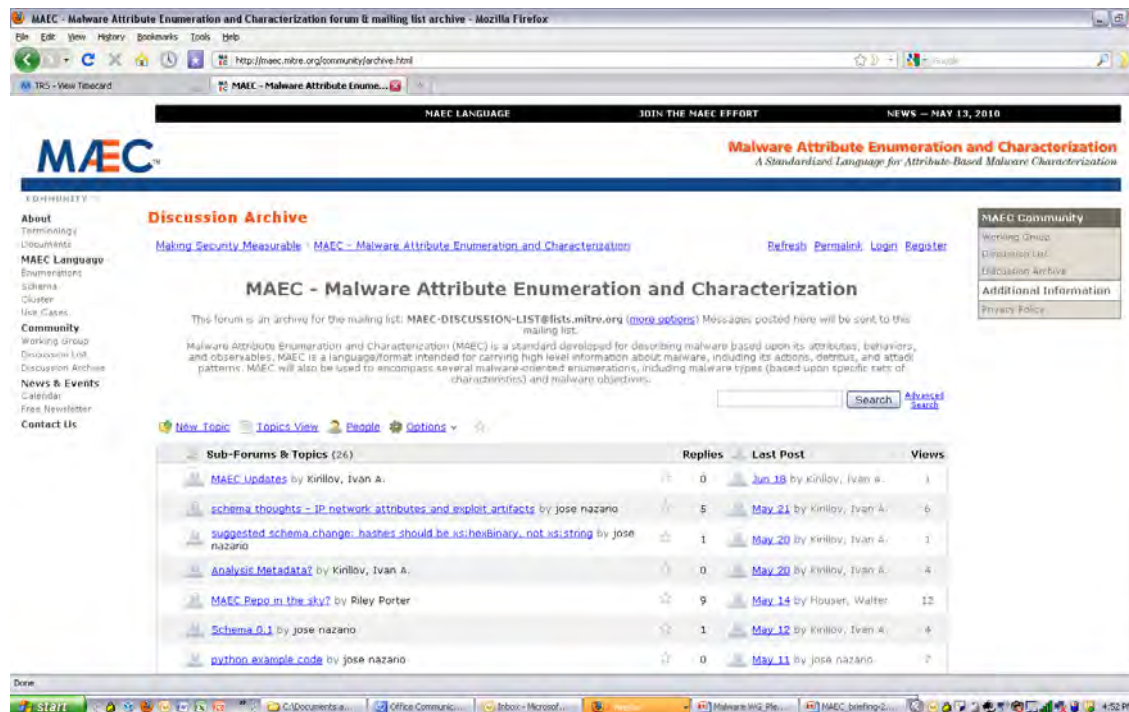
■ Related Making Security Measurable Efforts

- 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>
- Archives available



MAEC Community: MAEC Development Group on Handshake



- MITRE hosts a social networking collaboration environment:
<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**

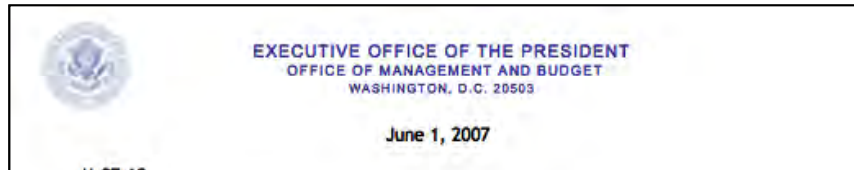
Remembering the Acronyms

What IT systems do I have in my enterprise?	• CPE (Platforms)
What vulnerabilities do I need to worry about?	• CVE (Vulnerabilities)
What vulnerabilities do I need to worry about RIGHT NOW?	• CVSS (Scoring System)
How can I configure my systems more securely?	• CCE (Configurations)
How do I define a policy of secure configurations?	• XCCDF (Configuration Checklists)
How can I be sure my systems conform to policy?	• OVAL (Assessment Language)
How can I be sure the operation of my systems conforms to policy?	• OCIL (Interactive Language)
What weaknesses in my software could be exploited?	• CWE (Weaknesses)
What attacks can exploit which weaknesses?	• CAPEC (Attack Patterns)
What should be logged, and how?	• CEE (Events)
How can I aggregate assessment results?	• ARF (Results)
How can we recognize malware?	• MAEC (Malware Attributes)

