

SACM
Internet-Draft
Intended status: Standards Track
Expires: September 8, 2016

D. Haynes
The MITRE Corporation
J. Fitzgerald-McKay
Department of Defense
L. Lorenzin
Pulse Secure
March 7, 2016

Endpoint Compliance Profile
draft-haynes-sacm-ecp-00

Abstract

This document specifies the Endpoint Compliance Profile, a high-level specification that describes a specific combination and application of NEA and TNC protocols and interfaces specifically designed to support ongoing assessment of endpoint posture and the controlled exposure of collected posture information to appropriate security applications. This document is a subset of the Trusted Computing Group's Endpoint Compliance Profile Version 1.0 specification.

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 September 8, 2016.

Copyright Notice

Copyright (c) 2016 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.	Preventative Posture Assessments	4
1.2.	Standardized Schema	4
1.3.	Secure Standardized Protocols	5
1.4.	Keywords	6
2.	Terminology	6
3.	Endpoint Compliance Profile	7
3.1.	Posture Assessments	7
3.2.	Data Storage	7
3.3.	Follow-up Actions	8
4.	Background	8
4.1.	Purpose of the Endpoint Compliance Profile	8
4.2.	Supported Use Cases	8
4.2.1.	Connected-and-Compliant	8
4.2.2.	Exposing Data to the Network	10
4.2.2.1.	Asset Management	12
4.2.2.2.	Vulnerability Searches	12
4.2.2.3.	Threat Detection and Analysis	12
4.2.3.	Non-supported Use Cases	12
4.2.4.	Profile Requirements	13
4.2.5.	Assumptions	14
5.	Endpoint Compliance Requirements	16
5.1.	Endpoint Pre-Provisioning	17
5.1.1.	SWID Tags	17
5.1.2.	Endpoint Identity and Machine Certificate	17
5.2.	Posture Validators and Posture Collectors	17
5.2.1.	SWID-Posture-Collectors-and-Posture-Validators	18
5.2.1.1.	The-SWID-Posture-Collector	18
5.2.1.2.	The SWID Posture Validator	18
5.3.	NEA Client (NEAC) and NEA Server (NEAS)	19
5.3.1.	NEAC	19
5.3.2.	NEAS	19
5.4.	Repository	19
6.	Posture Transport Client (PTC) and Posture Transport Server (PTS)	20
7.	Administrative Interface and API	20
8.	Endpoint Compliance Profile Examples	21
8.1.	Continuous Posture Assessment of an Endpoint	21
8.1.1.	Change on Endpoint Triggers Posture Assessment	22
8.2.	Administrator Searches for Vulnerable Endpoints	24

[9.](#) Acknowledgements [25](#)

[10.](#) IANA Considerations [27](#)

[11.](#) Security Considerations [27](#)

[11.1.](#) Security Benefits of Endpoint Compliance Profile [27](#)

[11.2.](#) Threat Model [29](#)

[11.2.1.](#) Endpoint Attacks [30](#)

[11.2.2.](#) Network Attacks [30](#)

[11.2.3.](#) Server Attacks [30](#)

[11.2.4.](#) Repository Attacks [31](#)

[11.3.](#) Countermeasures [31](#)

[11.3.1.](#) Countermeasures for Endpoint Attacks [31](#)

[11.3.2.](#) Countermeasures for Network Attacks [32](#)

[11.3.3.](#) Countermeasures for Server Attacks [32](#)

[11.3.4.](#) Countermeasures for Repository Attacks [33](#)

[12.](#) Privacy-Considerations [34](#)

[13.](#) References [34](#)

[13.1.](#) Informative References [34](#)

[13.2.](#) Normative References [34](#)

Authors' Addresses [35](#)

1. Introduction

The IETF NEA WG has defined an open architecture for network security, including standard protocols for endpoint posture assessment. The Endpoint Compliance Profile (ECP) builds on the NEA protocols, along with complementary interfaces from the Trusted Network Communications (TNC) WG of the Trusted Computing Group [[TNC](#)], to determine the posture of any type of endpoint on a network including user endpoints, servers, and infrastructure. The first generation of this specification focuses on reducing the security exposure of a network by confirming that all network-connected endpoints are:

- o known and authorized
- o running applications that are known and authorized
- o running applications that are patched and up-to-date; and,
- o applications with known vulnerabilities can be located and patched

When ECP is used, posture information is gathered by the NEA Client (NEAC) running on the endpoint and is forwarded to the NEA Server (NEAS), which stores it in a repository. This information is gathered while the endpoint is already connected to the network. Administrators will query the repository to determine the compliance status of an endpoint. For example, if a vulnerability is discovered in a product, an administrator may query the repository to determine

which endpoints have the vulnerable software installed and thus require some follow-up action.

Future versions of the ECP may want to address how to expose information--such as endpoint purpose, the software that is supposed to be running on an endpoint, and the activities an endpoint is supposed to be performing--to sensors that are looking for indicators of attacks and malicious activity on the network.

1.1. Preventative Posture Assessments

The value of continuous endpoint posture assessment is well established. Security experts have for years identified software updating and patching as a critical step for preventing intrusions. Application white listing, patching applications and operating systems, and using the latest versions of applications top the Defense Signals Directorate's "Top 4 Mitigations to Protect Your ICT System". [[DSD](#)] "Inventory of Authorized and Unauthorized Endpoints", "Inventory of Authorized and Unauthorized Software", and "Continuous Vulnerability Assessment and Remediation" are Critical Controls 1, 2, and 4, respectively, of the SANS "20 Critical Security Controls". [[SANS](#)] While there are commercially available solutions that attempt to address these security controls, these solutions do not run on all types of endpoints; consistently interoperate with other tools that could make use of the data collected; collect posture information from all types of endpoints in a consistent, standardized schema; or require vetted, standardized protocols that have been evaluated by the international community for cryptographic soundness.

As is true of most solutions offered today, the solution found in the ECP does not attempt to solve the lying endpoint problem. An endpoint that has already been infected with malicious software can provide false information about its identity and the software it is running. The primary purpose of the ECP is not to detect infected endpoints; rather, it focuses on ensuring that healthy endpoints remain healthy by keeping software up-to-date and patched. The first goal of the ECP is to help an administrator be able to readily determine which endpoints require some follow-up action. Future versions of the ECP may want to address how to expose posture information to sensors to aid the detection of attacks on endpoints and drive follow-up actions.

