Technical Introduction to SCAP

Charles Schmidt

The MITRE Corp.

MITRE

Approved for Public Release; Distribution Unlimited: 10-1786

What is SCAP?

- Security Content Automation Protocol
  - SCAP provides a standardized approach to maintaining the security of enterprise systems, such as…
    - automatically verifying the presence of patches
    - checking system security configuration settings
    - examining systems for signs of compromise
- Defined by NIST IR 800-117
- First formed in 2006
  - First validation requirements published in 2009

Approved for Public Release; Distribution Unlimited: 10-1786

What is SCAP, really?

- A super-standard
  - Comprised of 6 (going on 7) individually maintained standards
- Guides the use of several security automation standards
  - We have a standard that identifies platforms…
  - We have a standard to encapsulate guidance…
  - …But how do we know what guidance applies to what platform?
  - Answer: use SCAP
    - SCAP links security automation standards
- Today SCAP provides…
  - Guidance on use of these component standards
  - Procedures to validate compliance with this guidance

Why Standards?

- Everyone standards, but why are they useful here?
- Common understanding of “what”
  - “Are we talking about the same software vulnerability?”
  - “Do we agree on what a policy recommendation means and how to meet it?”
  - These are really hard questions without standards
- Common baseline of capabilities
  - Content authors know what to expect of tools
- Universal content
  - Content authors don’t need to write for each assessment tool
  - Establish a shared content repository everyone can use
    - And which all people will use with a consistent understanding
- Tool compatibility/Plug-n-play/Vendor Neutrality
  - Still working on this, but standards can support this too

Approved for Public Release; Distribution Unlimited: 10-1786

What Defines SCAP

- NIST SP800-117: Adopting and Using Security Content Automation Protocol
  - How to use SCAP in one's enterprise and how to create tools that fit into an SCAP-compatible architecture
- NIST SP800-126: Security Content Automation Protocol Specification
  - Technical overview of SCAP
- NIST IR-7511: SCAP Version 1.0 Validation Program Test Requirements
  - Detailed technical requirements for tools that wish to be validated as SCAP compliant

Who Influences SCAP?

- NIST
- Other Government Organizations
  - NSA and DHS have been the primary funders of this work
  - Other agencies, including DOE, are becoming more involved
- Vendors, Researchers, and Users
  - Microsoft, Red Hat, Sun, IBM, Cisco, McAfee, Symantec, SANS Institute, MITRE, and many, many others
- You
  - Mailing lists are open to anyone and we listen to all comments

Approved for Public Release; Distribution Unlimited: 10-1786

Important Terms

- Enumerations
  - Dictionaries used to provide a common identifiers for items
  - Not a database – entries provide just enough information to clearly describe the instances of the given item
    - Additional information could then be compiled in a separate database using the identifier as a key
- Languages
  - Interpreted by people/software to guide activities (in our case, security assessment)
  - Provide structure and organization of what would otherwise be narrative content
    - Helps to standardize and promote compatibility
- Metrics
  - Algorithm that helps users rank importance of items