Standardization Efforts leveraged by the Security Content Automation Protocol (SCAP)

What IT systems do I have in my enterprise?	• CPE (Platforms)
What vulnerabilities do I need to worry about?	• CVE (Vulnerabilities)
What vulnerabilities do I need to worry about RIGHT NOW?	• CVSS (Scoring System)
How can I configure my systems more securely?	• CCE (Configurations)
How do I define a policy of secure configurations?	• XCCDF (Configuration Checklists)
How can I be sure my systems conform to policy?	• OVAL (Assessment Language)
How can I be sure the operation of my systems conforms to policy?	• OCIL (Interactive Language)
What weaknesses in my software could be exploited?	• CWE (Weaknesses)
What attacks can exploit which weaknesses?	• CAPEC (Attack Patterns)
What should be logged, and how?	• CEE (Events)
How can I aggregate assessment results?	• ARF (Results)
How can we recognize malware?	• MAEC (Malware Attributes)

SCAP – FDCC and USGCB



M-07-18

MEMORANDUM FOR

FROM:

SUBJECT:

The Office of Management and Budget (OMB) is implementing of Commonly Accepted Security Configurations for Windows Operating Systems, which states and/or plans to upgrade configurations by February 2008.

This memorandum solicitation to ensure information technology configurations. Your agency is requested to provide information on the following:

- The provided information and operate information technology Windows XP settings, see: <http://csa.mitre.org>
- The standard software configuration Service for information technology silently installed
- Applications without elevated privileges

MITRE



EXECUTIVE OFFICE OF THE PRESIDENT
OFFICE OF MANAGEMENT AND BUDGET
WASHINGTON, D.C. 20503

August 11, 2008

M-08-22

MEMORANDUM FOR THE CHIEF INFORMATION OFFICERS

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

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

In March 2007, OMB Memorandum M-07-11 announced the "Implementation of Commonly Accepted Security Configurations for Windows Operating Systems," directing agencies with Windows XPTM deployed and/or plan to upgrade to the VistaTM 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 comment during the April and May 2008 public comment periods, analysis of the March 31, 2008, Agency FDCC reports and subject matter expertise. FDCC Major Version 1.0 settings are available at http://nvd.nist.gov/fdcc/download_fdcc.cfm.

