

Network Working Group  
Internet-Draft  
Intended status: Informational  
Expires: August 15, 2013

D. Waltermire, Ed.  
NIST  
February 11, 2013

Security Automation and Continuous Monitoring (SACM) Architecture  
draft-waltermire-sacm-architecture-00

Abstract

This document identifies the architectural components, data flows, and the supporting standards needed to define an interoperable automation infrastructure required to support timely, accurate and actionable situational awareness over an organization's IT systems. This architecture is based on previous use case and requirements analysis. Automation tools implementing the continuous monitoring approach described in this document will utilize this infrastructure together with existing and emerging event, incident and network management standards to provide visibility into the state of assets, user activities and network behavior. Stakeholders will be able to use these tools to aggregate and analyze relevant security and operational data to understand the organizations security posture, quantify business risk, and make informed decisions that support organizational objectives while protecting critical information. Organizations will be able to use these tools to augment and automate information sharing activities to collaborate with partners to identify and mitigate threats. Other automation tools will be able to integrate with these capabilities to enforce policies based on human decisions to harden systems, prevent misuse and reduce the overall attack surface.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 15, 2013.

## Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
1.1. Overview . . . . .	3
1.2. Terminology . . . . .	4
1.3. Requirements . . . . .	4
2. Functional Components . . . . .	4
2.1. Controller . . . . .	5
2.1.1. Functions . . . . .	5
2.1.2. Interactions . . . . .	5
2.2. Content Repository . . . . .	6
2.3. Evaluator . . . . .	6
2.4. Sensor . . . . .	6
2.5. Data Storage . . . . .	6
3. Data Flows . . . . .	6
3.1. DF1: Content Retrieval . . . . .	7
3.2. DF2: Collection Tasking . . . . .	7
3.3. DF3: Collected Data Publication . . . . .	7
3.4. DF4: Collected Data Query . . . . .	7
4. Data Exchange Models and Communications Protocols . . . . .	7
4.1. Data Formats . . . . .	8
4.2. Communication Protocols . . . . .	8
5. IANA Considerations . . . . .	8
6. Security Considerations . . . . .	8
7. Acknowledgements . . . . .	9
8. Informative References . . . . .	9
Appendix A. Additional Stuff . . . . .	9
Author's Address . . . . .	9

## 1. Introduction

This document provides an architectural approach for addressing the orchestration, collection and analysis of endpoint posture. This architecture addresses the SACM Architecture milestone defined in the draft SACM charter. The focus of this architecture is to being to define an interoperable, automation infrastructure required to support timely, accurate and actionable situational awareness over an organization's IT systems. This document enumerates components, data flows and the supporting standards needed to achieve this vision.

### 1.1. Overview

The architecture identified in this document provides a foundation for creating interoperable automation tools and continuous monitoring solutions that provide visibility into the state of assets, user activities, and network behavior. Stakeholders will be able to use tools based on this architecture to aggregate and analyze relevant security and operational data pertaining to endpoints to understand the organizations security posture and make informed decisions that support organizational objectives while protecting critical information. Organizations will be able to use tools supporting this architecture to augment and automate information sharing activities to collaborate with partners to identify and mitigate threats. Other automation tools will be able to integrate with these capabilities to enforce policies based on human decisions to harden systems, prevent misuse and reduce the overall attack surface.

The architecture diagram in Figure 1 illustrates the overall architecture approach. It identifies the components that participate in the architecture and the data flows (DF) that enable information to be exchanged between them.