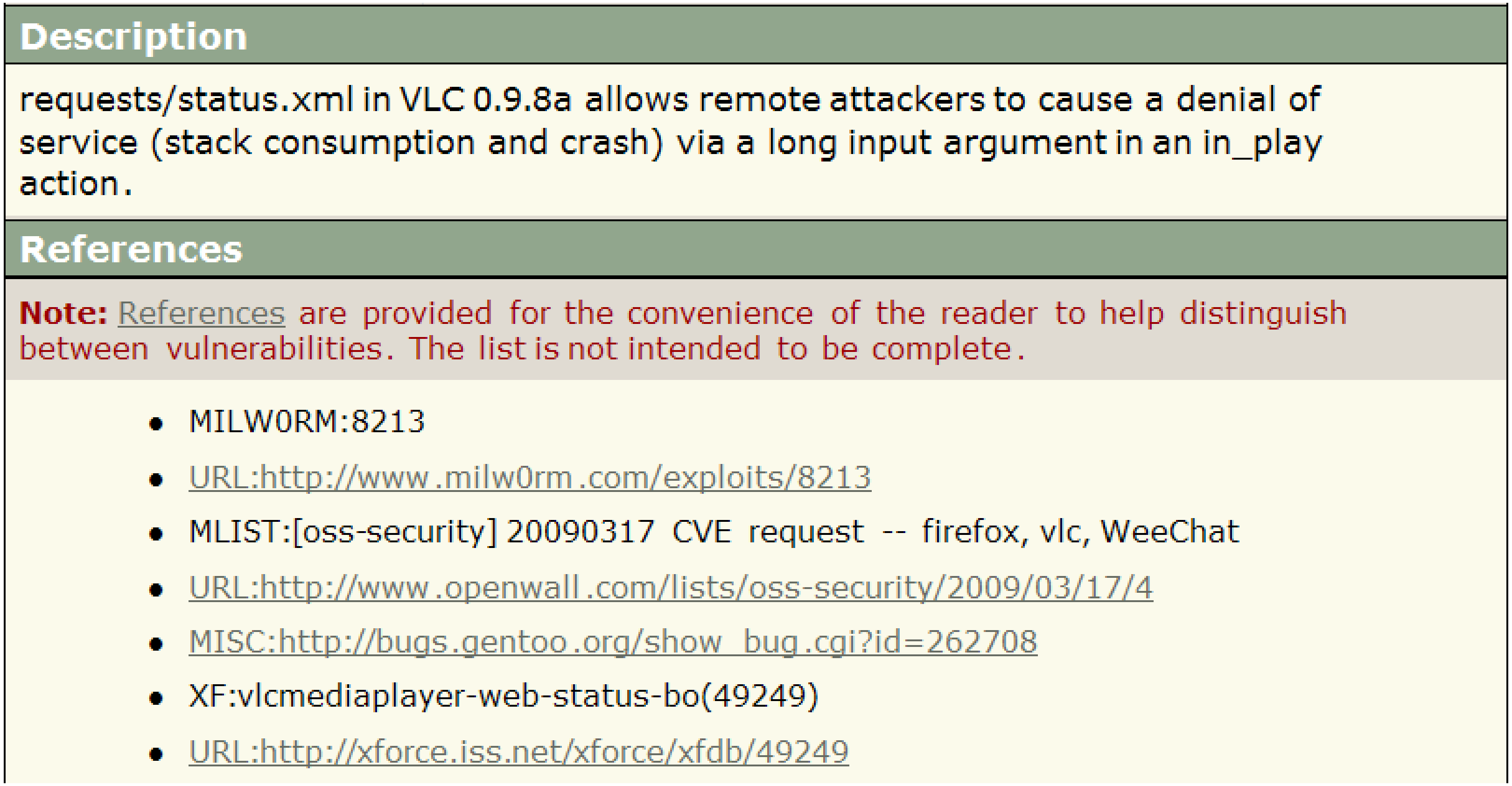
Approved for Public Release; Distribution Unlimited: 10-1786

The Components of SCAP

- CVE (Common Vulnerabilities and Exposures)
  - Enumeration of software vulnerabilities
- CCE (Common Configuration Enumeration)
  - Enumeration of configurable controls of software
- CPE (Common Platform Enumeration)
  - Enumeration of identities of software/hardware entities
- CVSS (Common Vulnerability Scoring System)
  - Metric used to assign a severity score to vulnerabilities entries
- XCCDF (eXtensible Configuration Checklist Description Format)
  - Language for encapsulating structure and content of security guidance
- OVAL (Open Vulnerability and Assessment Language)
  - Language to describe tests against system state
- OCIL (Open Checklist Interactive Language)
  - Language for user questionnaires (coming in SCAP 1.1)

CVE Enumeration

- Entries are given an identifier: CVE-*year-number*
  - CVE-2009-1045



Description

requests/status.xml in VLC 0.9.8a allows remote attackers to cause a denial of service (stack consumption and crash) via a long input argument in an in_play action.

