Toward an Ontology Architecture for Cyber-Security Standards

Mary C. Parmelee

The MITRE Corporation
7515 Colshire Drive,
McLean, VA 22102-7539, USA
mparmelee@mitre.org

Abstract. The rapid growth in magnitude and complexity of cyber-security information and event management (CSIEM) has ignited a trend toward security automation and information exchange standards. Making Security Measurable (MSM) references a collection of open community standards for the common enumeration, expression and reporting of cyber-security-related information. While MSM-related standards are valuable for enabling security automation; insufficient vocabulary management and data interoperability methods as well as domain complexity that exceeds current representation capabilities impedes the adoption of these important standards. This paper describes an Agile, ontology architecture-based approach for improving the ability to represent, manage, and implement MSM-related standards. Initial cross-standard analysis revealed enough common concepts to warrant four ontologies that are reusable across standards. This reuse will simplify standards-based data interoperability. Further, early prototyping enabled us to streamline vocabulary management processes and demonstrate the ability to represent complex domain semantics in OWL ontologies.

Keywords: cyber-security, ontology architecture, security standards, security automation, making security measurable, security information and event management, SIEM, semantic interoperability, Agile Development, OWL, RDF

Disclaimer. The views expressed in this chapter are those of the author's alone and do not reflect the official policy or position of The MITRE Corporation or any other company or individual.

1 Introduction

Through its Making Security Measurable [13] and related efforts to standardize the expression and reporting of cyber-security-related information, MITRE leads the development of several open community standards. These standards are primarily designed to support security automation and information interoperability, as well as facilitate human security analysis across much of the cyber-security information and event management (CSIEM) lifecycle. Some of the major security-related activities supported by the standards are: vulnerability management, intrusion detection, asset management, configuration guidance, incident management and threat analysis. MITRE’s support of the individual standards is funded by several federal government organizations. Many of the MSM-related standards have been adopted by the National Institute of Standards and Technology’s (NIST’s) Security Content Automation Protocol (SCAP) program [16]. Federal government organizations and security tool vendors are moving toward adoption of SCAP validated products to ensure baseline security data and tool interoperability [15].

While MSM-related standards are valuable for enabling security automation; insufficient vocabulary management and data interoperability methods as well as domain complexity that exceeds current representation capabilities impedes the adoption of these important standards. This paper describes an Agile Development [1], ontology architecture-based approach for improving the ability to represent, manage, and implement MSM-related standards. The Cyber-Security Ontology Architecture is a loosely-coupled, modular representation that is resilient to rapid change and complexity. Architecture-based services and applications are free to combine and extend architecture components at implementation time to fit application-specific contexts without having to implement a single monolithic model. The result is improved ability to support security automation, vocabulary management, and data interoperability. Initial cross-standard analysis revealed enough common concepts to warrant four ontologies that are reusable across standards. This reuse is one way that this approach will simplify standards-based data interoperability. Further, early prototyping enabled us to streamline vocabulary management processes and demonstrate the ability to represent complex domain semantics in OWL ontologies that are difficult or not possible to represent using the Relational Database (RDB) and XML Schema (XSD) [17, 30] technologies in which the standards are currently implemented.

2 Background

This section provides background descriptions of ontology architecture and controlled vocabulary in the context of this paper.

An ontology architecture is a conceptual information model comprised of a loosely-coupled federation of modular ontologies that form the structural and semantic framework of an information domain. Ontology architectures have been used to relate upper ontologies to their middle and domain level extensions [21]. Many of the concepts involved in ontology architecture are defined Ontology architectures are especially useful when applied to large, dynamic, complex domains such as cyber-security [17]. The major benefits of this federated approach to ontology application are [8, 23]:

1. Loose coupling and modularization makes it easier to add, remove and maintain individual ontologies;
2. Modular ontologies are easier to reuse and process than large monolithic ontologies;
3. Component ontologies can be dynamically combined on demand at implementation time to meet application-specific needs.

The vocabulary of complex, dynamic domains such as cyber-security often include atypical linguistic expressions such as acronyms, idioms, and numeric codes. It is important to recognize that although these linguistic expressions are not standard language terms, they form an accepted vocabulary in the context of the domain. This perspective of what constitutes a vocabulary calls for a broad definition of controlled vocabulary (CV). In this context, a controlled vocabulary is a collection of linguistic expressions that is vetted by an authority (e.g. a community) according to a set of criteria. All of the MSM standards maintain some form of a controlled vocabulary. These vocabularies were developed independently of each other, and are at various stages of maturity that range from a few months to ten years of active development.