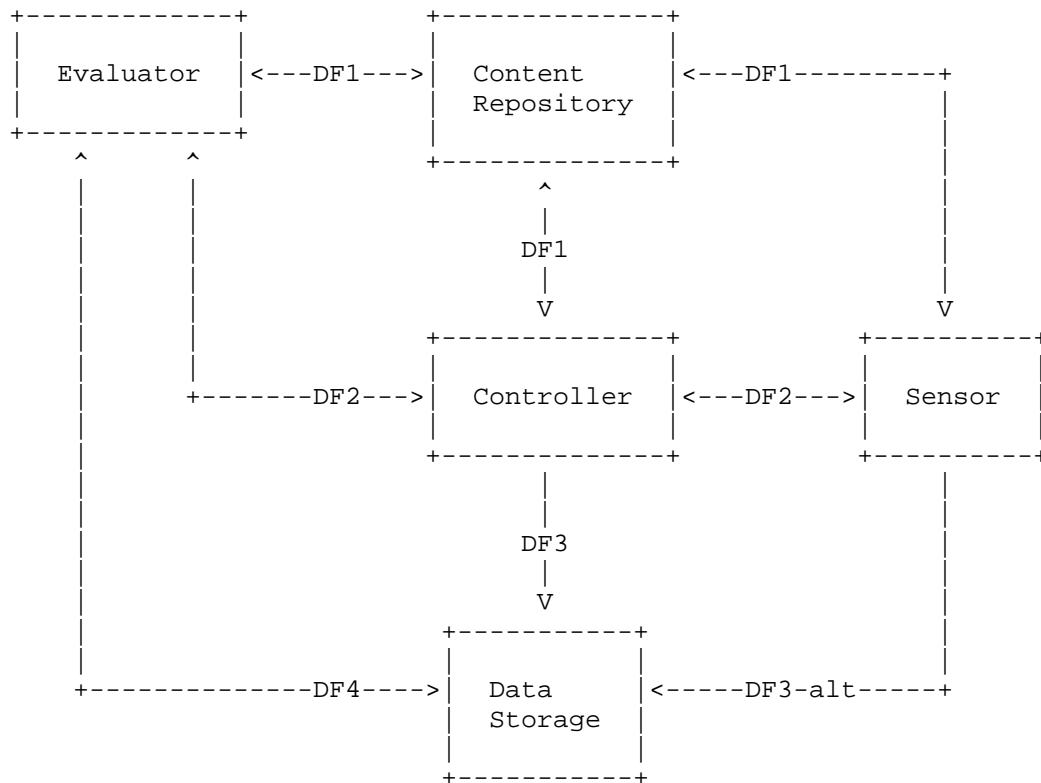


Figure 1

## 1.2. Terminology

Add in glossary items from use cases?

## 1.3. Requirements

Reference the SACM use cases document.

## 2. Functional Components

This section describes the functional components included in this architecture.

## 2.1. Controller

The Controller component is responsible for directing collection activities based on organizational security policy and available relevant metadata. It manages data collection tasks it receives, orchestrating sensors as needed to fulfill the tasks. The nature of the tasks received by the Controller may vary. They may be one-time tasks focused on collecting a single data set, reoccurring tasks that occur on a predefined interval, or real-time tasks that continue to collect information based on events

### 2.1.1. Functions

The controller provides the following functions:

#### Task Management

- \* The Controller processes incoming data collection task requests. It decomposes each task request into one or more data collection sub-tasks required to be performed by each Sensor.
- \* It creates sub-tasks for any scheduled tasking it is managing at the appropriate intervals.
- \* It tracks all sub-tasks currently being executed by sensors.

#### Sensor Management

- \* It dispatches any sub-tasks to the appropriate sensors.
- \* Collected data provided by the sensor is marshalled to the appropriate data store.

### 2.1.2. Interactions

The Controller interacts with other components in this architecture in the following ways:

- o The Controller receives data collection tasks from the Evaluator describing a new data collection task that needs to be performed.
- o The Controller retrieves content from the Content Repository that is needed to understand what specific data collections are required to be performed by each Sensor under its management to satisfy a data collection task.

- o The Controller interacts with each Sensor under its management that is needed to ensure that the appropriate data collection activities on the sensor are performed to address a data collection task. As data is collected and once data collection is complete the Controller receives data collection results from the sensor.

## 2.2. Content Repository

A repository of security metadata that can be used to drive security-oriented processes (e.g. vulnerability, configuration, asset data, assessment/collection methods). This is long-lived, infrequently changing information that is provided from a variety of external information sources.

The methods used to maintain information in a content repository is currently out of scope.

## 2.3. Evaluator

An upstream component that queries collected state information to perform analysis generating measurements and compliances results.

## 2.4. Sensor

Responsible for collecting actual system state information (e.g. configurations, software inventory, patch) based on data collection sub-tasks provided by the Controller. It uses data collection instructions provided by the content repository (e.g. SCAP-style assessment content). This could be an agent on an endpoint or a remote collection system with or without privileged access to the endpoint.

## 2.5. Data Storage

An upstream component that receives collected state information. This could be a data repository, an information processor that acts on the provided information or a process that routes information to other sources. This component supports SACM use cases UC2 and UC3.

## 3. Data Flows

The following data flows, also called interfaces, describe the nature of specific inter-component communications.

### 3.1. DF1: Content Retrieval

This data flow is used to provide any digital content and supporting metadata that is needed to drive data collection and analysis processes.

The following interactions are supported by this data flow:

- o The Controller uses this data flow to acquire the information it needs to determine what actions to instruct the sensors to perform. The Controller may also store policy decisions for future use in the content repository for future use.
- o The sensor uses this data flow to retrieve any data/content that is needed to perform collection activities.
- o The Evaluator uses this data flow to retrieve any content that describes the expected state and analysis rules needed to make measurements and determine compliance with organizational policy.

### 3.2. DF2: Collection Tasking

This is a control channel that is used to enable dynamic management of the information collected by the Sensor. Data collection tasks containing instruction of what to collect, and potentially how to collect, are exchanged using this data flow. These instructions may point to assessment content stored in the Content Repository.

### 3.3. DF3: Collected Data Publication

Used to make collected information available to other "upstream" components that archive the information for future use or perform additional analysis/processing.