References

Note: References are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- MILW0RM:8213
- URL:http://www.milw0rm.com/exploits/8213
- MLIST:[oss-security] 20090317 CVE request -- firefox, vlc, WeeChat
- URL:http://www.openwall.com/lists/oss-security/2009/03/17/4
- MISC:http://bugs.gentoo.org/show_bug.cgi?id=262708
- XF:vlcmediaplayer-web-status-bo(49249)
- URL:http://xforce.iss.net/xforce/xfdb/49249

From http://cve.mitre.org

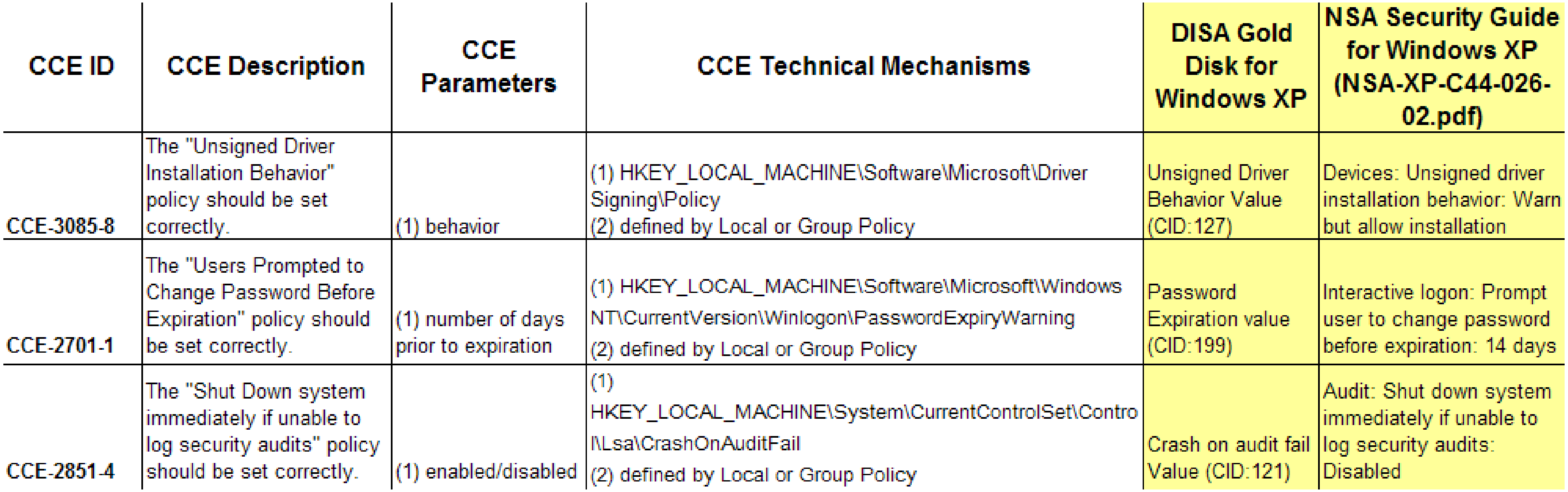
CVE

- Used for
  - Correlating vulnerability information
    - Between advisories, scan results, patch coverage, scanner capabilities, etc.

- Widely used (almost 100 software products make direct use of CVE)
- Many vendors and security researchers now publish CVE names in their bulletins
- More than 41,000 entries (with about 100 added every week)
- National Vulnerability Database (NVD, http://nvd.nist.gov/) annotates CVE entries with additional information

CCE Enumeration

- Entries are given an identifier – CCE-*number-checksum*
  - CCE-3291-2



CCE ID	CCE Description	CCE Parameters	CCE Technical Mechanisms	DISA Gold Disk for Windows XP	NSA Security Guide for Windows XP (NSA-XP-C44-026-02.pdf)
CCE-3085-8	The "Unsigned Driver Installation Behavior" policy should be set correctly.	(1) behavior	(1) HKEY_LOCAL_MACHINE\Software\Microsoft\Driver Signing\Policy (2) defined by Local or Group Policy	Unsigned Driver Behavior Value (CID:127)	Devices: Unsigned driver installation behavior: Warn but allow installation
CCE-2701-1	The "Users Prompted to Change Password Before Expiration" policy should be set correctly.	(1) number of days prior to expiration	(1) HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\PasswordExpiryWarning (2) defined by Local or Group Policy	Password Expiration value (CID:199)	Interactive logon: Prompt user to change password before expiration: 14 days
CCE-2851-4	The "Shut Down system immediately if unable to log security audits" policy should be set correctly.	(1) enabled/disabled	(1) HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\CrashOnAuditFail (2) defined by Local or Group Policy	Crash on audit fail Value (CID:121)	Audit: Shut down system immediately if unable to log security audits: Disabled

