Network Working Group   D. Waltermire, Ed.
Internet-Draft   NIST
Intended status: Informational   A. Montville
Expires: March 11, 2013   TW
                                                       September 7, 2012

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

draft-waltermire-sacm-use-cases-02

Abstract

This document identifies foundational use cases, derived functional capabilities and requirements, architectural components, 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. Automation tools implementing a continuous monitoring approach 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 6
  1.1. Requirements Language 6
2. Key Concepts 7
3. Use Cases 9
  3.1. UC1: System State Assessment 9
    3.1.1. Goal 9
    3.1.2. Main Success Scenario 9
    3.1.3. Extensions 9
  3.2. UC2: Enforcement of Acceptable State 9
    3.2.1. Goal 9
    3.2.2. Main Success Scenario 9
    3.2.3. Extensions 10
  3.3. UC3: Security Control Verification and Monitoring 10
    3.3.1. Goal 10
    3.3.2. Main Success Scenario 10
    3.3.3. Extensions 10
4. Functional Capabilities 10
  4.1. Capabilities Supporting UC1 11
    4.1.1. Asset Management 11
    4.1.2. Data Collection 12
      4.1.2.1. Security Configuration Management 12
      4.1.2.2. Vulnerability Management 12
    4.1.3. Assessment Result Analysis 13
    4.1.4. Content Management 13
  4.2. Capabilities Supporting UC2 14
    4.2.1. Assessment Query and Transport 14
    4.2.2. Acceptable State Enforcement 14
  4.3. Capabilities Supporting UC3 14
    4.3.1. Tasking and Scheduling 14
    4.3.2. Data Aggregation and Reporting 15
5. Functional Components 16
  5.1. Asset Management 16
    5.1.1. Discovery 16
    5.1.2. Characterization 16
      5.1.2.1. Logical 16
      5.1.2.2. Security 16
    5.1.3. Asset Identification 16
  5.2. Security Configuration Management 16
    5.2.1. Configuration Assessment 16
      5.2.1.1. Non-technical Assessment 16
      5.2.1.2. Technical Assessment 17
  5.3. Vulnerability Management 17
    5.3.1. Non-technical Vulnerability Assessment 17
    5.3.2. Technical Vulnerabiltiy Assessment 17
  5.4. Content Management 17
    5.4.1. Control Frameworks 17

5.4.2. Configuration Standards 17
5.4.3. Scoring Models 17
5.4.4. Vulnerability Information 17
5.4.5. Patch Information 17
5.4.6. Asset Information 17
5.5. Assessment Result Analysis 17
5.5.1. Comparing Actual to Expected State 17
5.5.2. Scoring Comparison Results 17
5.5.3. Relating Comparison Results to Requirements 17
5.5.4. Relating Requirements to Control Frameworks 17
5.6. Tasking and Scheduling 17
5.6.1. Selection of Assessment Criteria 18
5.6.2. Defining In-scope Assets 18
5.6.3. Defining Periodic Assessments 18
5.6.4. Defining Assessment Triggers 18
5.7. Data Aggregation and Reporting 18
5.7.1. By Asset Characterization 18
5.7.2. By Assessment Criteria 18
5.7.3. By Control Framework 18
5.7.4. By Benchmark 18
5.7.5. By Ad Hoc/Extended Properties 18
6. Data Exchange Models and Communications Protocols 18
6.1. Data Exchange Models 19
6.1.1. Control Expression 19
6.1.1.1. Technical Control Expression 19
6.1.1.2. Non-technical Control Expression 19
6.1.2. Control Frameworks 19
6.1.2.1. Logical Expression and Syntactic Binding(s) 19
6.1.2.2. Relationships 19
6.1.2.3. Substantiation (Control Requirement) 19
6.1.2.4. Reporting 19
6.1.3. Asset Expressions 19
6.1.3.1. Asset Identification 19
6.1.3.2. Asset Classification (Type) 19
6.1.3.3. Asset Attributes 20
6.1.3.4. Information Expression (non-identifying) 20
6.1.3.5. Reporting 20
6.1.4. Benchmark/Checklist Expression 20
6.1.4.1. Logical Expression and Bindings 20
6.1.4.2. Checking Systems 20
6.1.4.3. Results and Scoring 20
6.1.4.4. Reporting 20
6.1.5. Check Language 20
6.1.5.1. Logical Expression and Syntactic Binding(s) 20
6.1.5.2. Reporting 20
6.1.6. Targeting Expression 20
6.1.6.1. Information Owner 20
6.1.6.2. System Owner 20