### 3.4. DF4: Collected Data Query

Used by the Evaluator and other external components to query previously collected data.

## 4. Data Exchange Models and Communications Protocols

Document where existing work exists, what is currently defined by SDOs, and any gaps that should be addressed. Point to existing standards when available. Describe emerging efforts that may be used for the creation of new standards. For gaps provide insight into what would be a good fit for SACM or another IETF working groups.

This will help us to identify what is needed for SACM to work on. This section will help determine which of the specifications can be normatively referenced and what needs to be addressed in the IETF. This should help us determine any protocol or guidance documentation we will need to generate.

Things to address:

For IETF related efforts, discuss work in NEA and MILE working groups. Address SNMP, NetConf and other efforts as needed.

Reference any Security Automation work that is applicable.

#### 4.1. Data Formats

The functional capabilities described in the SACM Use Cases document require a significant number of models to be selected or defined. A "model" in this sense is a logical arrangement of information that may have more than one syntactic binding. For the purpose of this document, only the logical data model is considered. However, where appropriate, example data models that may have well-defined syntactic expressions may be referenced.

#### 4.2. Communication Protocols

Document these.

### 5. IANA Considerations

This memo includes no request to IANA.

All drafts are required to have an IANA considerations section (see RFC 5226 [RFC5226] for a guide). If the draft does not require IANA to do anything, the section contains an explicit statement that this is the case (as above). If there are no requirements for IANA, the section will be removed during conversion into an RFC by the RFC Editor.

### 6. Security Considerations

All drafts are required to have a security considerations section. See RFC 3552 [RFC3552] for a guide.



## 7. Acknowledgements

The author would like to acknowledge the members of the SACM mailing list for their keen and insightful feedback on the concepts and text within this document.

## 8. Informative References

- [RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, July 2003.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

## Appendix A. Additional Stuff

This becomes an Appendix if needed.

## Author's Address

David Waltermire (editor)  
National Institute of Standards and Technology  
100 Bureau Drive  
Gaithersburg, Maryland 20877  
USA

Phone:  
Email: david.waltermire@nist.gov



Network Working Group  
Internet-Draft  
Intended status: Informational  
Expires: August 15, 2013

D. Waltermire, Ed.  
NIST  
A. Montville  
TW  
February 11, 2013

Analysis of Security Automation and Continuous Monitoring (SACM) Use  
Cases

draft-waltermire-sacm-use-cases-04

Abstract

This document identifies use cases, derived functional capabilities, and requirements needed to provide a foundation for creating interoperable automation tools and continuous monitoring solutions that provide visibility into the state of endpoints, user activities, and network behavior. Stakeholders will be able to use these tools to aggregate and analyze relevant security and operational data to understand the organizations security posture, quantify business risk, and make informed decisions that support organizational objectives while protecting critical information. Organizations will be able to use these tools to augment and automate information sharing activities to collaborate with partners to identify and mitigate threats. Other automation tools will be able to integrate with these capabilities to enforce policies based on human decisions to harden systems, prevent misuse and reduce the overall attack surface.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 15, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	4
1.1. Requirements Language . . . . .	4
2. Key Concepts . . . . .	5
3. Use Cases . . . . .	7
3.1. UC1: Endpoint Posture Assessment . . . . .	7
3.1.1. Goals . . . . .	7
3.1.2. Main Success Scenario . . . . .	7
3.2. UC2: Enforcement of Acceptable State . . . . .	8
3.2.1. Goal . . . . .	8
3.2.2. Main Success Scenario . . . . .	8
3.3. UC3: Security Control Verification and Monitoring . . . . .	8
3.3.1. Goal . . . . .	8
3.3.2. Main Success Scenario . . . . .	8
4. Functional Capabilities and Requirements . . . . .	9
4.1. Capabilities Supporting UC1 . . . . .	9
4.1.1. Asset Management . . . . .	9
4.1.1.1. Concepts . . . . .	10
4.1.1.2. Requirements . . . . .	10
4.1.2. Data Collection . . . . .	11
4.1.2.1. Concepts . . . . .	11
4.1.2.2. Requirements . . . . .	12
4.1.3. Assessment Result Analysis . . . . .	13
4.1.3.1. Concepts . . . . .	13
4.1.3.2. Requirements . . . . .	13
4.1.4. Content Management . . . . .	14
4.1.4.1. Concepts . . . . .	14
4.1.4.2. Requirements . . . . .	14
4.2. Capabilities Supporting UC2 . . . . .	15
4.2.1. Assessment Query and Transport . . . . .	15
4.2.2. Acceptable State Enforcement . . . . .	15
4.3. Capabilities Supporting UC3 . . . . .	15
4.3.1. Tasking and Scheduling . . . . .	15
4.3.2. Data Aggregation and Reporting . . . . .	16
5. IANA Considerations . . . . .	17
6. Security Considerations . . . . .	17
7. Terms and Definitions . . . . .	17
8. Acknowledgements . . . . .	19
9. References . . . . .	19
9.1. Normative References . . . . .	19
9.2. Informative References . . . . .	19
Authors' Addresses . . . . .	20

## 1. Introduction

This document addresses foundational use cases in security automation. These use cases may be considered when establishing a charter for the Security Automation and Continuous Monitoring (SACM) working group within the IETF. This working group will address as many of the standards needed to define an interoperable, automation infrastructure required to support timely, accurate and actionable situational awareness over an organization's IT assets. This document enumerates use cases and breaks down related concepts and related requirements for capabilities that cross many IT security information domains.

