        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

Copyright Notice

Table of Contents

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.

```
    +-------------+          +--------------+
    |             |          |              |
    |  Evaluator  |<---DF1--->|  Content     |<---DF1---------+
    |             |          |  Repository  |                |
    +-------------+          |              |                |
       ^        ^            +--------------+                |
       |        |                  ^                         |
       |        |                  |                         |
       |        |                 DF1                        |
       |        |                  |                         |
       |        |                  V                         V
       |        |            +--------------+        +----------+
       |        |            |              |        |          |
       |      +-------DF2--->|  Controller  |<---DF2--->| Sensor   |
       |      |             |              |        |          |
       |      |             +--------------+        +----------+
       |      |                    |                     |
       |      |                    |                     |
       |      |                   DF3                    |
       |      |                    |                     |
       |      |                    V                     |
       |      |              +-----------+               |
       |      |              |           |               |
    +--------------DF4---->|  Data     |<-----DF3-alt-----+
       |      |              |  Storage  |
       |      |              |           |
       |      |              +-----------+
```

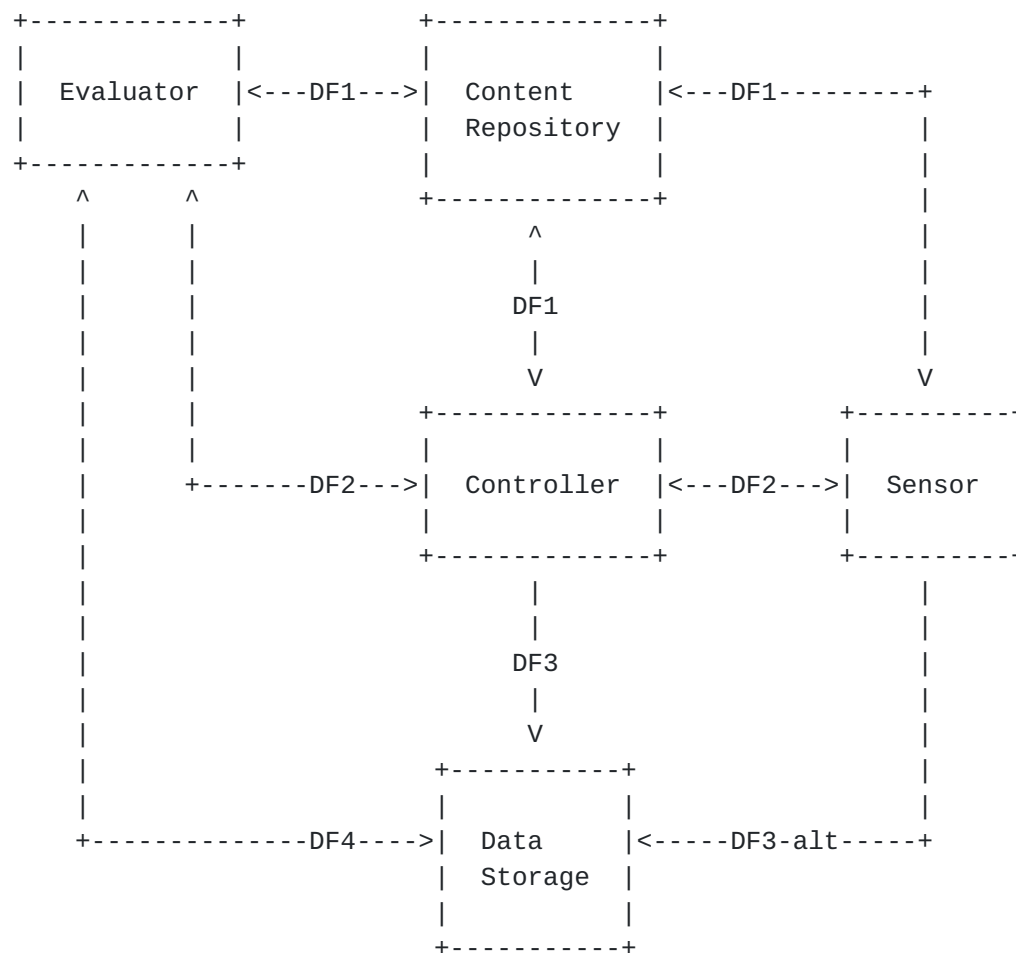   Figure 1

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 an 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