1.2. Standardized Schema

The ECP requires the use of standardized schema for the exchange of posture information. This helps to ensure that the posture information sent from endpoints to the repository can be easily stored, due to their known format, and shared with authorized

endpoints and users. Standardized schema also enable collection from myriad types of endpoints. Such standardization saves implementers time and money--time that does not have to be spent integrating new schema into the enterprise's reporting mechanisms, and money that does not have to be spent on developing tools to parse information from each type of endpoint connected to the network. Standardized schema also enable the development of standardized client software. This allows endpoint vendors to include their own client software that can interoperate with posture assessment infrastructure and thus not have to introduce third party code in their products.

1.3. Secure Standardized Protocols

Posture information must be sent over mature, standardized protocols to ensure the confidentiality and authenticity of this data while in transit. The ECP requires use of the NEA PT-TLS protocol [[RFC6876](#)] for communication between the endpoint and the server. This protocol allows networks that implement this solution to collect large amounts of posture information from an endpoint in order to make decisions about that endpoint's compliance to some policy. This Profile offers a solution for all endpoints already connected to the network. Periodic assessments and automated reporting of changes to installed software allow for instantaneous identification of connected endpoints that are no longer compliant to some policy.

The IETF NEA WG has designed an architecture to support endpoint posture assessment. Figure 1 illustrates the architectural components used in the Endpoint Compliance Profile:

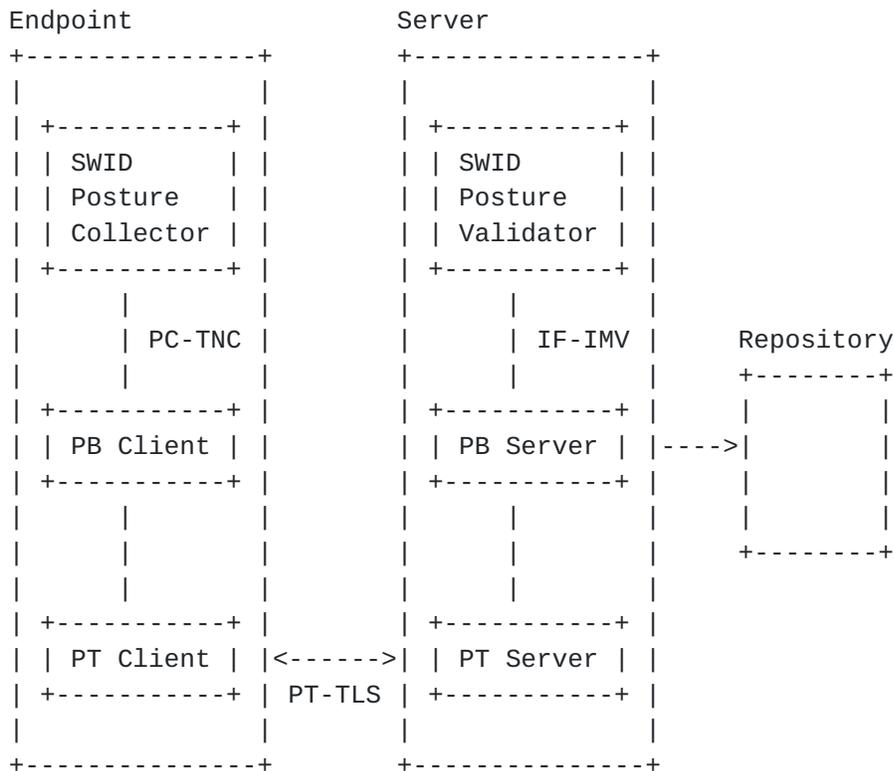


Figure 1: The Endpoint Compliance Architecture

Note that the SWID Posture Collector and SWID Posture Validator are implementations of NEA's Posture Collector (PC) and Posture Validator (PV) architectural components, respectively. Requirements for each of the components in the diagram above are contained in this profile. The reader should consult [RFC5209] for additional information on these components. All current repository requirements are contained within the Endpoint Compliance Profile.

1.4. Keywords

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 [RFC2119]. This specification does not distinguish blocks of informative comments and normative requirements. Therefore, for the sake of clarity, note that lower case instances of must, should, etc. do not indicate normative requirements.

2. Terminology

This document uses terms as defined in [I-D.ietf-sacm-terminology] unless otherwise specified.

3. Endpoint Compliance Profile

The Endpoint Compliance Profile describes how NEA and TNC specifications can be used to support the posture assessment of endpoints on a network. This profile does not generate new schema or protocols; rather, it offers a full end-to-end solution for posture assessment, as well as a fresh perspective on how existing standards can be leveraged against vulnerabilities.

3.1. Posture Assessments

The Endpoint Compliance Profile 1.0 describes how NEA and TNC specifications make it possible to perform posture assessments against all network-connected endpoints by:

1. uniquely identifying the endpoint;
2. collecting and assessing posture based on data from the endpoint;
3. creating a secure, authenticated, confidential channel between the endpoint and the server;
4. enabling the endpoint to notify the server about changes to its configuration;
5. enabling the server to request information about the configuration of the endpoint; and
6. storing the posture information in a repository linked to the identifier for the endpoint.

3.2. Data Storage

The ISO/IEC Software Identification Tag standard [[SWID](#)] has defined a schema for identifying applications installed on endpoints and their patch status. The Endpoint Compliance Profile 1.0 focuses on being able to collect this information from an endpoint and store it in a repository. This makes posture information from a network's endpoints available to authorized parties. Uses of this data are innumerable--vulnerability management, asset management, software asset management, and configuration management solutions, analytics tools, endpoints that need to make connectivity decisions, and metrics reporting scripts, among others, are all able to reference the data stored in the repository to achieve their purposes.

3.3. Follow-up Actions

The ability of the endpoint to notify the server whenever a modification is made to the endpoint enables immediate identification of endpoints that fall out of compliance. The Endpoint Compliance Profile 1.0 does not specify requirements for how these endpoints should be addressed. However, the TNC specifications do support the ability to send instructions that drive access control enforcement decisions for a non-compliant endpoint. Additional information about the types of follow-up actions an enterprise may want to support can be found in [[RFC7632](#)].

There is a clear need for nuanced, automated instructions sent from the server to the endpoint (for example, to update an endpoint's software, or remove a piece of non-compliant software). Those messages are complicated to define and may have to be tailored to a particular operating system. Future versions of this specification may want to address which instructions can be defined based on the configuration content that is collected from endpoints.

4. Background

4.1. Purpose of the Endpoint Compliance Profile

The Endpoint Compliance Profile describes a standard way to communicate endpoint posture information such as software identity and software version and to make it available to other authorized parties. The Endpoint Compliance Profile 1.0 focuses on collecting the application information available in SWID tags, as specified in [[SWID](#)]. Future versions of the Endpoint Compliance Profile could describe how additional types of posture information can be collected and communicated in a standardized way.