      6.1.6.3. Assessor 20
      6.1.6.4. Computing Device 20
      6.1.6.5. Targeting Extensibility 20
    6.2. Communication Protocols 21
      6.2.1. Asset Management Interface 21
  7. IANA Considerations 21
  8. Security Considerations 21
  9. Acknowledgements 21
  10. References 21
    10.1. Normative References 21
    10.2. Informative References 21
  Appendix A. Additional Stuff 22
  Authors' Addresses 22

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 a 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 systems. This document enumerates use cases and breaks down related concepts that cross many IT security information domains.

Sections Section 2, Section 3, Section 4, and Section 5 of this document respectively focus on:

Defining the key concepts and terminology 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;

A break down of architectural components that address one or more functional capabilities that can be used in various combinations 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 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.

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 standards and 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.

```
     +---------------------------------------+  +-------------+
     |                                       |  |             |
     |     Operational Risk Management       |  |             |
     |                                       |  |             |
     +---------------------------------------+  |             |
                                                |             |
     +---------------------------------------+  |             |
     |                                       |  |             |
     |     Information Risk Management       |  |   Policy    |
     |                                       |  |   Process   |
     +---------------------------------------+  |   Procedure |
                                                |             |
     +---------------------------------------+  |             |
     |                                       |  |             |
     |     Control Frameworks                |  |             |
     |                                       |  |             |
     +---------------------------------------+  |             |
                                                |             |
     +---------------------------------------+  |             |
     |                                       |  |             |
     |     Controls                          |  |             |
     |                                       |  |             |
     +---------------------------------------+  +-------------+
```

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. An Information Risk Management program is most likely a constituent of Operational Risk Management (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: System State Assessment, Enforcement of Acceptable State, Security Control Verification and Monitoring.

3.1. UC1: System State Assessment

3.1.1. Goal

Assess security state of a given system to be in compliance with enterprise standards and, therefore, ensure alignment with enterprise policy.

3.1.2. Main Success Scenario

1. Define target system to be assessed

2. Select acceptable state policies to apply to defined target

3. Collect actual state values from target

4. Compare actual state values collected from target with expected state values as expressed in acceptable state policies

3.1.3. Extensions

None.

3.2. UC2: Enforcement of Acceptable State

3.2.1. Goal

Allow or deny access to a desired resource based on system characteristics compliance with enterprise policy.

3.2.2. Main Success Scenario

1. An entity (user on a system or the system itself) requests access to a given resource (i.e. network connection)

2. Assessment of system state is achieved using Section 3.1

3. Based on assessment results (i.e. compliance level with enterprise policy)

    A. System is allowed access to requested resource, or

    B. System is denied access to requested resource

3.2.3. Extensions

None.

3.3. UC3: Security Control Verification and Monitoring

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 targets to be assessed.

2. Select acceptable state policies to apply to set of targets

3. Define assessment trigger based on either a

    A. Time period, or

    B. System/enterprise event.

4. Define result reporting/alerting criteria

5. Enable continuous assessment

3.3.3. Extensions

None.

4. Functional Capabilities

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

At the most basic level, an enterprise needing to satisfy UC1 and UC2 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

As described in Section Section 3.1, the required capabilities need to support assessing host and/or network state in an automated manner. This is, essentially, a configuration assessment check before allowing a full connection to the network.

