The Evolving Use Cases of XCCDF

Charles Schmidt - Moderator

June 10, 2009

MITRE

Session Objectives

- Review existing use cases
  - Do the use cases themselves reflect current practice/need?
  - Are current features sufficient to support these cases?
- Discussion of proposed new/revised use cases
  - Do proposed use cases represent a real community need?
  - What features needed to support these cases?
  - Is XCCDF where they belong?

Current Use Case Requirements

1) Creation of security guidance checklists by subject matter experts
2) Tailoring by auditors/system administrators
   a) Include structure and text to guide tailoring steps
3) Generation of human-readable documentation
4) Support of translation to HTML
5) Support of translation to other XML formats
6) Facilitate the normalization of configuration content through automated security tools
   a) Creation of normalized scan results
7) Encapsulate remediation information
8) Support vulnerability alerts by encapsulating descriptions and detection procedures

Taken from pages 6 & 7 of NISTIR 7275r3 – "Specification for the Extensible Configuration Checklist Description Format (XCCDF) Version 1.1.4"

Use Cases

- Simple content creation – full guidance and ad-hoc
  - Inheritance
  - Groups to provide structure
- Tailoring
  - Profiles, Rule/Group selected, Value selectors, Value references (in checks and text)
  - Rule/Group questions
- Document generation
  - Description, rationale, Rule/Group hidden
- Checking
  - Check, complex-check, check-export
  - Scoring
- Reporting
  - TestResult
- Remediation
  - Check/fixtext

Tailoring

- Not-in-place Tailoring
  - Creating tailored versions of XCCDF documents require modification of the document
    - New profile
    - Manual selection/de-selection/Value modification
  - Users have noted advantages of external tailoring structures
    - Preservation of document signatures
    - Tailoring potentially preserved after an update of the source
- Automatic tailoring
  - Currently, CPE selection provides some automatic tailoring
  - Do we want automated Profile selection?
  - Do we want more explicit automatic tailoring (e.g. Rule sets that allow selection of Profiles/Groups/Rules)
    - Instead/in addition to CPE

Remediation

- Currently, remediation content is provided in a mixed content "fixtext" field
- A new language, OVRL (Open Vulnerability Remediation Language), is under development to provide canonical remediation information
- Request have been made to make XCCDF better support this new standards-based approach to remediation

Checker Control

- XCCDF provides input to a checking tool, but does not control that tool's actions
  - Tools may execute Rules in any order
  - Tools may arbitrarily select from multiple checking mechanisms in a single Rule
  - Tools may determine whether complex checks are complete or short-circuit
- Do we want to use XCCDF as instructions (rather than input) to checking tools?

Checker Control – Possible Features

- Deterministic ordering of Rules and Groups
  - Dynamic selection – change selection values based on Rule results
- Support for chained tests
  - If check 1 = true, run check 2
- Periodicity instructions
  - "Results become stale after x hours/days"

Check Result to Rule Result

- **Currently mapping is by convention**

OVAL Definition Result		XCCDF Rule Result
Error		Error
Unknown		Unknown
Not applicable		Not applicable
Not evaluated		Not checked
		Not selected
		Informational
		Fixed
Definition Class	**Definition Result**	
Compliance	True	
Vulnerability	False	Pass
Inventory	True	
Patch	False	
Definition Class	**Definition Result**	
Compliance	False	
Vulnerability	True	Fail
Inventory	False	
Patch	True	

  - **OCIL - pass, fail, error, unknown, not checked, not applicable**
  - **OCRL → XCCDF Fixed?**
- **Proposal A – codify mappings once and for all**
- **Proposal B – allow XCCDF to explicitly map (no codification)**

Additional Value Capabilities

- Lists in Value objects
  - Currently, Values export a single value
  - Suggestion to allow lists to be exported
- Support for automatic Value population
  - Currently Values are read-only after tailoring
  - Suggestions have been made to populate Value values from checks
  - Check 1 collects data; used to select a new value for Value 1

Versioning

- Currently there is an optional version field in Items/ Benchmark/Profile
- Proposals
  - Versions mandatory
  - Version behavior dictated by spec
  - Additional version metadata