4.2. Supported Use Cases

The Endpoint Compliance Profile focuses on the posture assessment of enterprise endpoints on enterprise networks. Use cases supported by the Endpoint Compliance Profile 1.0 are as follows:

4.2.1. Connected-and-Compliant

A network-connected endpoint sends posture information using standard schemas such as SWID over NEA protocols.

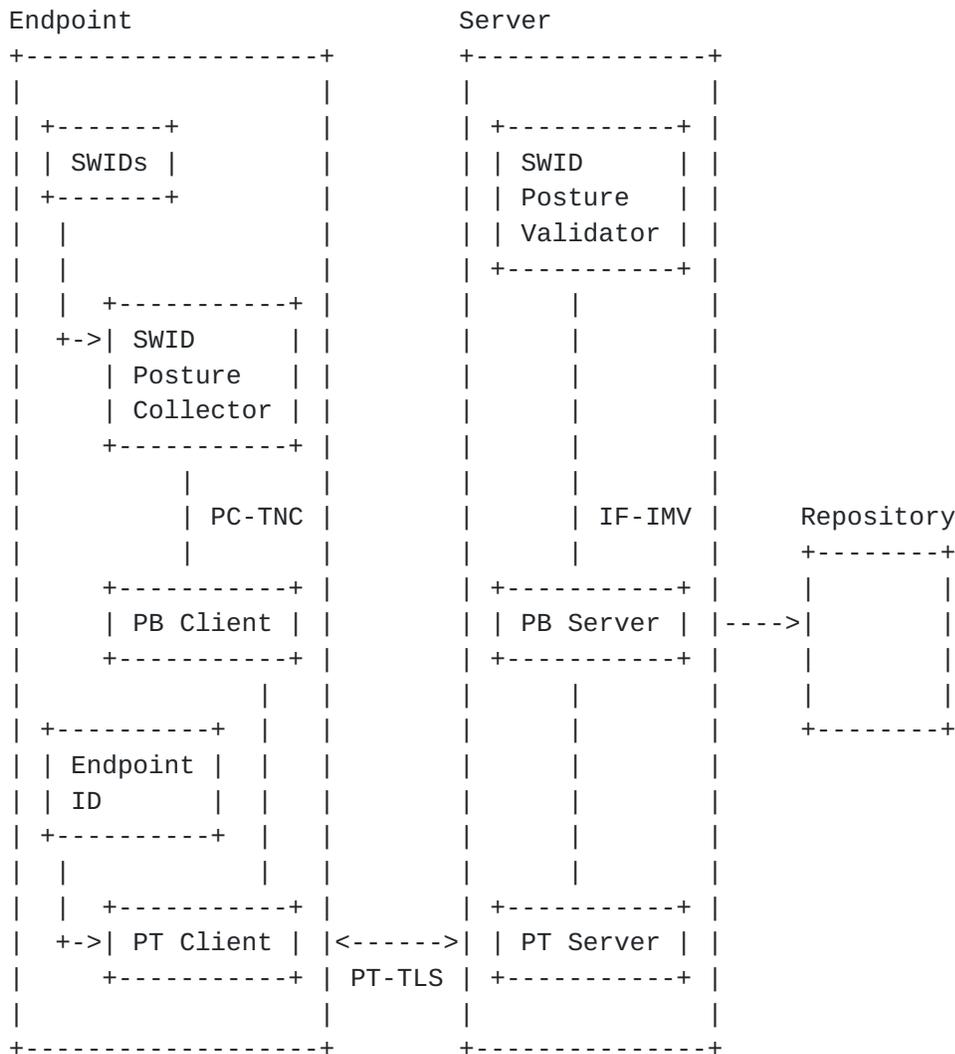


Figure 2: Connected and Compliant Use Case

1. If necessary, the endpoint finds and validates the server in compliance with [[Server-Discovery](#)].
2. The Posture Transport Client (PTC) on the endpoint and Posture Transport Server (PTS) on the server complete a TLS handshake, during which endpoint identity information is exchanged.
3. Either the NEA Server (NEAS) on the server or the NEA Client (NEAC) on the endpoint initiates a posture assessment. Checks may be triggered for multiple reasons, including:
 - (a) policy states that a previous assessment has aged out and become invalid;

- (b) the NEAC notices that the relevant posture information on the endpoint has changed, (for example, due to application updates, deletions or additions); or
- (c) the NEAS is alerted by a sensor or an administrator (via the server's user interface) that an assessment must be completed.

All information exchanges between the PCs and PVs are subject to the enterprise's policy, which may limit the content or size of information sent between the endpoint and the server.

4. The SWID Posture Collector on the endpoint collects from the SWID tag directory on the endpoint. This data is sent via the NEAC and PTC to the server.
5. Once the posture information is received by the PTS, it is forwarded to the SWID Posture Validator via the NEAS. The SWID Posture Validator also forwards the posture information to the repository. The posture information is stored along with past posture information collected about the endpoint.

4.2.2. Exposing Data to the Network

Because the endpoint posture information was sent in a standards-based schema (ISO/IEC 19770-2:2009) over secure, standardized protocols, and the SWID tags are stored in a centralized repository linked to unique endpoint identifiers, authorized parties are able to access the posture information. Such authorized parties may include, but are not limited to, administrators or endpoint owners (via the server's administrative interface), and other pieces of infrastructure that can make use of this data (via the server's API). The server will provide:

- o a standard administrative interface that allows data sharing with authorized parties;
- o a standard API that allows data sharing with authorized infrastructure and software;
- o a persistent account of endpoints that have connected to the network over a period of time set by the administrator;
- o the identities provided by those endpoints; and
- o what SWIDs were reported by the endpoint.

The endpoint will publish updates as its local SWID directory changes, as well as each time it disconnects and reconnects to the network.