4.1.1. Asset Management

Effective Asset Management is a critical foundation upon which all else in risk management is based. There are two important facets to asset managment: 1) understanding coverage (how many assets are under control) and, 2) understaning specific asset details. Coverage is fairly straightforward - assessing 80% of the enterprise is better than assessing 50% of the enterprise. 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 are considered suspect. Assessing assets (managed and unmanaged) requires that we see and properly characterize our assets at the outset and over time.

What we need to do initially is discover and characterize our assets, 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 (later described in Section 5):

- Discover assets in the enterprise
- Characterize assets according to security and non-security asset properties
- Identify and describe assets using a common vocabulary between implementations
- Reconcile asset representations originating from disparate tools

- Manage asset information throughout the asset's life cycle

4.1.2. Data Collection

Related to managing assets, and central to any automated assessment solution is the ability to collect data from target hosts (some might call this "harvesting"). Of particular interest are data representing the security state of a target, be it a computing device, network hardware, operating system, or application. The primary interest of the activities demanding data collection is centered on object state collection, where objects may be file attributes, operating system and/or application configuration items, and network device configuration items among others.

4.1.2.1. Security Configuration Management

There are many valid perspectives to take when considering required capabilities, but the industry seems to have roughly settled upon the notion of "Security Configuration Management" (there are variants of the term). Security Configuration Management (SCM) is a simple way to reference several supporting capabilities involving technical and non-technical assessment of systems.

The following capabilities support SCM:

- Target Assessment
  - Collect the state of non-technical controls commonly called administrative controls (i.e. policy, process, procedure)
  - Collect the state of technical controls including, but not necessarily limited to:
    - Target configuration items
    - Target patch level
    - Target object state

4.1.2.2. Vulnerability Management

SCM is only part of the solution, as it deals exclusively with the configuration of computing devices, including software vulnerabilities (by testing for patch levels). 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.

The following capabilities support Vulnerability Management:

1. Assessment
    - Non-technical Vulnerability Assessment (i.e. interrogative)
    - Technical Vulnerability Assessment

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. Such standards vary between enterprises, but commonly take a similar form.

The following capabilities support the analysis of assessment results:

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

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.

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:

- Configuration Standards
- Scoring Models
- Vulnerability Information
- Patch Information
- Asset Characterization

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

4.2. Capabilities Supporting UC2

UC2 is dependent upon UC1 and, therefore, includes all of the capabilities described in Section 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 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 [RFC3588].

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:

- Selection of assessment criteria

- Defining in-scope assets (i.e. targeting)

- Defining periodic assessments for a given set of tasks

- 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

- By asset characterization

- By assessment criteria

- By control framework

- By benchmark
- By other attributes/properties of assessment characteristics
- Extensible aggregation and reporting

5. Functional Components

This section describes the functional components alluded to in the previous section Section 4. In keeping with the organization of the previous section, the following high-level functional capabilities are decomposed herein: Asset Management, Security Configuration Management, Vulnerability Management, Content Management, Assessment Result Analysis, Tasking and Scheduling, and Data Aggregation and Reporting.

5.1. Asset Management

As previously mentioned, asset management is a critically important component of any risk management program. If you stop to consider the different tools used to support a risk management program (i.e. IDS/IPS, Firewalls, NAC devices, WAFs, SCM, and so on), they all need, to some degree, an element of asset management. In this context, asset management is defined as the maintenance of necessary and accurate asset characteristics. Management of assets requires the ability to discover, characterize, and subsequently identify assets across enterprise tools. The components described herein support Section 4.1.1