- CCEs do not contain recommendations – policy neutral
- CCEs do not map to just one way of controlling a configuration – procedurally neutral
- CCE is *NOT* platform neutral – each piece of software has its own list of CCEs

Approved for Public Release; Distribution Unlimited: 10-1786

CCE

- Used for
  - Convey universal understanding of what a policy configures
  - Track recommendations against specific configuration requirements of other policies
  - Ensure policy comparisons are between equivalent recommendations

- More than 5000 entries (focusing on controls that appear in major security guides)
- Some vendors now publishing guides with CCE-identified controls

Approved for Public Release; Distribution Unlimited: 10-1786

CPE Enumeration

- CPE names are composed of a descriptive URI
  - cpe:/{part}:{vendor}:{product}:{version}:{update}:{edition}:{language}
  - Part is “o” for Operating System, “a” for Application, or “h” for Hardware
  - Empty blocks cover all possible values (e.g. all versions or all editions)
- Examples:
  - cpe:/o:microsoft:windows_xp::sp1
    - Microsoft windows xp_sp1 (all versions, editions, and languages)
  - cpe:/a:ibm:tivoli_configuration_manager:4.2
    - IBM Tivoli Configuration Manager 4.2 (all updates, editions & languages)

Approved for Public Release; Distribution Unlimited: 10-1786

CPE

- Used for
  - Automated software inventories
  - Mapping platforms to vulnerability or policy statements

- Over 20,000 official CPE names
- All NVD entries are annotated with CPE information

Approved for Public Release; Distribution Unlimited: 10-1786

CVSS Algorithm

- Scores a given vulnerability based on its likely danger
  - Score runs between 0 (no danger) and 10 (extreme danger)
- Three parts
  - Base – the inherent danger of the vulnerability
    - A provider can fill this out ahead of time
  - Temporal – changes over time
    - Depends of maturity of exploits and remediations
  - Environmental – reflects specific dangers to an enterprise
    - Depends on how critical the threatened component is and the impact of failure

Approved for Public Release; Distribution Unlimited: 10-1786

CVSS

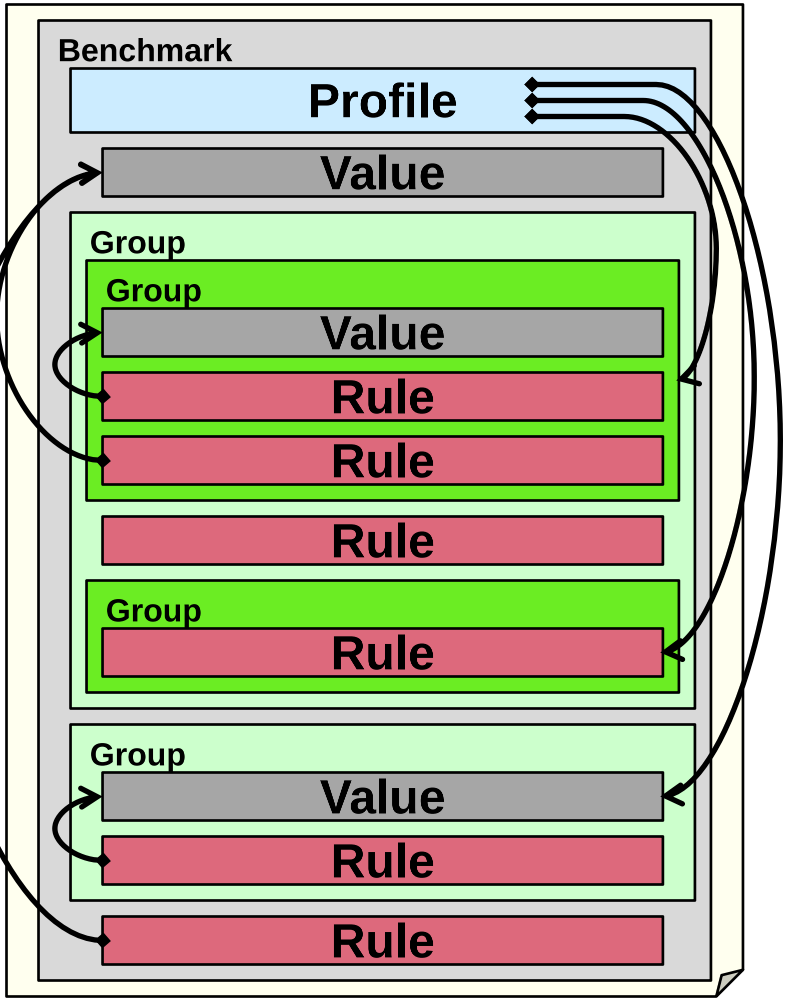
- Used for
  - Prioritizing responses to published vulnerabilities
  - Weighing the cost of mission-impacting remediations against allowing a vulnerability to persist