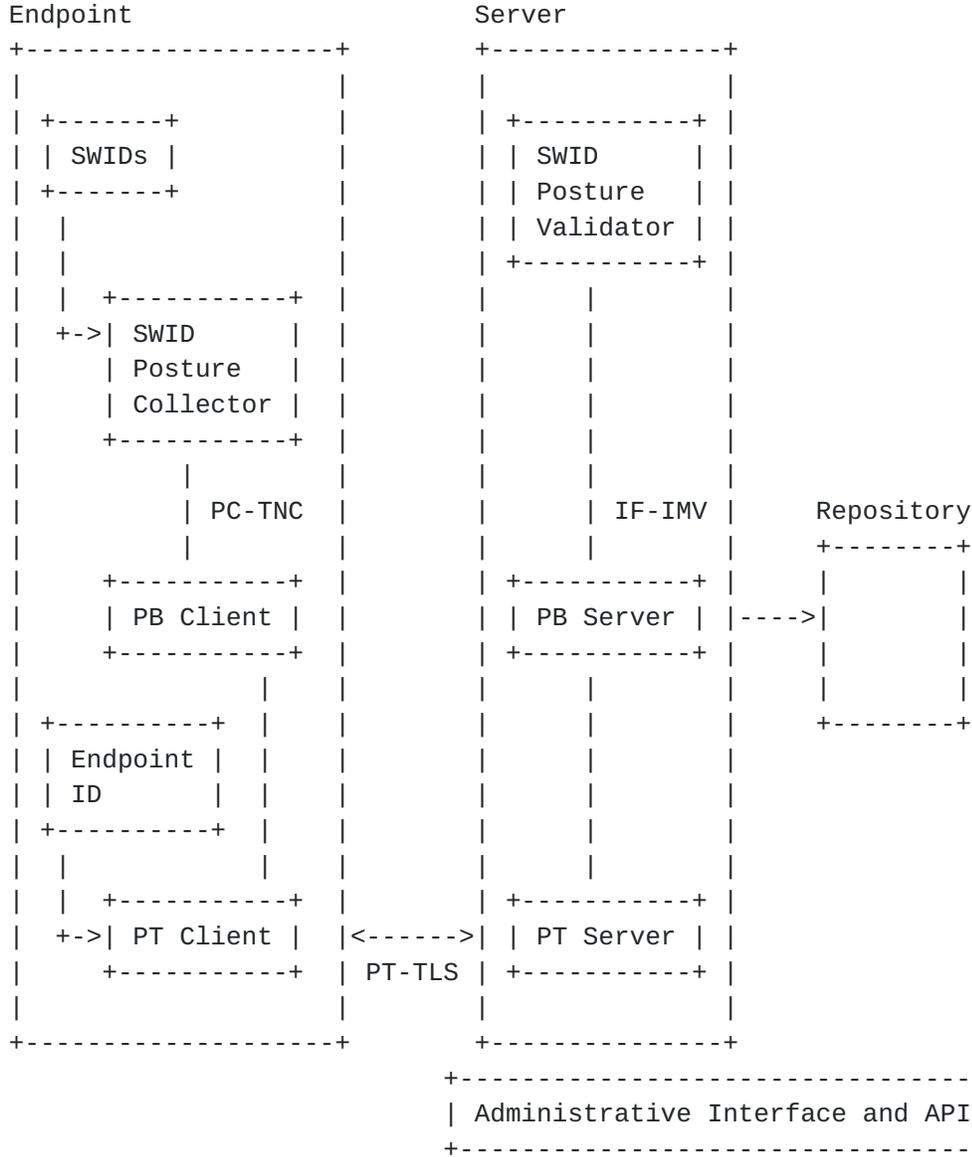


Figure 3: Exposing Data to the Network

It should be noted that the neither the Endpoint Compliance Profile nor the protocols, interfaces, and data models that it references provide solutions to the server capabilities listed above. However, these capabilities are useful and solutions for them should be pursued in the future.

4.2.2.1. Asset Management

Using the administrative interface on the server, an authorized user can learn:

- o what endpoints are connected to the network at any given time; and
- o what SWID tags were reported for the endpoints.

The ability to answer these questions offers a standards-based approach to asset management, which is a vital part of enterprise processes such as compliance report generation for the Federal Information Security Modernization Act (FISMA), Payment Card Industry Data Security Standard (PCI DSS), Health Insurance Portability and Accountability Act (HIPAA), etc.

4.2.2.2. Vulnerability Searches

The administrative interface also provides the ability for authorized users or infrastructure to locate endpoints running software for which vulnerabilities have been announced. Because of

1. the unique IDs assigned to each endpoint; and
2. the rich application data provided in the endpoints' posture information,

the repository can be queried to find all endpoints running a vulnerable application. Endpoints suspected of being vulnerable can be addressed by the administrator or flagged for further scrutiny.

4.2.2.3. Threat Detection and Analysis

The repository's standardized API allows authorized infrastructure endpoints and software to search endpoint posture assessment information for evidence that an endpoint's software inventory has changed, and can make endpoint software inventory data available to other endpoints. This automates security data sharing in a way that expedites the correlation of relevant network data, allowing administrators and infrastructure endpoints to identify odd endpoint behavior and configuration using secure, standards-based schema and protocols.

4.2.3. Non-supported Use Cases

Several use cases, including but not limited to these, are not covered by the Endpoint Compliance Profile 1.0:

- o Gathering other types of posture information: The Endpoint Compliance Profile does not prevent administrators from collecting other types of posture information other than SWIDs from the endpoint; however it does not set requirements for doing so.
- o Solving the lying endpoint problem: The Endpoint Compliance Profile does not address the lying endpoint problem; the Profile makes no assertions that it can catch an endpoint that is, either maliciously or accidentally, reporting false posture information to the server. However, other solutions may be able to use the posture information collected using the capabilities described in this profile to catch an endpoint in a lie. For example, a sensor may be able to compare the posture information it has collected on an endpoint's activity on the network to what the endpoint reported to the server and flag discrepancies. However, these particular capabilities are not described in this profile.
- o Publish/subscribe repository interface: Future versions of the Endpoint Compliance Profile may specify a publish/subscribe interface for the repository, so infrastructure endpoint can subscribe to and receive published posture assessment results from the repository regarding endpoint configuration changes. However, the Endpoint Compliance Profile 1.0 includes no such requirements.

4.2.4. Profile Requirements

Here are the requirements that the Endpoint Compliance Profile protocol must meet in order to successfully fit in the SACM architecture.

- o Meets the needs of the SACM architecture: The Endpoint Compliance Profile must support the use cases described in [[RFC7632](#)] as they apply to endpoint self-reporting and endpoint posture assessment.
- o Efficient: To minimize user frustration, it is essential to minimize delays by making endpoint posture information collection, transmission, and assessment as brief and efficient as possible.
- o Extensible: The Endpoint Compliance Profile needs to expand over time as new features are added to the SACM architecture. The solution must allow new features to be added easily, providing for a smooth transition and allowing newer and older architectural components to continue to work together. Further, the Endpoint Compliance Profile and the specifications referenced here must define safe extensibility mechanisms that enable innovation without breaking interoperability.

- o Easy to implement: The Endpoint Compliance Profile should be easy for vendors to implement in their products, and should result in products that are easy for administrators to implement on their networks. Products conformant to the Endpoint Compliance Profile should interoperate seamlessly, and be simple to integrate into existing network infrastructure.
- o Easy to use: The Endpoint Compliance Profile should describe a simple, integrated user interface that administrators can use to perform the activities listed in the profile's use cases. The Endpoint Compliance Profile should not constrain innovation by specifying details of the user interface but rather functional requirements.
- o Platform-independent: Since network environments may contain many different types of endpoints, the solution should operate independently of the endpoint platform.
- o Scalable: The Endpoint Compliance Profile must be designed to scale to very large numbers of endpoints.