3 Obstacles to Standards Adoption

The three major obstacles inhibiting the widespread adoption of the MSM-related standards are:

1. Unsustainable vocabulary management processes: Vocabulary management involves thousands of manually developed and managed value enumerations and vocabulary representations that are mostly encoded in XSD. The MSM-related standards are growing rapidly in number, volume and complexity. Some of the standards are adding hundreds to thousands of enumeration entries per month. A semantic approach to vocabulary management would streamline the vocabulary management process and reduce human error.

2. Ineffective data interoperability methods**:** Data interoperability activities are largely driven by the SCAP Validation program, which among other things, requires security tool vendors to translate proprietary output to a common expression and reporting form in order to achieve SCAP compliance [15]. This data interoperability is typically accomplished with manual ETL-style mappings to each of the SCAP-required standards. This mapping process would be more tractable, even semi-automatable if common concepts were represented more consistently across standards. A well-designed ontology architecture would facilitate this consistency.

3. Rapidly evolving, complex domain semantics that exceed the representation capability of the RDB and XSD technologies in which the standards are currently implemented: Domain complexity issues such as how to represent the behavioral

aspects of malware, and relating numerous software versioning schemes, call for a more semantic representation than either XSD or RDB technologies alone can readily provide. The semantics of these technologies are currently represented mostly in human interpretable documentation, which is not automatable or machine processable.

The following sections of this document describe how a well-designed ontology architecture coupled with a semantic technology-based approach to information management could improve the productivity and efficiency of MSM-related standards development, management and implementation [19, 20].

4 Agile Development Approach

We take an Agile Development approach (Agile approach), to ontology architecture design, development, and implementation [1]. Agile Development begins with an envisioning phase in which we rapidly collect and prioritize user needs, perform coarse grained architecture modeling, and roughly estimate scope. Then we implement the architecture by building incremental capability in short design and development cycles called sprints. The intent is to allow the architecture to gradually evolve based on emerging stakeholder requirements and lessons learned from each sprint [1]. When fully mature, the Cyber-Security Ontology Architecture will represent a comprehensive, standards-based family of ontologies.

4.1 Envisioning Phase

We gathered high level requirements from domain experts, which are expressed as obstacles to adoption in Section 3 of this document. Then we developed a coarse model of the CSIEM lifecycle to provide a rough estimate of scope. We mapped the current MSM-related controlled vocabularies (CVs) to the CSIEM lifecycle model to produce a CV architecture as illustrated in Figure 1. Acronym expansions for the standard names in Figure 1 are located in the References section, reference numbers 1,2,3,4,5,6,7,12,14,18, and 29.

Finally, we performed a vocabulary analysis, identifying gaps and overlaps while extracting common concepts for reuse across vocabularies. Results are illustrated in the first draft Cyber-Security Ontology Architecture as illustrated in Figure 2 [2,4,6,18]. The top two layers of the architecture designates the ontology-level tiers. We will eventually fill the gaps with new or existing ontologies while reducing vocabulary overlap to only intentional variation in order to control complexity and improve structural and syntactic information interoperability.

The lowest tier of the architecture designates the standards value-level CV content followed by the CV representations in the third tier. These two CV tiers are the sources for the upper two ontology-level architecture tiers. Above the CV tiers, the third tier contains ontologies that are specific to the cyber-security domain. Finally the upper-most tier contains common ontologies that emerged from vocabulary overlap analysis. Development of the first draft Cyber-Security Ontology Architecture marks the end of the envisioning phase of development and the beginning of Sprint 1 implementation. The ontologies that are encircled with red ovals are those that have been developed or adopted during the Sprint 1 implementation phase. We adopt or derive from existing ontologies where possible. The ontologies are encoded in the Web Ontology Language (OWL) [25].