Sections Section 2, Section 3, and Section 4 of this document respectively focus on:

- Defining the key concepts used within the document providing a common frame of reference;

- Identifying foundational use cases that represent classes of stakeholders, goals, and usage scenarios;

- A set of derived functional capabilities and associated requirements that are needed to support the use cases;

The concepts identified in this document provide a foundation for creating interoperable automation tools and continuous monitoring solutions that provide visibility into the posture of endpoints, user activities, and network behavior. Stakeholders will be able to use these tools to aggregate and analyze relevant security and operational data to understand the organizations security posture, quantify business risk, and make informed decisions that support organizational objectives while protecting critical information. Organizations will be able to use these tools to augment and automate information sharing activities to collaborate with partners to identify and mitigate threats. Other automation tools will be able to integrate with these capabilities to enforce policies based on human decisions to harden systems, prevent misuse and reduce the overall attack surface.

### 1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

## 2. Key Concepts

The operational methods we use within the bounds of our present realities are failing us - we are falling behind. We have begun to recognize that the evolution of threat agents, increasing system complexity, rapid situational security change, and scarce resources are detrimental to our success. There have been efforts to remedy our circumstance, and these efforts are generally known as "Security Automation."

Security Automation is a general term used to reference specifications originally created by the National Institute of Standards and Technology (NIST) and/or the MITRE Corporation. Security Automation generally includes languages, protocols (prescribed ways by which specification collections are used), enumerations, and metrics.

These specifications have provided an opportunity for tool vendors and enterprises building customized solutions to take the appropriate steps toward enabling Security Automation by defining common information expressions. In effect, common expression of information enables interoperability between tools (whether customized, commercial, or freely available). Another important capability common expression provides is the ability to automate portions of security processes to gain efficiency, react to new threats in a timely manner, and free up security personnel to work on more advanced problems within the processes in which they participate.

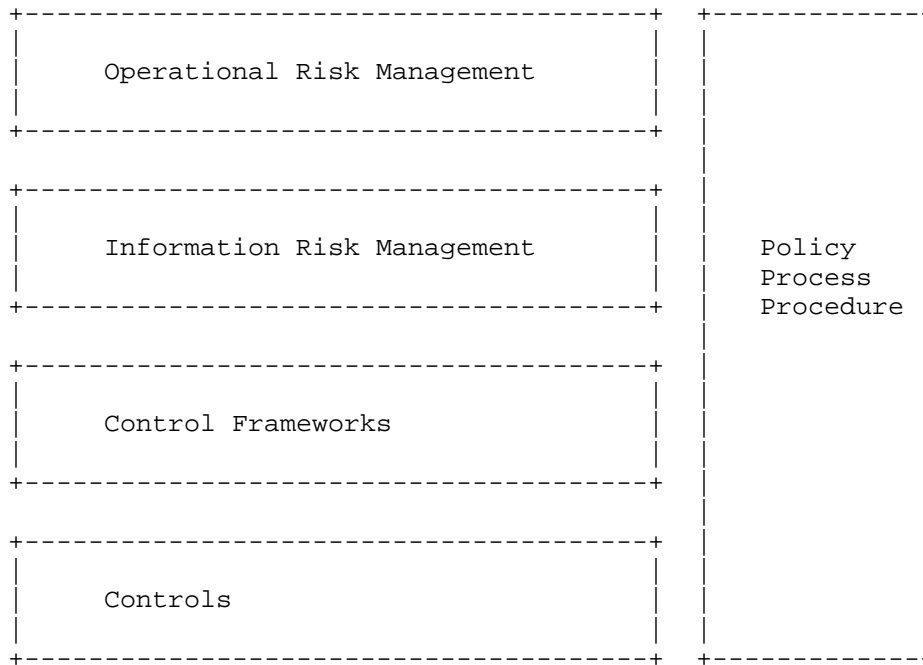


Figure 1

The figure above provides some context for our focus area. Organizations of all sizes will have a more or less formal risk management program, depending upon their maturity and organization-specific needs. A small business with only a few employees may not have a formally recognized risk management program, but they still lock the doors at night. Typically, financial entities and governments sit at the other end of the spectrum with often large, laborious risk frameworks. The point is that all organizations practice, to some degree, Operational Risk Management (ORM). An Information Risk Management (IRM) program is most likely a constituent of ORM (another constituent might be Financial Risk Management). In the Information Risk Management domain, we often use control frameworks to provide guidance for organizations practicing ORM in an information context, and these control frameworks define a variety of controls.