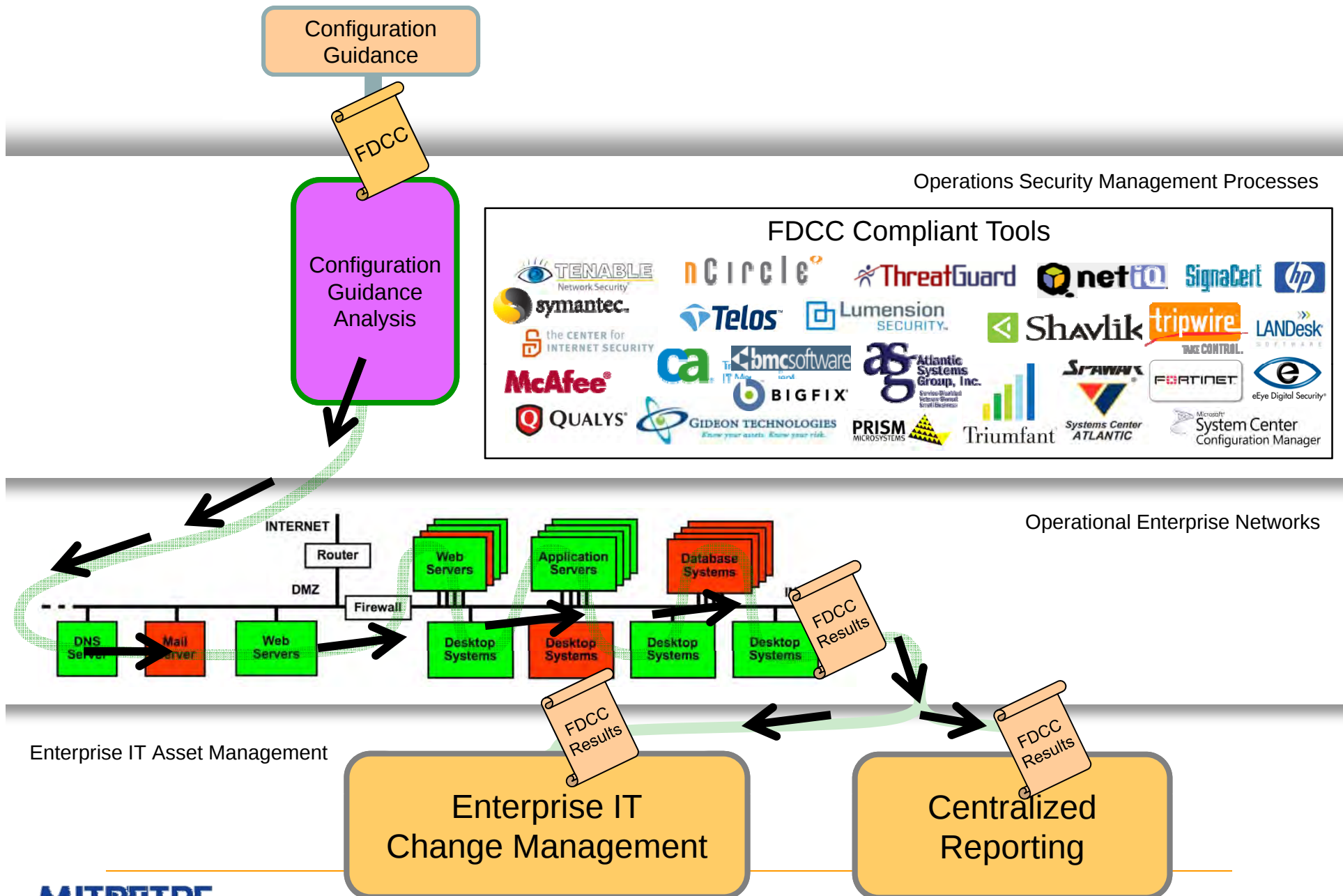
Federal Desktop Core Configuration Major Version 1.0

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 SP3, 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 checking is based on Windows XP/SP2, we do not anticipate any significant measurement issues for Windows XP/SP3. NIST is currently working with IT product vendors to develop additional SCAP Content based on the FDCC settings for other platforms and applications.