4.2.5. Assumptions

Here are the assumptions that the Endpoint Compliance Profile makes about other components in the SACM architecture.

- o Existence of a server and repository: The Endpoint Compliance Profile assumes that a server and repository exist.
- o Endpoint SWID installation: The Endpoint Compliance Profile assumes that an endpoint has been pre-provisioned with Software Identification Tags for its applications, and that these SWID tags are formatted and stored in conformance with [\[SWID\]](#).
- o Certificate provisioning: In order to implement the most secure endpoint identification option, the Endpoint Compliance Profile assumes that the enterprise has set up a certificate root authority, and has provisioned each endpoint with an endpoint identification certificate. This is not required if an enterprise chooses to use other endpoint authentication methods.

In addition, the Endpoint Compliance Profile makes the following assumptions about the SACM ecosystem:

- o All network-connected endpoints are endpoints: As defined by [\[I-D.ietf-sacm-terminology\]](#), an endpoint is any physical or virtual computing endpoint that can be connected to a network. Posture assessment against policy is equally, if not more,

important for continuously connected endpoints, such as enterprise workstations and infrastructure endpoints, as it is for sporadically connected endpoints. Continuously connected endpoints are just as likely to fall out of compliance with policy, and a standardized posture assessment method is necessary to ensure they can be properly handled.

- o All endpoints on the network must be uniquely identified: Many administrators struggle to identify what endpoints are connected at any given time. By requiring a standardized method of endpoint identity, the Endpoint Compliance Profile will enable administrators to answer the basic question, "What is on my network?" Unique endpoint identification also enables the comparison of current and past endpoint posture assessments, by allowing administrators to correlate assessments from the same endpoint. This makes it easier to flag suspicious changes in endpoint posture for manual or automatic review, and helps to swiftly identify malicious changes to endpoint applications.
- o Posture assessments must occur over secure, standardized protocols: Endpoint identity and application information is very valuable, both to administrators and to attackers. Therefore, it must be kept confidential, using secure protocols to transport it from the endpoint to network infrastructure endpoints. Additionally, it is critical that only authorized parties be capable of requesting information, receiving information, or taking action to change an endpoint's connectivity status. Relying on standardized protocols to provide this security enables greater interoperability and compatibility between endpoints, and allows for the development of compliance testing to ensure that each endpoint operates securely and in conformance with appropriate specifications. A standards body provides a process for experts in protocols and cryptography to evaluate the soundness of protocols and security management procedures; a set of security standards allows an enterprise to make the most effective use of their investment in a security management infrastructure.
- o Posture assessment results must be formatted using standardized schema: Well-known, standard schema allow for a universal language for generating compliance reports. With each endpoint speaking the same language, the Endpoint Compliance Profile enables information sharing between user endpoints and infrastructure endpoints, and between infrastructure endpoints that perform different security tasks.
- o Posture information must be stored by the repository and must be exposed to an interface at the server: A standard schema enables

standard queries from an interface exposed to an administrator at the server console. A repository must retain any current posture information retrieved from the endpoint and store it indexed by the unique identifier for the endpoint. Any PV specified by this profile must be able to ascertain from its corresponding PC whether the posture information is up to date. An interface on the server must support a request to the PV to obtain up-to-date information when an endpoint is connected. This interface must also support the ability to make a standard set of queries about the posture information stored by the repository. In the future, some forms of posture information might be retained at the endpoint. The interface on the server must accommodate the ability to make a request through the PV to the corresponding PC about the posture of the endpoint. Standard schema and protocols also enable the security of posture assessment results. By storing these results indexed under the endpoint's unique identification, secure storage itself enables endpoint posture information correlation, and ensures that the enterprise's Repositories always offer the freshest, most up-to-date view of the enterprise's endpoint posture information possible.

- o Posture information can be shared: By exposing posture information using a standard interface and API, other security and operational components have a high level of insight into the enterprise's endpoints and the software installed on them. This will support innovation in the areas of asset management, vulnerability scanning, and administrative interfaces, as any authorized infrastructure endpoint can interact with the posture information.
- o Owners and administrators must have complete control of posture information, policy, and endpoint mitigation: Enterprise asset posture information belongs to the enterprise. Standardized schema, protocols and interfaces help to ensure that this posture information is not locked in proprietary databases, but is made available to its owners. This enables administrators to develop as nuanced a policy as necessary to keep their networks secure.

5. Endpoint Compliance Requirements

These requirements are written with a view to performing a posture assessment on an endpoint; as the Endpoint Compliance Profile grows and evolves, these requirements will be expanded to address issues that arise. Note that these requirements refer to defined components of the NEA architecture. As with the NEA architecture, implementers have discretion as to how these NEA components map to separate pieces of software or endpoints.

5.1. Endpoint Pre-Provisioning

The following requirements assume that the platform or OS vendor supports the use of SWID tags and has identified a standard directory location for the SWID tags to be located as specified by [[SWID](#)].

5.1.1. SWID Tags

The primary content for the Endpoint Compliance Profile 1.0 is the information conveyed in the elements of a SWID tag.

The endpoint **MUST** have SWID tags stored in a directory specified in [[SWID](#)]. The tags **SHOULD** be provided by the software vendor; they **MAY** also be generated by:

- o the software installer; or
- o third-party software that creates tags based on the applications it sees installed on the endpoint.

The elements in the SWID tag **MUST** be populated as specified in [[SWID](#)]. These tags, and the directory in which they are stored, **MUST** be updated as software is added, removed, or updated.

5.1.2. Endpoint Identity and Machine Certificate

The endpoint **SHOULD** authenticate to the server using a machine certificate during the establishment of the outer tunnel achieved with PT. [[IF-IMV](#)] specifies how to pull an endpoint ID out of a machine certificate. An endpoint ID **SHOULD** be created in conformance with [[IF-IMV](#)] from a machine certificate sent via [[RFC6876](#)].

In the future, the identity could be a hardware certificate compliant with [[IEEE-802-1ar](#)]; ideally, this ID **SHOULD** be associated with the identity of a hardware cryptographic module, in accordance with [[IEEE-802-1ar](#)], if present on the endpoint. The enterprise **SHOULD** stand up a certificate root authority; install its root certificate on endpoints and on the server; and provision the endpoints and the server with machine certificates. The endpoint **MAY** authenticate to the server using a combination of the machine account and password; however, this is less secure and not recommended.