Fig. 1. CSIEM CV Architecture

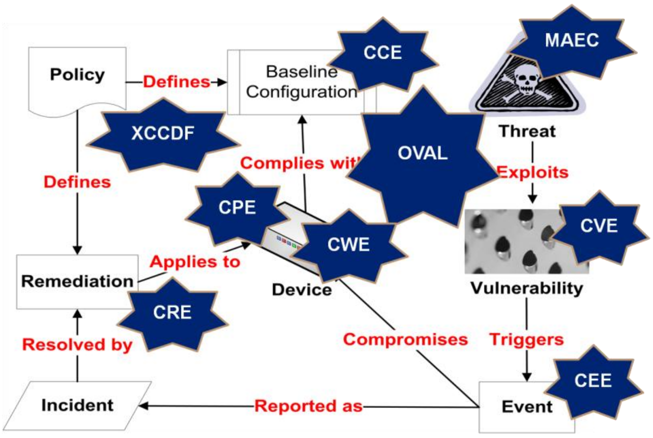
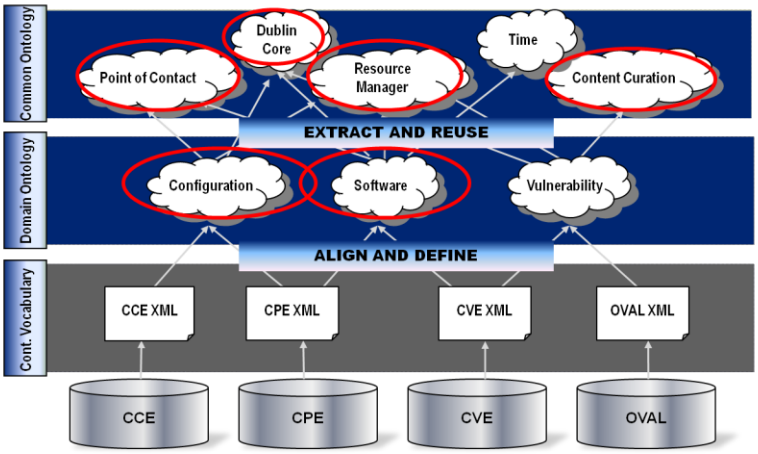


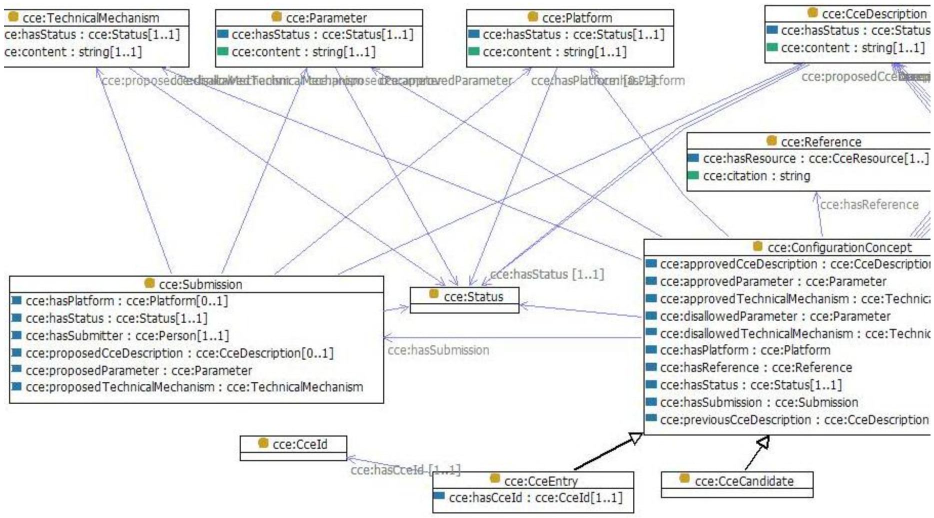
Fig. 2. Cyber-Security Ontology Architecture Concept Diagram



4.2 Cyber-Security Ontology Architecture Implementation Sprint 1

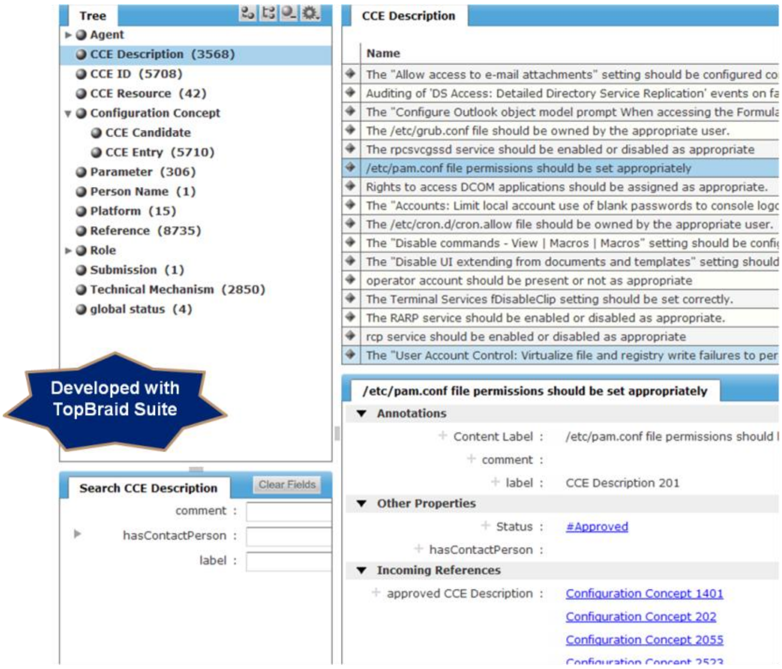
Sprint 1 focused on improving the vocabulary management process and produced five ontologies. Four of these are common ontologies, including: an OWL (Web Ontology Language) representation of the Dublin Core metadata standard [9,25]; a Resource Manager ontology which imports the Dublin Core model and references parts of SKOS (Simple Knowledge Organization System) [28]; a Point-of-Contact ontology (which was derived from the FOAF [10] and VCard ontologies) [26]; and a Content Curation ontology. The domain ontology was derived from the Common Configuration Enumeration (CCE) CV. It includes the Content Curation ontology and parts of the other three common ontologies. Figure 3 illustrates the structure of the CCE Vocabulary Manager Ontology's core concepts.