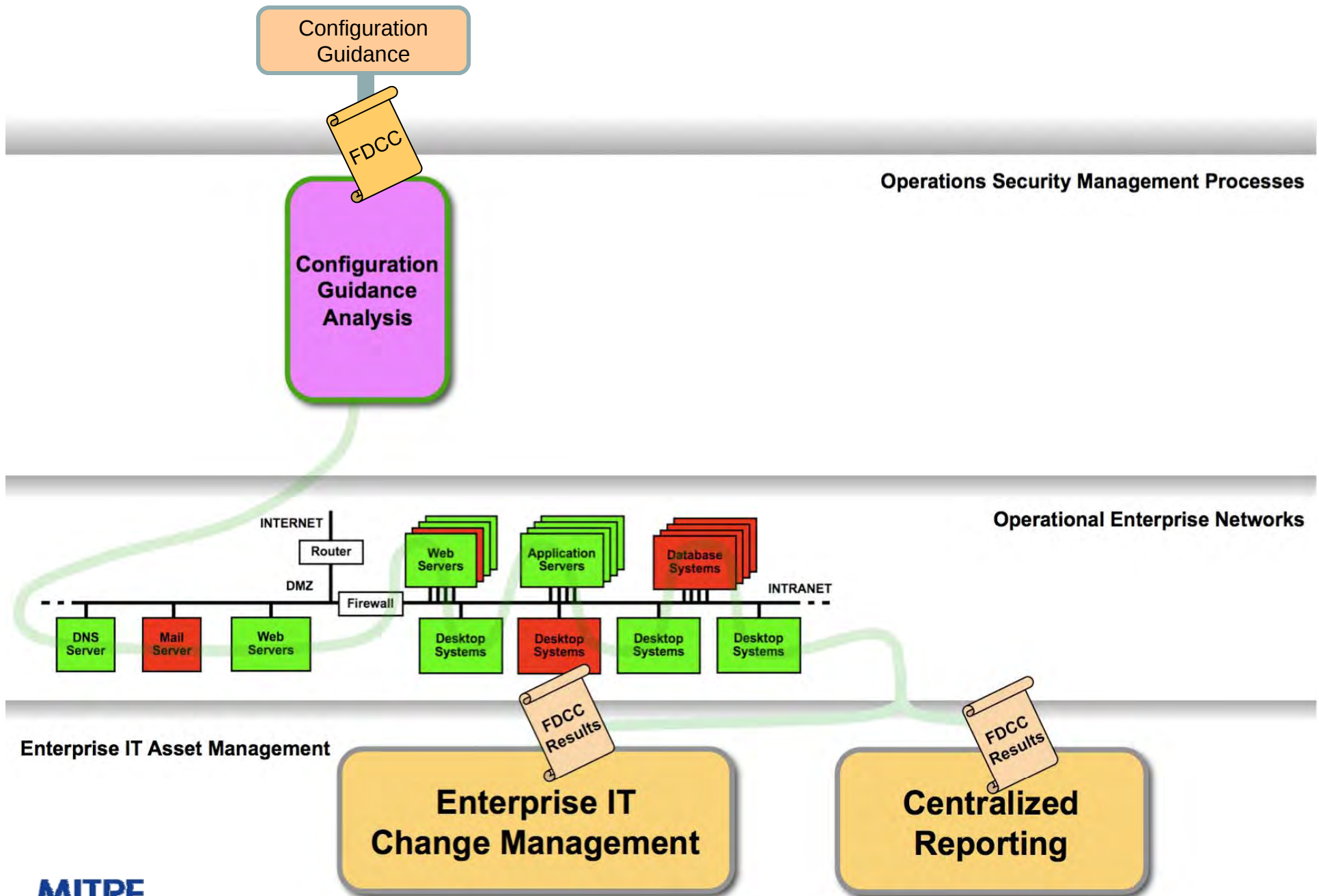
To coincide with the release of FDCC Major Version 1.0, new SCAP Content has also been made available. This SCAP Content is inclusive of the 40 FDCC settings changes. At this time, the FDCC is comprised of settings located at <http://fdcc.nist.gov> that can be checked using the updated SCAP Content and SCAP-validated tools with FDCC Scanning capability as specified on the NIST website at <http://nvd.nist.gov/scapproducts.cfm>. Not all FDCC settings can be checked using automated scanning tools. NIST is coordinating the refinement of SCAP Content