- All NVD vulnerabilities are annotated with CVSS information
- Many government agencies and corporations use CVSS in their vulnerability management strategies

Approved for Public Release; Distribution Unlimited: 10-1786

XCCDF Language

- Encapsulates guidance information such as security policies



- Rules – Recommendations
- Values – Variables
  - Rules reference Values
- Groups – Structuring
- Profiles – Tailoring
  - Profiles reference Rules, Groups, and Values
  - Rules & Groups can be enabled or disabled
  - Values can have their value adjusted

Approved for Public Release; Distribution Unlimited: 10-1786

XCCDF

- Used for
  - Encapsulation of security policy recommendations
  - Annotating of ad-hoc checking mandates
  - Driving of automated assessments

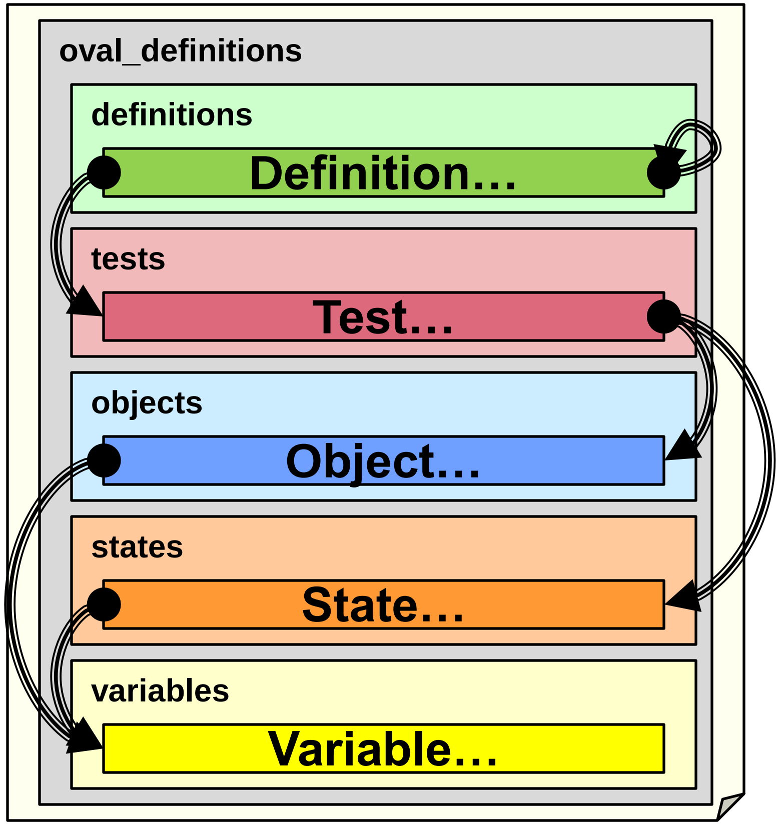
- National Checklist Program contains almost two dozen security guides written in XCCDF
- Documents can be converted to human-readable output and/or be processed by tools to automate assessment
- Many XCCDF-compatible tools are currently on the market
- Configurable design simplifies tailoring existing content to meet local mission needs