From ORM, IRM, control frameworks, and the controls themselves, organizations derive a set of organization-specific policies, processes, and procedures. Such policies, processes, and procedures make use of a library of supporting information commonly stipulated by the organization (i.e. enterprise acceptable use policies), but



often prescribed by external entities (i.e. Payment Card Industry Data Security Standards, Sarbanes-Oxley, or EU Data Privacy Directive). The focus of this document spans controls, certain aspects of policy, process, and procedure, and control frameworks.

### 3. Use Cases

This document addresses three use cases: Endpoint Posture Assessment, Enforcement of Acceptable State, Security Control Verification and Monitoring. Currently, the first use case, Endpoint Posture Assessment, is being pursued under the SACM charter. The additional use cases are included to provide broader context to this work and represents additional work that may be considered by SACM or another IETF working group in the future.

#### 3.1. UC1: Endpoint Posture Assessment

The Endpoint Posture Assessment use case involves collecting information about the posture of a given endpoint. This posture information is gathered and then published to appropriate data repositories to make collected information available for further analysis supporting organizational security processes.

##### 3.1.1. Goals

The primary goals of the endpoint Posture Assessment use case is:

- o To collect the posture of a given endpoint;
- o Make that posture available to the enterprise for further analysis and action; and
- o To assess that the endpoint's posture is in compliance with enterprise standards and, therefore, ensure alignment with enterprise policy.

##### 3.1.2. Main Success Scenario

1. Define a target endpoint to be assessed
2. Select acceptable state policies to apply to the defined target
3. Identify the endpoint being assessed
4. Collect posture attributes from the target

5. Communicate target identity and collected posture to external system for evaluation
6. Compare collected posture attributes from the target endpoint with expected state values as expressed in acceptable state policies

### 3.2. UC2: Enforcement of Acceptable State

Controlling access to a desired resource based on the compliance of an endpoint or user with enterprise policy.

#### 3.2.1. Goal

Allow or deny access to a desired resource based on the compliance of an endpoint or user with enterprise policy.

#### 3.2.2. Main Success Scenario

1. An entity (user on an endpoint or the endpoint itself) requests access to a given resource (i.e. network connection, service)
2. Assessment of endpoint posture is achieved using UC1: Endpoint Posture Assessment
3. Based on assessment results (i.e. compliance level with enterprise policy)
  - A. Endpoint or user is allowed access to requested resource, or
  - B. Endpoint or user is denied access to requested resource

### 3.3. UC3: Security Control Verification and Monitoring

This use case involves continuous (uninterrupted) and continual (periodic) monitoring of a set of target endpoints to determine the degree of compliance with acceptable state policies within an enterprise.

#### 3.3.1. Goal

Continuous assessment of the implementation and effectiveness of security controls based on machine processable content.

#### 3.3.2. Main Success Scenario

1. Define set of target endpoints to be assessed.
2. Select acceptable state policies to apply to set of target endpoints
3. Define assessment trigger based on either a
  - A. Time period, or
  - B. An event (e.g. endpoint, network, organizational).
4. Define result reporting/alerting criteria
5. Enable continuous assessment

#### 4. Functional Capabilities and Requirements

In general, the activities of managing assets, configurations, and vulnerabilities are common between UC1, UC2, and UC3. UC2 uses these activities to either grant or deny an entity access to a requested resource. UC3 uses these activities in support of compliance measurement on a periodic basis.

At the most basic level, an enterprise needing to satisfy these use cases will need certain capabilities to be met. Specifically, we are talking about risk management capabilities. This is the central problem domain, so it makes sense to be able to convey information about technical and non-technical controls, benchmarks, control requirements, control frameworks and other concepts in a common way.

##### 4.1. Capabilities Supporting UC1

The capabilities in this section support assessing endpoint posture in an automated manner as described in Section 3.1.

##### 4.1.1. Asset Management

Organizations manage a variety of assets within their enterprise. Supporting the use cases in this document requires management of assets including: endpoints, the hardware they are composed of, installed software, hardware/software licenses used, and any appropriate configurations. Effective Asset Management is a critical foundation upon which all else in risk management is based. There are two important facets to asset management: 1) understanding coverage (what and how many assets are under control) and, 2) understanding specific asset details. Coverage is fairly straightforward - assessing 80% of the enterprise assets is better

than assessing 50% of the enterprise assets. Getting asset details is comparatively subtle - if an enterprise does not have a precise understanding of its assets, then all acquired data and consequent actions taken based on the data are considered suspect. Assessing assets (managed and unmanaged) requires that we have visibility into the posture of endpoints, the ability to understand the composition and relationships between different assets types, and the ability to properly characterize them at the outset and over time.