UNCLASSIFIED

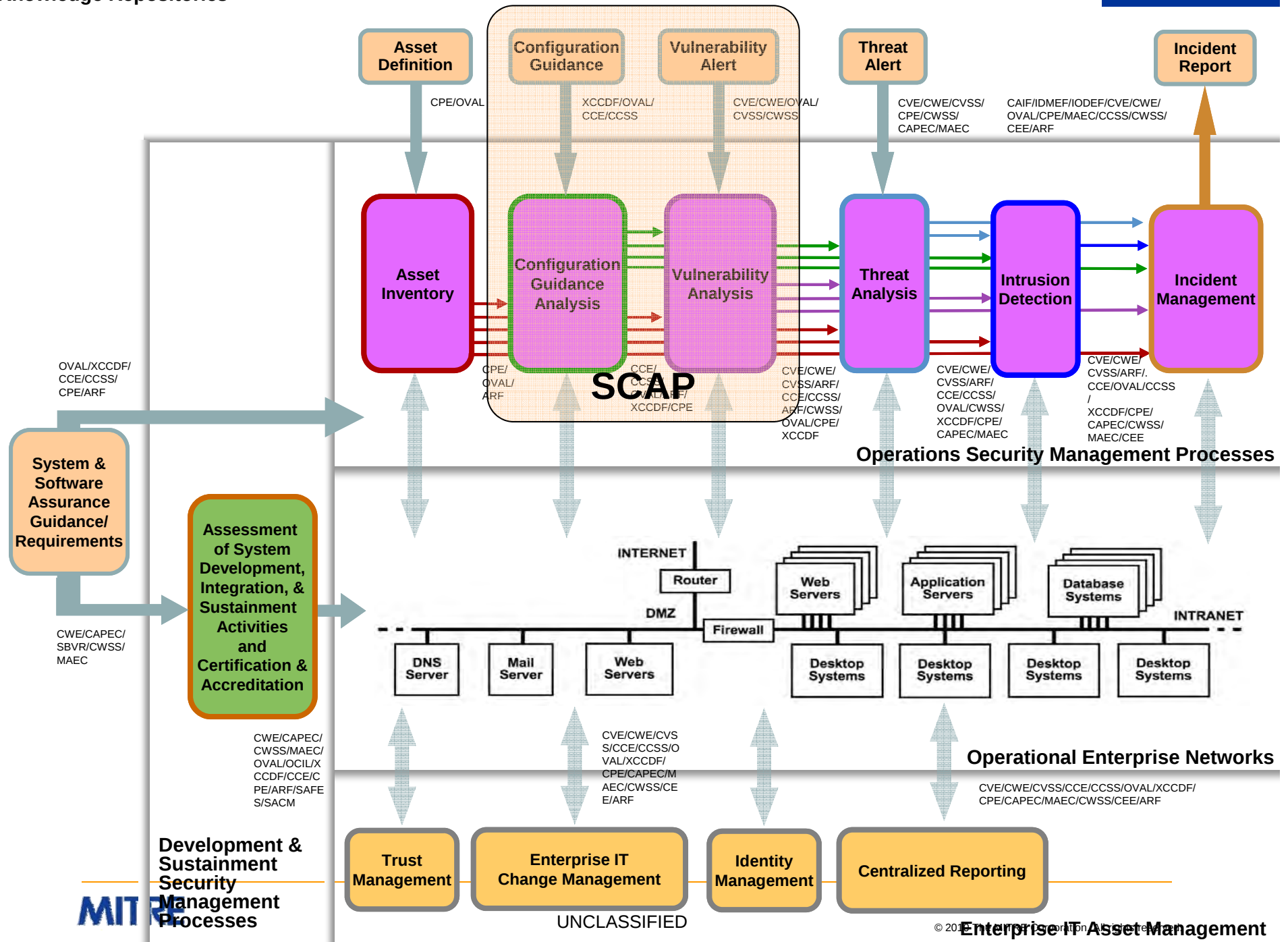
SCAP-Based FDCC Guidance



SCAP-Based FDCC Reporting



Knowledge Repositories



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).