Fig. 3. CCE Vocabulary Manager Ontology Core Concepts


cce:TechnicalMechanism
ce:hasStatus : cce:Status[1..1]
ce:content : string[1..1]
cce:Parameter
cce:hasStatus : cce:Status[1..1]
cce:content : string[1..1]
cce:Platform
cce:hasStatus : cce:Status[1..1]
cce:content : string[1..1]
cce:CceDescription
cce:hasStatus : cce:Status
cce:content : string[1..1]
cce:proposedCce
cce:Reference
cce:hasResource : cce:CceResource[1..]
cce:citation : string
cce:hasReference
cce:ConfigurationConcept
cce:approvedCceDescription : cce:CceDescriptio
cce:approvedParameter : cce:Parameter
cce:approvedTechnicalMechanism : cce:Technic
cce:disallowedParameter : cce:Parameter
cce:disallowedTechnicalMechanism : cce:Technic
cce:hasPlatform : cce:Platform
cce:hasReference : cce:Reference
cce:hasStatus : cce:Status[1..1]
cce:hasSubmission : cce:Submission
cce:previousCceDescription : cce:CceDescription
cce:hasStatus [1..1]
cce:Status
cce:hasSubmission
cce:Submission
cce:hasPlatform : cce:Platform[0..1]
cce:hasStatus : cce:Status[1..1]
cce:hasSubmitter : cce:Person[1..1]
cce:proposedCceDescription : cce:CceDescription[0..1]
cce:proposedParameter : cce:Parameter
cce:proposedTechnicalMechanism : cce:TechnicalMechanism
cce:CceId
cce:hasCceId [1..1]
cce:CceEntry
cce:hasCceId : cce:CceId[1..1]
cce:CceCandidate

We converted the existing CCE XML content into over 27,000 RDF [27] instances to create the CCE Vocabulary Manager knowledge base, which contains over 500,000 RDF triples. Then we implemented a reference Semantic Web application using Top Quadrant's TopBraid Suite [24]. This application enables CCE content analysts to view, query, navigate, edit and track the status of CCE content in the knowledge base. Figure 4 shows a screenshot of the CCE vocabulary management application. The RDF graph structure eliminates the need for redundant content that is required of tabular and hierarchical structures. The OWL ontology expands the single tacit CCE Entry relation to many explicit user-defined relations among CCE instances. These capabilities, among others, have the potential to streamline vocabulary management processes and improve content quality across MSM-related standards.

Fig. 4. CCE Vocabulary Management Web Application



comment :
hasContactPerson :
label :

Name
The "Allow access to e-mail attachments" setting should be configured co
Auditing of 'DS Access: Detailed Directory Service Replication' events on fa
The "Configure Outlook object model prompt When accessing the Formula
The /etc/grub.conf file should be owned by the appropriate user.
The rpcsvcgssd service should be enabled or disabled as appropriate
/etc/pam.conf file permissions should be set appropriately
Rights to access DCOM applications should be assigned as appropriate.
The "Accounts: Limit local account use of blank passwords to console logo
The /etc/cron.d/cron.allow file should be owned by the appropriate user.
The "Disable commands - View | Macros | Macros" setting should be confi
The "Disable UI extending from documents and templates" setting should
operator account should be present or not as appropriate
The Terminal Services fDisableClip setting should be set correctly.
The RARP service should be enabled or disabled as appropriate.
rcp service should be enabled or disabled as appropriate
The "User Account Control: Virtualize file and registry write failures to per

/etc/pam.conf file permissions should be set appropriately

▼ Annotations

Content Label : /etc/pam.conf file permissions should l
comment :
label : CCE Description 201

▼ Other Properties

Status : #Approved
hasContactPerson :

▼ Incoming References

approved CCE Description : Configuration Concept 1401
Configuration Concept 202
Configuration Concept 2055
Configuration Concept 2523

5 Future Work

In the near future, we will refine the vocabulary management reference application while building out the ontology architecture. A longer term goal is to develop an end user reference implementation that semi-automates the mapping of proprietary tool output to standard vocabularies.