#### 4.1.1.1. Concepts

Managing endpoints and the different types of assets that compose them involves initially discovering and characterizing each asset instance, and then identify them in a common way. Characterization may take the form of logical characterization or security characterization, where logical characterization may include business context not otherwise related to security, but which may be used as information in support of decision making later in risk management workflows.

The following list details the requisite Asset Management capabilities:

- o Discover assets in the enterprise
- o For a given endpoint, understand the composition and relationship of its constituent assets
- o Characterize assets according to security and non-security asset properties
- o Identify and describe assets using a common vocabulary between implementations
- o Reconcile asset representations originating from disparate tools
- o Manage asset information throughout the asset's life cycle

#### 4.1.1.2. Requirements

A method **MUST** be provided for identifying an endpoint (asset identification) as a unique entity within the enterprise.

The endpoint identifier **SHOULD** be able to be determined in an automated manner.

The endpoint identifier, as communicated between entities, **SHOULD** be held to a minimal size.

A method **MUST** be provided for defining an endpoint (asset classification) based on a set of organizationally relevant properties (e.g. organizational affiliation, criticality, function).

#### 4.1.2. Data Collection

Related to managing the assets related to endpoints, and central to any automated assessment solution, is the ability to collect data from (or related to) an endpoint (some might call this "harvesting"). Of particular interest is data representing the security state of the endpoint and its constituent assets. The primary interest of the activities demanding data collection is centered on policy attribute collection related to installed hardware and software configuration items, and network device configuration items among others.

##### 4.1.2.1. Concepts

There are many valid perspectives to take when considering required data collection capabilities. The nature of data collected relating to endpoints supports a variety of information domains including: security configuration management (SCM) and vulnerability management. SCM deals with the configuration of endpoints (infrastructure devices and computing hosts) including the software installed and in use on these devices. Vulnerability management involves identifying the patch level of software installed on the device and the identification of insecure custom code (e.g. web vulnerabilities). All vulnerabilities need to be addressed as part of a comprehensive risk management program, which is a superset of software vulnerabilities. Thus, the capability of assessing non-software vulnerabilities applicable to the in-scope system is required. Additionally, it may be necessary to support non-technical assessment of data relating to assets such as aspects related to operational and management controls.

The following assessment capabilities support SCM relative to a target asset:

- o Collect the state of technical controls including, but not necessarily limited to:
  - \* Software inventory (e.g. operating system, applications, patches)
  - \* Configuration settings
- o Collect the state of non-technical controls commonly called administrative controls (i.e. policy, process, procedure)

#### 4.1.2.2. Requirements

One or more data formats MUST be identified to describe instructions, data collection methods, to drive data collection (e.g. technical, interrogative).

One or more data formats MUST be identified to instruct what posture attributes need to be collected for a specific set of endpoints.

A method MUST be provided to include OPTIONAL instructions on describing what content must be run on the endpoint.

A method MUST be provided to include OPTIONAL instructions that determine how to collect data supporting any particular test for that endpoint.

A method MUST be provided for retrieving data collection instructions from a remote host (see Section Section 4.1.4).

One or more data formats MUST be identified to capture the results of data collection.

This expression MUST be capable of supporting the characterization of assets and any related configuration settings that together compose an endpoint.

A mechanism MUST be provided to identify the software and hardware asset instances that compose an endpoint.

An asset identifier SHOULD be able to be determined in an automated manner

An asset identifier, as communicated between entities, SHOULD be held to a minimal size.

An asset identifier SHOULD be able to be represented in a simple unambiguous manner, such as a reference, so that its embedded use in places like applicability clauses for individual benchmark tests can be kept from making their usage unwieldy.

A mechanism MUST be provided to associate configuration settings values to the installed software.

A mechanism MUST be provided to identify additional collected posture attribute/value pairs related to an endpoint.

A mechanism MUST be provided to identify the endpoint the results pertain to (see Section Section 4.1.1).

A mechanism MUST be provided to associate the data collection method with the collected value.

A mechanism MUST be provided to include provenance information describing what sensor or software collected the data.

A mechanism MUST be provided to include entailment information, perhaps by reference, describing the methodology used to collect the data.

A method of communicating data collection results to another system for further analysis MUST be identified.

TODO: Communicate, unambiguously and to the necessary level of detail\*\*, the asset details between software components

#### 4.1.3. Assessment Result Analysis

At the most basic level, the data collected needs to be analyzed for compliance to a standard stipulated by the enterprise. Analysis methods may vary between enterprises, but commonly take a similar form.

##### 4.1.3.1. Concepts

The following capabilities support the analysis of assessment results:

- o Comparing actual state to expected state
- o Scoring/weighting individual comparison results
- o Relating specific comparisons to benchmark-level requirements
- o Relating benchmark-level requirements to one or more control frameworks

##### 4.1.3.2. Requirements

A method MUST be provided for selecting acceptable state policy, describing how to evaluate collected information, based on characteristics of the endpoint and organizational policy.