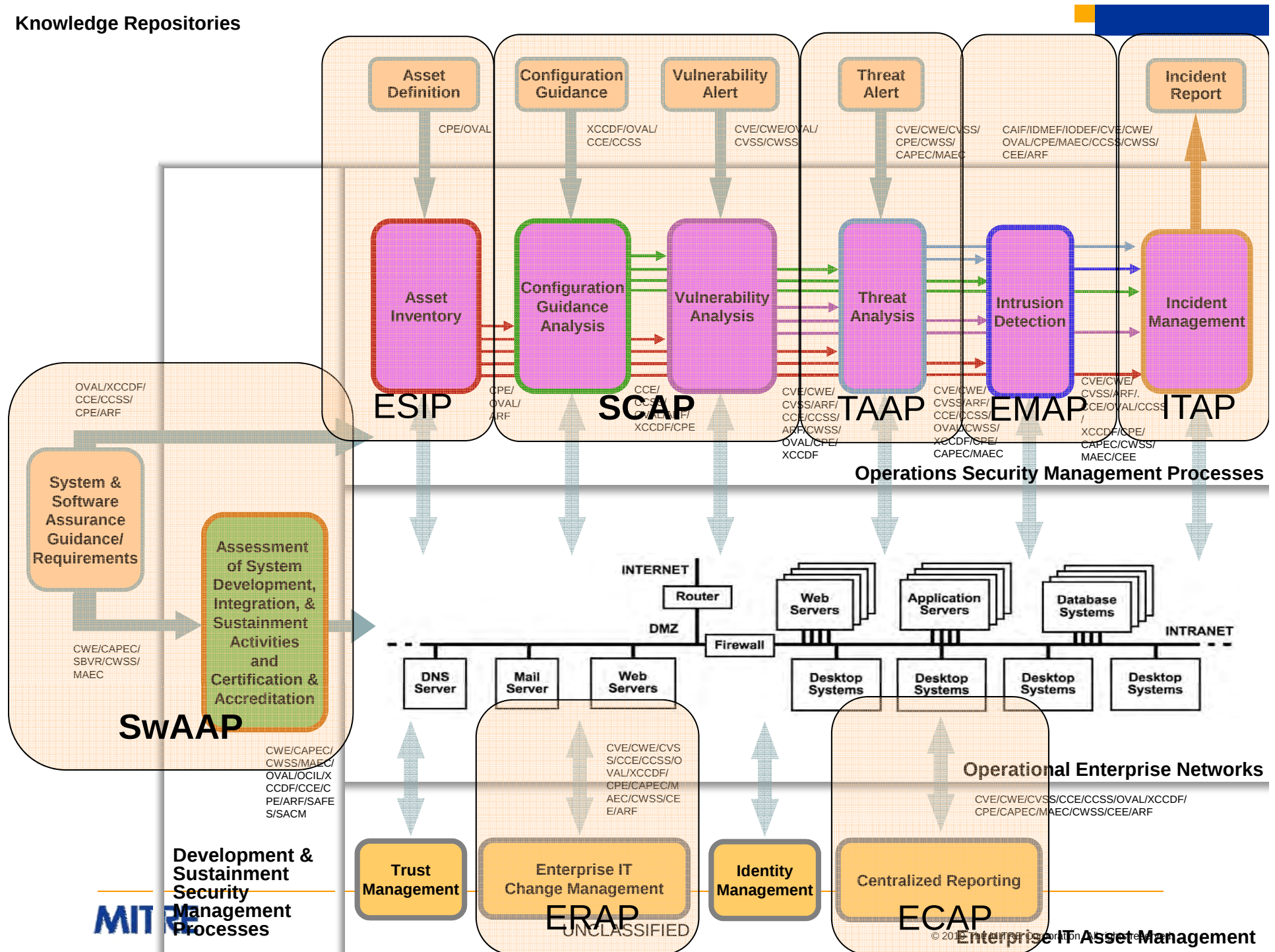
■ Enterprise Remediation Automation Protocol (ERAP)

- 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



[makingsecuritymeasurable.mitre.org]

Making Security Measurable

A Collection of Information Security Community Standardization Activities and Initiatives

[Home](#) | [About](#) | [Current Collection](#) | [Incubator](#) | [Events & Participation](#) | [Feedback Requested](#)

MITRE, in collaboration with government, industry, and academic stakeholders, is improving the measurability of security through enumerating baseline security data, providing standardized languages as means for accurately communicating the information, and encouraging the sharing of the information with users by developing repositories.

The other activities and initiatives listed here have similar concepts or compatible approaches to MITRE's. Together all of these efforts are helping to make security more measurable by defining the concepts that need to be measured, providing for high fidelity communications about the measurements, and providing for sharing of the measurements and the definitions of what to measure.

Measurable security pertains at a minimum to the following areas:

Vulnerability Management

Intrusion Detection

Asset Security Assessment

Asset Management

Configuration Guidance

Patch Management

Malware Response

Incident Management

Threat Analysis

Enumerations

Languages

Repositories

[Common Vulnerabilities and Exposures \(CVE®\)](#) - common vulnerability identifiers

[Common Weakness Enumeration \(CWE™\)](#) - list of software weakness types

[Common Attack Pattern Enumeration and Classification \(CAPEC™\)](#) - list of common attack patterns

[Common Configuration Enumeration \(CCE™\)](#) - common security configuration identifiers

[Common Platform Enumeration \(CPE™\)](#) - common platform identifiers

[CWE/SANS Top 25](#) - consensus list of the 25 most dangerous programming errors

[Center for Internet Security \(CIS\) Consensus Security Metrics Definitions](#) - set of standard metrics and data definitions that can be used across organizations to collect and analyze data on security process performance and outcomes

[Twenty Most Important Controls and Metrics for Effective Cyber Defense and Continuous FISMA Compliance](#) - twenty key actions or security "controls" that organizations must take to block or mitigate known and reasonably expected attacks