Approved for Public Release; Distribution Unlimited: 10-1786

OVAL Language

- Describes how to locate and test system state information

- Definition – top-level structure of a check
- Test – link to “locators” and “evaluators”
- Object – locate entities
  - Each type of entity has its own Object type
- State – evaluate entities
  - Each type of entity has its own State type
- Variable



Approved for Public Release; Distribution Unlimited: 10-1786

OVAL

- Used for
  - Precise expression of …
    - what it means to be (non)compliant with a policy recommendation
    - vulnerability presence
    - inventory measures
    - patch detection
  - Driving of automated system scans

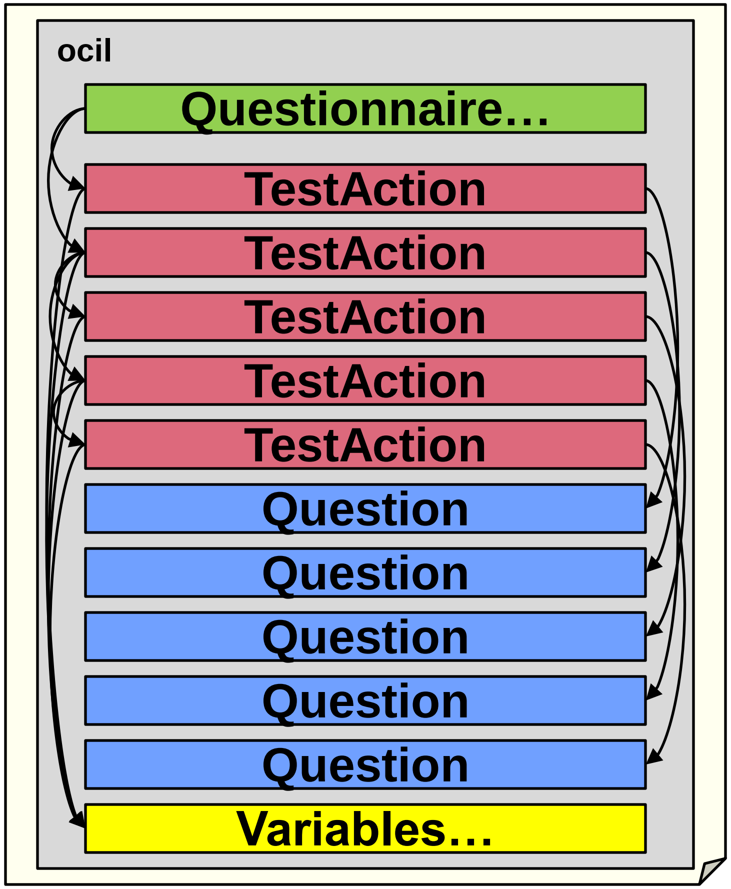
- The public OVAL repository contains over 7000 definitions
- OVAL now published with RedHat advisories
- Community-contributed content is often available shortly after alerts are published

Approved for Public Release; Distribution Unlimited: 10-1786

OCIL Language

- Describes chains of questions to pose to a user

- Questionnaire – top-level structure
- TestAction – Matches questions to follow-on actions
- Question – The question and optionally a list of responses
- Variables



Approved for Public Release; Distribution Unlimited: 10-1786

OCIL

- Used for
  - Queries regarding non-technical policy recommendations
  - Manual collection of artifacts that provide evidence of security posture

- OCIL will be part of SCAP 1.1, released in January 2011

Approved for Public Release; Distribution Unlimited: 10-1786

What Resources Does SCAP Have?