A method MUST be provided for comparing collected data to expected state values (test evaluation).

Any results produced by analysis processes MUST be capable of being transformed into a human-readable format.

#### 4.1.4. Content Management

It should be clear by now that the capabilities required to support risk management state measurement will yield volumes of content. The efficacy of risk management state measurement depends directly on the stability of the driving content, and, subsequently, the ability to change content according to enterprise needs.

##### 4.1.4.1. Concepts

Capabilities supporting Content Management should provide the ability to create/define or modify content, as well as store and retrieve said content of at least the following types:

- o Configuration checklists
- o Assessment rules
- o Data collection rules and methods
- o Scoring models
- o Vulnerability information
- o Patch information
- o Asset characterization data and rules

Note that the ability to modify content is in direct support of tailoring content for enterprise-specific needs.

##### 4.1.4.2. Requirements

A protocol MUST be identified for retrieving SACM content from a content repository

A protocol MUST be identified for querying SACM content held in a content repository. The protocol MUST support querying content by applicability to asset characteristics.

TODO: Determine what content can or must be run on the endpoint

A protocol MUST be identified for curating SACM content in a content repository. Note: This might be an area where we can limit the scope of work relative to the initial SACM charter.



#### 4.2. Capabilities Supporting UC2

UC2 is dependent upon UC1 and, therefore, includes all of the capabilities described in Section 4.1. UC2 describes the ability to make a resource access decision based on an assessment of the requesting system (either by the system itself or on behalf of a user operating that system). There are two chief capabilities required to meet the needs expressed in Section 3.2: Assessment Query and Transport, and Acceptable State Enforcement.

##### 4.2.1. Assessment Query and Transport

Under certain circumstances, the system requesting access may be unknown, which can make querying the system problematic (consider a case where a system is connecting to the network and has no assessment software installed). Note that The Network Endpoint Assessment (NEA) protocols (PA-TNC [RFC5792], PB-TNC [RFC5793], PT-TLS [I-D.ietf-nea-pt-tls], and PT-EAP [I-D.ietf-nea-pt-eap]) may be used to query and transport the things to be measured.

##### 4.2.2. Acceptable State Enforcement

Once the assessment has been performed a decision to allow or deny access to the requested resource can be made. Making this decision is a necessary but insufficient condition for enforcement of acceptable state, and an implementation must have the ability to actively allow or deny access to the requested resource. For example, network enforcement may be implemented with RADIUS [RFC2865] or DIAMETER [RFC6733].

#### 4.3. Capabilities Supporting UC3

Recall that UC3 is dependent upon UC1 and therefore includes all of the capabilities described in Section 4.1. The difference in UC3 is the notion of when to assess rather than what to assess. Therefore, the capabilities described in this section are relevant only to the "when" and not to the "what."

##### 4.3.1. Tasking and Scheduling

The ability to task and schedule assessments is requisite for any effective risk management program. Tasking refers to the ability to create a set of instructions to be conveyed at a later time via scheduling. Tasking, therefore, involves selecting a set of assessment criteria, assigning that set to a group of assets, and expressing that information in a manner that can be consumed by a collection tool. Scheduling comes into play when the enterprise determines when to perform a specific assessment task (or set of

tasks). Scheduling may be expressed in a way that constrains tasks to execute only during defined periods, can be ad hoc, or may be triggered by the analysis of previous assessment results or events detected in the enterprise.

The following capabilities support Tasking and Scheduling:

- o Selection of assessment criteria
- o Defining in-scope assets (i.e. targeting)
- o Defining periodic assessments for a given set of tasks
- o Defining assessment triggers for a given set of tasks

#### 4.3.2. Data Aggregation and Reporting

Assessment results are produced for every asset assessed, and these results must be reported not only individually, but in the aggregate, and in accordance with enterprise needs. Enterprises should be able to aggregate and report on the data their assessments produce in a number of different ways in order to support different levels of decision making. At times, security operations personnel may be interested in understanding where the most critical risks exist in their enterprise so as to focus their remediation efforts in the most effective way (in terms of cost and return). At other times, only aggregated scores will matter, as might be the case when reporting to an information security manager or other executive-level role.

It is not the position of these capabilities to provide explicit details about how reports should be formatted for presentation, but only what information they should contain for a particular purpose. Furthermore, it is quite easy to imagine the need for a capability providing extensibility to aggregation and reporting.

Aggregating assessment results by the following capabilities supports Data Aggregation and Reporting

- o By asset characterization
- o By assessment criteria
- o By control framework
- o By benchmark
- o By other attributes/properties of assessment characteristics

- o Extensible aggregation and reporting

## 5. IANA Considerations

This memo includes no request to IANA.