[SANS Top Twenty](#) - SANS/FBI consensus list of the Twenty Most Critical Internet Security Vulnerabilities that uses CVE-IDs to identify the issues

[OWASP Top Ten](#) - ten most critical Web application security flaws

[WASC Web Security Threat Classification](#) - list of Web security threats

[Open Vulnerability and Assessment Language \(OVAL®\)](#) - standard for determining vulnerability and configuration issues

[Common Event Expression \(CEE™\)](#) - standardizes the way computer events are described, logged, and exchanged

[Malware Attribute Enumeration and Characterization \(MAEC™\)](#) - standardized language for attribute-based malware characterization

[Benchmark Development](#) - resources for creating standards-based, structured, and automatable security guidance

[OVAL Interpreter](#) - free tool for collecting information for testing, carrying out OVAL Definitions, and presenting results of the tests

[Benchmark Editor™](#) - free tool that enhances and simplifies creation and editing of benchmark documents written in XCCDF and OVAL

[Recommendation Tracker™](#) - free tool that facilitates the development of automated security benchmarks

[Extensible Configuration Checklist Description Format \(XCCDF\)](#) - specification language for uniform expression of security checklists, benchmarks, and other configuration guidance

[Open Checklist Interactive Language \(OCIL\)](#) - standardized language for expressing and evaluating non-automated security checks

[Common Vulnerability Scoring System \(CVSS\)](#) - open standard that conveys vulnerability severity and helps determine urgency and priority of response

[Policy Language for Assessment Results Reporting \(PLARR\)](#) - language for requesting IT asset assessment results from tools, databases, and other products

[Assessment Results Format \(ARF\)](#) - open language for exchanging per-device assessment results data between assessment tools, asset databases, and other products that manage asset information

[Assessment Summary Results \(ASR\)](#) - language for exchanging summarized assessment results data

[OVAL Repository](#) - community-developed OVAL Vulnerability, Compliance, Inventory, and Patch Definitions

[National Vulnerability Database \(NVD\)](#) - U.S. vulnerability database based on CVE that integrates all publicly available vulnerability resources and references

[NIST Security Content Automation Protocol \(SCAP\)](#) - security content for automating technical control compliance activities, vulnerability checking, and security measurement

[Red Hat Repository](#) - OVAL Patch Definitions corresponding to Red Hat Errata security advisories

[Novell Repository](#) - OVAL Definitions for SUSE Linux Enterprise compliance checking

[Debian Repository](#) - OVAL Definitions corresponding to Debian security advisories

[National Checklist Program Repository](#) - U.S. government repository of publicly available security checklists/benchmarks

[Center for Internet Security \(CIS\) Benchmarks](#) - best-practice security configurations accepted for compliance with FISMA, the ISO standard, GLB, SOx, HIPAA, and FIRPA, and other regulatory requirements for information security

[DISA Security Technical Implementation Guides \(STIGs\)](#) - U.S. Defense Information Systems Agency's (DISA) STIGs are configuration standards for DOD information assurance and information assurance-enabled devices and systems

[Common Frameworks for Vulnerability Disclosure and Response \(CVRF\)](#) - standard format for reporting and sharing vulnerability information among multiple organizations

[Federal Desktop Core Configuration \(FDCC\)](#) - OMB-mandated security configuration for Microsoft Windows Vista and XP operating system software

[United States Government Configuration Baseline \(USGCB\)](#) - security configuration baselines for IT products deployed across federal agencies

View the current collection of organizations, activities, and initiatives.

[Disclaimer](#)

This Web site is hosted by The MITRE Corporation. © 2010 The MITRE Corporation. CVE and OVAL are registered trademarks and the Making Security Measurable logo, CCE, CWE, CPE, CAPEC, CEE, MAEC, Benchmark Editor, and Recommendation Tracker are trademarks of The MITRE Corporation. All other trademarks are the property of their respective owners. Contact us: measurablesecurity@mitre.org

Page Last Updated: September 02, 2010

Making Security Measurable

MITRE

UNCLASSIFIED

© 2010 The MITRE Corporation. All rights reserved.