5.2. Posture Validators and Posture Collectors

Any PC used in an Endpoint Compliance Profile solution **MUST** be conformant with PC-TNC; an Internet-Draft, under development, that is a subset of the TCG TNC Integrity Measurement Collector interface [[IF-IMC](#)] and will be submitted in the near future. Any Posture

Validator used in an Endpoint Compliance Profile solution MUST be conformant with [[IF-IMV](#)].

5.2.1. SWID-Posture-Collectors-and-Posture-Validators

5.2.1.1. The-SWID-Posture-Collector

For the Endpoint Compliance Profile, the SWID Posture Collector MUST be conformant with [SWID-Message-for-PA-TNC], which includes requirements for:

1. Collecting SWID tags from the SWID directory
2. Monitoring the SWID directory for changes
3. Initiating a session with the server to report changes to the directory
4. Maintaining a list of changes to the SWID directory when updates take place and no PT-TLS connection can be created with the server
5. Responding to a request for SWID tags from the SWID Posture Validator on the server
6. Responding to a query from the SWID Posture Validator as to whether all updates have been sent

The SWID Posture Collector is not responsible for detecting that the SWID directory was not updated when an application was either installed or uninstalled.

5.2.1.2. The SWID Posture Validator

Conformance to [SWID-Message-for-PA-TNC] enables the SWID Posture Validator to:

1. Send messages to the SWID Posture Collector (at the behest of the administrator at the server console) requesting updates for SWID tags located on endpoint
2. Ask the SWID Posture Collector whether all updates to the SWID directory located at the server have been sent
3. Compare an endpoint's SWID posture information to policy, and make a recommendation to the NEAS about the endpoint

In addition to these requirements, a SWID Posture Validator used in conformance with this profile MUST be capable of passing information from the posture assessment results and the endpoint identity associated with those results to the repository for storage.

5.3. NEA Client (NEAC) and NEA Server (NEAS)

[RFC5793] describes a standard way for the NEAC and the NEAS to exchange messages.

5.3.1. NEAC

The NEAC MUST conform to [RFC5793], which levies a number of requirements against the NEAC. A NEAC that complies with these requirements will be able to:

1. attempt to initiate a session with the NEAS if the SWID Posture Collector makes a request to send an update to the SWID directory to the server;
2. notify the SWID Posture Collector if no PT-TLS session with the server can be created;
3. notify the SWID Posture Collector when a PT-TLS session is established; and
4. receive information from the PCs, forward this information to the server via the PTC.

The NEAC MUST also conform to PC-TNC to enable communications with the SWID Posture Collector.

5.3.2. NEAS

The NEAS MUST conform to all requirements in the [RFC5793] and [IF-IMV] specifications. Conformance to [IF-IMV] enables the NEAS to obtain endpoint identity information from the PTS, and pass this information to any IMVs on the server.

5.4. Repository

ECP 1.0 requires a simple administrative interface for the repository. PVs on the server receive the endpoint data via PA-TNC [RFC5792] messages sent from corresponding PCs on an endpoint and store this information in the repository linked to the identity of the endpoint where the PCs are located.

The administrative interface SHOULD enable an administrator to:

1. Query which endpoints have reported SWID tags for a particular application
2. Query which SWID tags are installed on a particular endpoint
3. Query tags based on characteristics, such as vendor, publisher, etc.

In the future, if SACM decides to develop an interface to the repository server, it should consider requirements for:

1. Creating a secure channel between a publisher and the repository
2. Creating a secure channel between a subscriber and the repository
3. The types of interactions that must be supported between publishers and subscribers to a repository

6. Posture Transport Client (PTC) and Posture Transport Server (PTS)

The PT-TLS protocol provides a transport service for carrying the PB-TNC protocol messages between the endpoint and the server.

The PTC and PTS MUST implement PT-TLS, since a connection is needed that:

- o Can handle large volumes of data, which might require multiple roundtrips, to be sent while the endpoint is connected
- o Allows either the NEAC or NEAS to initiate a connection
- o Supports secure transport based on machine certificates at both ends of the connection

The PTC and PTS MUST support the use of machine certificates for TLS at each endpoint consistent with the requirements stipulated in [[RFC6876](#)] and [[Server-Discovery](#)].

The PTC MUST be able to locate an authorized server, and switch to a new server when required by the network, in conformance with [[Server-Discovery](#)].

7. Administrative Interface and API

An interface is necessary to allow administrators to manage the endpoints and software used in the Endpoint Compliance Profile. This interface SHOULD be accessible either on or through (as in the case

of a remotely hosted interface) the server. Using this interface, an authorized user or administrator SHOULD be able to:

- o Query the repository
- o Send commands to the PVs, requesting information from the associated PCs residing on network endpoints
- o Update the policy that resides on the server

An API is necessary to allow infrastructure endpoints and software access to the information stored in the repository. Using this API, an authorized endpoint SHOULD be able to:

- o Query the repository

8. Endpoint Compliance Profile Examples

8.1. Continuous Posture Assessment of an Endpoint

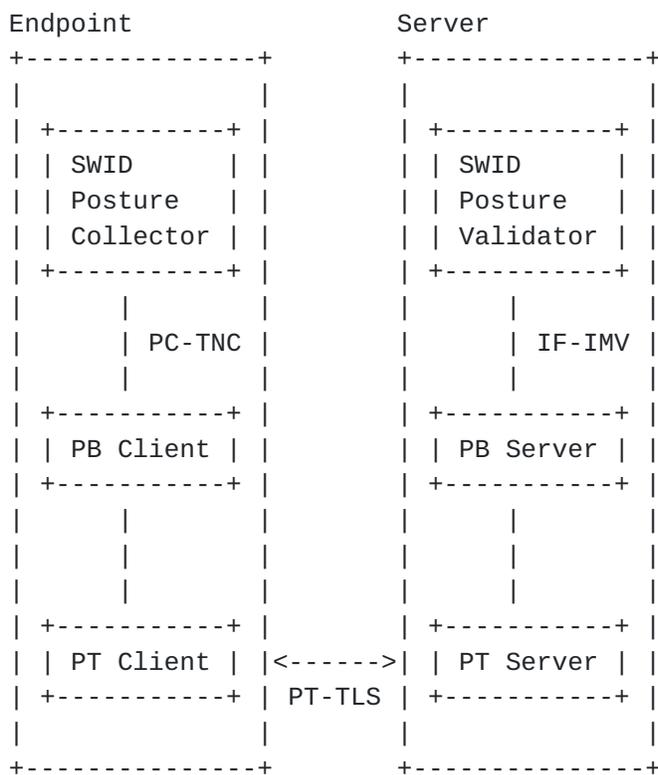


Figure 4: Continuous Posture Assessment of an Endpoint

8.1.1.1. Change on Endpoint Triggers Posture Assessment

A new application is installed on the endpoint, and the SWID directory is updated. This triggers an update from the SWID Posture Collector to the SWID Posture Validator. The message is sent down the NEA stack, encapsulated by NEA protocols until it is sent by the PTC to the PTS. The PTS then forwards it up through the stack, where the layers of encapsulation are removed until the SWID Message arrives at the SWID Posture Validator.