Acknowledgement Thank you to MITRE subject matter experts Matthew Wojcik, Jonathan Baker and David Mann for their valuable contributions to this research.

References

1. Amber, Scott W.: Agile Model Driven Development, http://www.agilemodeling.com/essays/amdd.htm
2. CCE: Common Configuration Enumeration, http://cce.mitre.org/
3. CEE Board: Common Event Expression Technical Report, Department G026, The MITRE Corporation (2007)
4. CPE: Common Platform Enumeration, http://cpe.mitre.org/files/cpe-specification_2.2.pdf

5. CRE: Common Remediation Enumeration, http://scap.nist.gov/events/2010/saddw/presentations/remediation.pdf
6. CVE: Common Vulnerability and Exposures, http://cve.mitre.org/
7. CWE: Common Weakness Enumeration, http://cwe.mitre.org/
8. Deshayes, Laurent; Foufou, Sebti; et al.: An Ontology Architecture for Standards Integration and Conformance in Manufacturing, 6th International IDDME, Grenoble, France, May 17-19 2006. http://stl.mie.utoronto.ca/publications/P0057paper.pdf
9. Dublin Core Metadata Inititative: Dublin Core Element Set, http://dublincore.org/documents/dces/
10. FOAF: Friend-of-a-Friend Vocabulary Specification, http://xmlns.com/foaf/spec/
11. ISO: ISO 639-4:2010, http://www.iso.org/iso/catalogue_detail.htm?csnumber=39535
12. MAEC: Malware Attribute Enumeration and Characterization, http://maec.mitre.org/
13. MSM: Making Security Measurable, http://measurablesecurity.mitre.org/
14. Mann, David: An Introduction to the Common Configuration Enumeration (CCE), Technical Report, Department G022, The MITRE Corporation (2008)
15. NIST: Interagency Report 7511, SCAP Validation Derived Test Requirements, http://csrc.nist.gov/publications/drafts/nistir-7511/draft-nistir-7511_rev1.pdf (2009)
16. NIST: SCAP (Security Content Automation Protocol), http://scap.nist.gov/
17. Obrst, Leo: *Ontological Architectures*, *Chapter 2 in Part One: Ontology as Technology* in the book: TAO – Theory and Applications of Ontology, Volume 2: The Information-science Stance, Michael Healy, Achilles Kameas, Roberto Poli, eds. Springer, (2010).
18. OVAL: Open Vulnerability and Assessment Language, http://oval.mitre.org/
19. Parmelee, Mary: Toward the Semantic Interoperability of the Security Information and Event Management Lifecycle, In: AAAI Intelligent Security Workshop, http://www.tzi.de/~edelkamp/secart/IntSec.pdf (2010)
20. Parmelee, Mary; Nichols, Deborah; Obrst, Leo: A Net-Centric Metadata Framework for Service Oriented Environments. IJMSO 4 (4): 250 – 260 (2009)
21. Pease, A., Niles, I., and Li, J.: The Suggested Upper Merged Ontology: A Large Ontology for the Semantic Web and its Applications. In Working Notes of the AAAI-2002 Workshop on Ontologies and the Semantic Web, Edmonton, Canada (2002)
22. Princeton University: WordNet, http://wordnet.princeton.edu/
23. Probst, F., M. Lutz: Giving Meaning to GI Web Service Descriptions, WSMAI (2004)
24. Top Quadrant: TopBraid Suite, http://topquadrant.com/products/TB_Suite.html
25. W3C: OWL Overview, http://www.w3.org/TR/owl-features/ (2004)
26. W3C: Representing vCard Objects in RDF, http://www.w3.org/Submission/vcard-rdf/ (2010)
27. W3C: Resource Description Framework (RDF) Semantics, W3C Recommendation http://www.w3.org/TR/rdf-mt/ (2004)
28. W3C SWDWG: SKOS, http://www.w3.org/2004/02/skos/ (2004)
29. XCCDF: Specification for the Extensible Configuration Checklist Description Format (XCCDF) Version 1.1.4, http://csrc.nist.gov/publications/nistir/ir7275r3/NISTIR-7275r3.pdf (2008)
30. W3C XSWG: XML Schema Part 1: Structures, http://www.w3.org/TR/2001/PR-xmlschema-1-20010330/ (2001)