- National Vulnerability Database
  - Vulnerability Search Engine
    - Annotated CVE entries include CVSS scores and vectors, CPEs, and other information
  - National Checklist Program Repository
    - Guidance for many applications and operating systems
    - Many guides use SCAP – usable by SCAP compatible tools
    - Includes STIG, FDCC, USGCB, and vendor benchmarks
  - CPE dictionary
    - All official CPE names for platforms
- Component standard sites
  - OVAL – OVAL repository with over 7000 definitions
  - CCE – The official list of CCE entries
  - Documentation, use cases, and other information on all sites
- Mailing lists and archives

Approved for Public Release; Distribution Unlimited: 10-1786

General Use Cases

- Security Configuration Verification and Description
  - XCCDF, OVAL, and OCIL can describe policy checks
    - Consistent and universal understanding of the recommendations
  - CCE identifies the controls affected by policy
  - CPE identifies the platforms affected by policy
- Vulnerability Measurement and Identification
  - CVE provides a universal name for vulnerabilities
  - OVAL can detect the presence of vulnerabilities and the installation of specific patches
  - CVSS helps prioritize remediation actions
  - CPE identifies the platforms affected by vulnerabilities
- Inventory Naming and Automation
  - OVAL can detect application installation
  - CPE provides a universal name for installed applications

SCAP Applied Use Cases (1)

- Policy Authors - Create organizational policy
  - Create normative configuration guidance
  - Identify appropriate (and inappropriate) inventory elements

- Benefits
  - Benefit from a body of modular, extensible base content
  - Ensure universal, consistent understanding of requirements
  - Measurements returned with common format - supports analysis

Approved for Public Release; Distribution Unlimited: 10-1786

SCAP Applied Use Cases (2)

- Incident Responders - Craft responses to specific threats
  - Receive vulnerability information and track fixes
  - Craft configuration changes to policy to deal with threats
  - Track susceptible inventory

- Benefits
  - Solid correlations between alerts, evidence, and responses
  - Guidance on prioritizing responses by magnitude of threat
  - OVAL content is often publicly available shortly after alerts
  - Precise understanding of what software, version, edition, etc. is present
  - Measurements returned with common format - supports analysis

Approved for Public Release; Distribution Unlimited: 10-1786

SCAP Applied Use Cases (3)

- Administrators - Configure and assess end systems
  - Update and verify that systems meet configuration guidance requirements
  - Update and verify that system are not vulnerable to known threats
  - Track enterprise inventory and correlate with the above

- Benefits
  - Receive exact understandings of what is required
  - Does not require detailed read of instructions – can focus on areas of special concern and let automation handle the rest
  - Recommendations can be tailored to meet enterprise mission
  - Automation reduces time demands and increases accuracy
  - Content usable by many tools
  - Measurements returned with common format - supports analysis

Approved for Public Release; Distribution Unlimited: 10-1786

Looking around

- Remediation standards
- Software Assurance Standards
  - Common Weakness Enumeration (CWE) – Encyclopedia of software weakness types
  - Common Attack Pattern Enumeration and Classification (CAPEC) – Encyclopedia of general attack methods
  - Malware Attribute Enumeration and Classification (MAEC) – Standardized descriptors of malware
- Event Management Standards
  - Common Event Expression (CEE) – Standard log language
  - Log manipulation languages
  - Enumeration of events
  - Scoring system for events
- Assessment Control Standards
  - Standardize invocation and control of assessment actions

Approved for Public Release; Distribution Unlimited: 10-1786

For More Information…

- More information on the standards
  - CVE – Vulnerabilities; http://cve.mitre.org
  - CCE – Configuration controls; http://cce.mitre.org
  - CPE – Platforms/applications; http://cpe.mitre.org
  - OVAL – Checking language; http://oval.mitre.org
  - OCIL – Questionnaire language; http://scap.nist.gov/specifications/ocil
  - XCCDF – Structuring; http://nvd.nist.gov/xccdf.cfm
  - CVSS – Scores severity of vulnerabilities; http://www.first.org/cvss/
  - NVD – Resources for SCAP users; http://nvd.nist.gov/home.cfm
  - Making Security Measureable – More resources on SCAP and beyond; http://measurablesecurity.mitre.org/
- MITRE provides free training on benchmark development
  - See our web site for more information: http://benchmarkdevelopment.mitre.org/