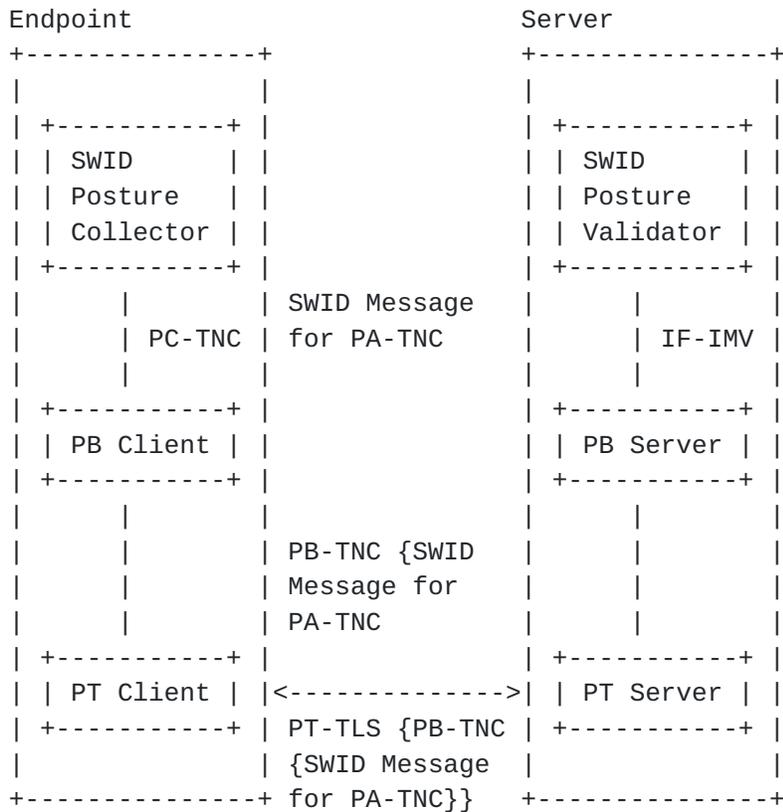


Figure 5: Compliance Protocol Encapsulation

The SWID Posture Validator stores the new tag information in the repository. If the tag indicates that the endpoint is compliant to the policy, then the process is complete until the next time an update is needed (either because policy states that the endpoint must submit posture assessment results periodically or because an install/uninstall/update on the endpoint triggers a posture assessment).

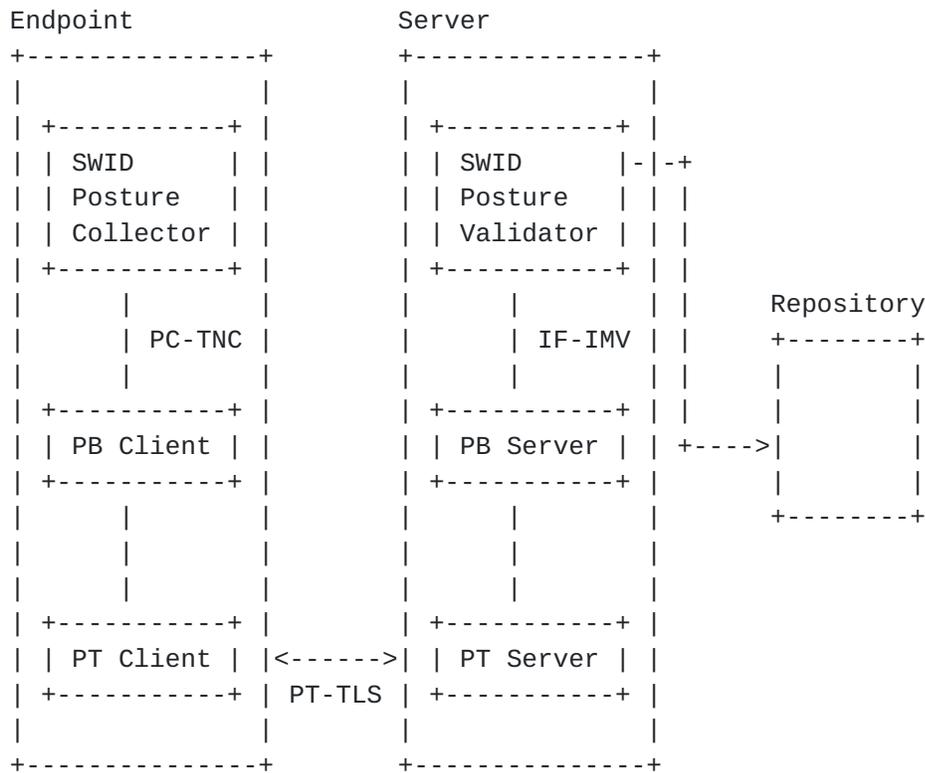


Figure 6: Storing SWIDs in the Repository

If the endpoint has fallen out of compliance with a policy, the server can alert the administrator via the server's administrative interface. The administrator can then take steps to address the problem. If the administrator has already established a policy for automatically addressing this problem, that policy will be followed.