5.1.1. Discovery

5.1.2. Characterization

5.1.2.1. Logical

5.1.2.2. Security

5.1.3. Asset Identification

5.2. Security Configuration Management

The components described herein support Section 4.1.2

5.2.1. Configuration Assessment

5.2.1.1. Non-technical Assessment

5.2.1.2. Technical Assessment

5.2.1.2.1. Configuration Assessment

5.2.1.2.2. Patch Assessment

5.2.1.2.3. Object State Assessment

5.3. Vulnerability Management

The components described herein support Section 4.1.2

5.3.1. Non-technical Vulnerability Assessment

5.3.2. Technical Vulnerabiltiy Assessment

5.4. Content Management

The components described herein support Section 4.1.4

5.4.1. Control Frameworks

5.4.2. Configuration Standards

5.4.3. Scoring Models

5.4.4. Vulnerability Information

5.4.5. Patch Information

5.4.6. Asset Information

5.5. Assessment Result Analysis

The components described herein support Section 4.1.3

5.5.1. Comparing Actual to Expected State

5.5.2. Scoring Comparison Results

5.5.3. Relating Comparison Results to Requirements

5.5.4. Relating Requirements to Control Frameworks

5.6. Tasking and Scheduling

The components described herein support Section 4.3.1

5.6.1. Selection of Assessment Criteria

5.6.2. Defining In-scope Assets

5.6.3. Defining Periodic Assessments

5.6.4. Defining Assessment Triggers

5.7. Data Aggregation and Reporting

The components described herein support Section 4.3.2

5.7.1. By Asset Characterization

5.7.2. By Assessment Criteria

5.7.3. By Control Framework

5.7.4. By Benchmark

5.7.5. By Ad Hoc/Extended Properties

6. 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 event, incident and network management 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 be successful. 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 to support the described use cases.

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.

6.1. Data Exchange Models

The functional capabilities described in Section 4 require a significant number of models to be selected or defined in order to meet the needs of the three use cases presented in Section 3. 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.

6.1.1. Control Expression

For each we need an identification method, a logical expression and one or more syntactic bindings to that expression. For some, we may wish to associate a method of risk scoring.

6.1.1.1. Technical Control Expression

6.1.1.2. Non-technical Control Expression

6.1.1.2.1. Configuration Controls

6.1.1.2.2. Patches

6.1.1.2.3. Vulnerabilities

6.1.1.2.4. Object (Non-security) State

6.1.2. Control Frameworks

6.1.2.1. Logical Expression and Syntactic Binding(s)

6.1.2.2. Relationships

6.1.2.3. Substantiation (Control Requirement)

6.1.2.4. Reporting

6.1.3. Asset Expressions

6.1.3.1. Asset Identification

6.1.3.2. Asset Classification (Type)

6.1.3.3. Asset Attributes
6.1.3.3.1. Criticality
6.1.3.3.2. Classification (security)
6.1.3.3.3. Owner
6.1.3.4. Information Expression (non-identifying)
6.1.3.5. Reporting
6.1.4. Benchmark/Checklist Expression
6.1.4.1. Logical Expression and Bindings
6.1.4.2. Checking Systems
6.1.4.3. Results and Scoring
6.1.4.4. Reporting
6.1.5. Check Language
6.1.5.1. Logical Expression and Syntactic Binding(s)
6.1.5.1.1. Technical
6.1.5.1.2. Non-technical
6.1.5.2. Reporting
6.1.6. Targeting Expression
6.1.6.1. Information Owner
6.1.6.2. System Owner
6.1.6.2.1. Computing Device(s)
6.1.6.2.2. Network(s)
6.1.6.3. Assessor
6.1.6.4. Computing Device
6.1.6.5. Targeting Extensibility

6.2. Communication Protocols

6.2.1. Asset Management Interface

7. 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.

8. Security Considerations

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

9. 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 thier keen and insightful feedback on the concepts and text within this document.

10. References

10.1. Normative References

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

10.2. 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-02 (work in progress), May 2012.

[I-D.ietf-nea-pt-tls]
Sangster, P., Cam-Winget, N., and J. Salowey, "PT-TLS: A

TCP-based Posture Transport (PT) Protocol", draft-ietf-nea-pt-tls-05 (work in progress), May 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.

[RFC3588] Calhoun, P., Loughney, J., Guttman, E., Zorn, G., and J. Arkko, "Diameter Base Protocol", RFC 3588, September 2003.

[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.

Appendix A. Additional Stuff

This becomes an Appendix if needed.

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