All drafts are required to have an IANA considerations section (see RFC 5226 [RFC5226] for a guide). If the draft does not require IANA to do anything, the section contains an explicit statement that this is the case (as above). If there are no requirements for IANA, the section will be removed during conversion into an RFC by the RFC Editor.

## 6. Security Considerations

All drafts are required to have a security considerations section. See RFC 3552 [RFC3552] for a guide.

This section needs to be fleshed out to include concerns including:

- o Authentication
- o Authorization
- o Endpoint and user identity management
- o Encryption of communications
- o Content signing and validation
- o etc...

While not strictly a security concern, network bandwidth and similar communications requirements also need to be addressed.

## 7. Terms and Definitions

assessment

Defined in [RFC5209] as "the process of collecting posture for a set of capabilities on the endpoint (e.g., host-based firewall) such that the appropriate validators may evaluate the posture against compliance policy."

Within this document the use of the term is expanded to support other uses of collected posture (e.g. reporting, network enforcement, vulnerability detection, license management). The phrase "set of capabilities on the endpoint" includes: hardware and software installed on the endpoint."

#### asset

Defined in [RFC4949] as "a system resource that is (a) required to be protected by an information system's security policy, (b) intended to be protect by a countermeasure, or (c) required for a system's mission.

#### attribute

Defined in [RFC5209] as "data element including any requisite meta-data describing an observed, expected, or the operational status of an endpoint feature (e.g., anti-virus software is currently in use)."

#### endpoint

Defined in [RFC5209] as "any computing device that can be connected to a network. Such devices normally are associated with a particular link layer address before joining the network and potentially an IP address once on the network. This includes: laptops, desktops, servers, cell phones, or any device that may have an IP address."

Network infrastructure devices (e.g. switches, routers, firewalls), which fit the definition, are also considered to be endpoints within this document.

Based on the previous definition of an asset, an endpoint is a type of asset.

#### posture

Defined in [RFC5209] as "configuration and/or status of hardware or software on an endpoint as it pertains to an organization's security policy."

This term is used within the scope of this document to represent the state information that is collected from an endpoint (e.g. software/hardware inventory, configuration settings).

#### posture attributes

Defined in [RFC5209] as "attributes describing the configuration or status (posture) of a feature of the endpoint. For example, a Posture Attribute might describe the version of the operating system installed on the system."

Within this document this term represents a specific assertion about endpoint state (e.g. configuration setting, installed software, hardware). The phrase "features of the endpoint" refers to installed software or software components.

#### system resource

Defined in [RFC4949] as "data contained in an information system; or a service provided by a system; or a system capacity, such as processing power or communication bandwidth; or an item of system equipment (i.e., hardware, firmware, software, or documentation); or a facility that houses system operations and equipment."

## 8. Acknowledgements

The author would like to thank Kathleen Moriarty and Stephen Hanna for contributing text to this document. The author would also like to acknowledge the members of the SACM mailing list for their keen and insightful feedback on the concepts and text within this document.

## 9. References

### 9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

### 9.2. Informative References

[I-D.ietf-nea-pt-eap]  
Cam-Winget, N. and P. Sangster, "PT-EAP: Posture Transport (PT) Protocol For EAP Tunnel Methods",  
draft-ietf-nea-pt-eap-06 (work in progress),  
December 2012.

[I-D.ietf-nea-pt-tls]  
Sangster, P., Cam-Winget, N., and J. Salowey, "PT-TLS: A TLS-based Posture Transport (PT) Protocol",  
draft-ietf-nea-pt-tls-08 (work in progress), October 2012.

- [RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.
- [RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, July 2003.
- [RFC4949] Shirey, R., "Internet Security Glossary, Version 2", RFC 4949, August 2007.
- [RFC5209] Sangster, P., Khosravi, H., Mani, M., Narayan, K., and J. Tardo, "Network Endpoint Assessment (NEA): Overview and Requirements", RFC 5209, June 2008.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.
- [RFC5792] Sangster, P. and K. Narayan, "PA-TNC: A Posture Attribute (PA) Protocol Compatible with Trusted Network Connect (TNC)", RFC 5792, March 2010.
- [RFC5793] Sahita, R., Hanna, S., Hurst, R., and K. Narayan, "PB-TNC: A Posture Broker (PB) Protocol Compatible with Trusted Network Connect (TNC)", RFC 5793, March 2010.
- [RFC6733] Fajardo, V., Arkko, J., Loughney, J., and G. Zorn, "Diameter Base Protocol", RFC 6733, October 2012.

#### Authors' Addresses

David Waltermire (editor)  
National Institute of Standards and Technology  
100 Bureau Drive  
Gaithersburg, Maryland 20877  
USA

Phone:  
Email: david.waltermire@nist.gov

Adam W. Montville  
Tripwire, Inc.  
101 SW Main Street, Suite 1500  
Portland, Oregon 97204  
USA

Phone:  
Email: [amontville@tripwire.com](mailto:amontville@tripwire.com)