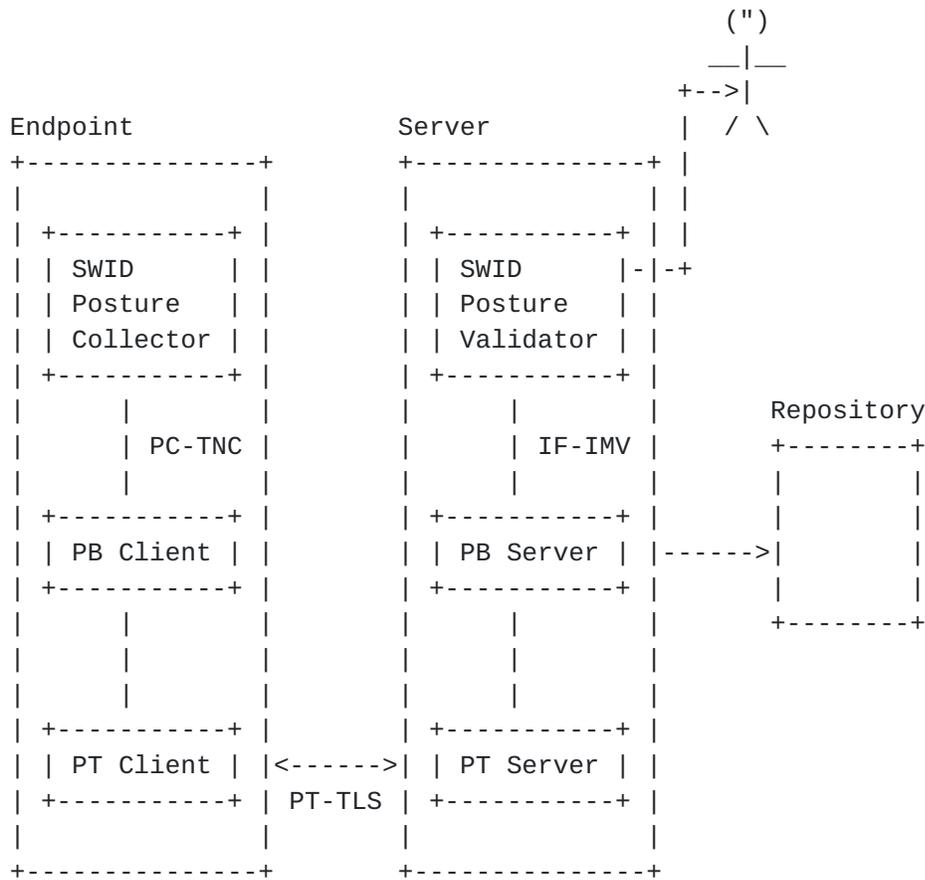


Figure 8: Admin Searches for Vulnerable Endpoints

The repository returns a list of entries in the matching the administrator's search. The administrator can then address the vulnerable endpoints by taking some follow-up action such as removing it from the network, quarantining it, or updating the vulnerable software.

9. Acknowledgements

The authors wish to thank all of those in the TCG TNC work group who contributed to development of the TNC ECP specification upon which this document is based.

Member	Organization
Padma Krishnaswamy	Battelle Memorial Institute
Eric Fleischman	Boeing
Richard Hill	Boeing

Steven Venema	Boeing
Nancy Cam-Winget	Cisco Systems
Scott Pope	Cisco Systems
Max Pritikin	Cisco Systems
Allan Thompson	Cisco Systems
Nicolai Kuntze	Fraunhofer Institute for Secure Information Technology (SIT)
Ira McDonald	High North
Dr. Andreas Steffen	HSR University of Applied Sciences Rapperswil
Josef von Helden	Hochschule Hannover
James Tan	Infoblox
Steve Hanna (TNC-WG Co-Chair)	Juniper Networks
Cliff Kahn	Juniper Networks
Lisa Lorenzin	Juniper Networks
Atul Shah (TNC-WG Co-Chair)	Microsoft
Jon Baker	MITRE
Charles Schmidt	MITRE
Rainer Enders	NCP Engineering
Dick Wilkins	Phoenix Technologies
David Waltermire	NIST
Mike Boyle	U.S. Government
Emily Doll	U.S. Government
Jessica Fitzgerald-	U.S. Government

McKay		
Mary Lessels	U.S. Government	
Chris Salter	U.S. Government	
+-----+	+-----+	+-----+

Table 1: Members of the TNC that Contributed to the Document

10. IANA Considerations

This document does not define any new IANA registries. However, this document does reference other documents that do define IANA registries. As a result, the IANA Considerations section of the referenced documents should be consulted.

11. Security Considerations

The Endpoint Compliance Profile offers substantial improvements in endpoint security, as evidenced by the Australian Defense Signals Directorate's analysis that 85% of targeted cyber intrusions can be prevented through application white listing, patching applications and operating systems, and using the latest versions of applications. [DSD] Despite these gains, some security risks continue to exist and must be considered.

To ensure that these benefits and risks are properly understood, this Security Considerations section includes an analysis of the benefits provided by the Endpoint Compliance Profile (Section 11.1), the attacks that may be mounted against systems that implement the Endpoint Compliance Profile (Section 11.2), and the countermeasures that may be used to prevent or mitigate these attacks (Section 11.3). Overall, a substantial reduction in cyber risk can be achieved.

11.1. Security Benefits of Endpoint Compliance Profile

Security weaknesses of the components for this profile should be considered in light of the practical considerations that must be addressed to have a viable solution.

Posture assessment has two parts: assessment and follow-up actions. The point of posture assessment is to ensure that authorized users are using authorized software configured to be as resilient as possible against an attack.

Posture assessment answers the question whether the endpoint is healthy. Our goal for posture assessment is to make it harder for an adversary to execute code on one of our endpoints. This profile

represents an important first step in reaching that goal. If we keep our endpoints healthier, we are able to prevent more attacks on our endpoints and thus on our information systems.

The goal of ECP is to address posture assessment in stages. Stage 1 is the ability to ascertain whether all endpoints are authorized and whether all applications are authorized and up to date. Stage 2 will attempt to address the harder problem of whether all software is configured safely. Eventually, the goal is to also address remediation which is currently out-of-scope for the SACM WG; that presents a far greater security challenge than reporting, since remediation implies the ability of a remote party to modify software or its settings on endpoints.

A second security consideration is how to gain visibility over every type of endpoint and every piece of software installed on the endpoint. This is a problem of scaling and observation. A solution is needed that can report from every type of endpoint. All software on the endpoint has to be discovered. Information about the software has to be up to date and accurate. The information that is discovered has to be reported in a consistent format, so administrators do not have to squander time deciphering proprietary systems and the information can be made readily useful for other security automation purposes.

ECP is based on a model of a standards-based schema, a standards-based set of protocols and interfaces, and the existence of an oversight group, the IETF, that can update the schemas and protocols to meet new use cases and security issues that may be discovered.

The data elements in the schema determine what work can be done consistently for every endpoint and every piece of software. How the data gets populated is an important consideration. ECP leverages the SWID tags from ISO 19770-2 because the tag originates with a single authoritative source, the application vendor itself. Moreover, there is a natural incentive for the vendor to create this content, since it makes it easier for enterprises and vendors to track whether software is licensed. Practical considerations are security considerations. A sustainable business model for obtaining all the necessary content is a fundamental requirement.

The NEA model is based on having a NEAC run on an endpoint that publishes posture information to a server. The advantages are easy to list. A platform vendor can implement its own NEAC and have it be compatible with the NEAS from a different vendor. The interfaces are layered on top of mature protocols such as TLS. TLS is the protocol of choice for ECP, since:

- o it has proven secure properties,
- o it can be implemented on most types of endpoints,
- o it allows the gathering of large amounts of information when a endpoint is connected, and
- o it enables use of a mechanism to ensure that the client is authenticated (authorized) - a client certificate - which also provides a consistent identifier.

Mature protocols that can be implemented on most types of endpoints and a standards-based schema with a sustainable business model are both critical security considerations for compliance.

Additionally, it is important to consider the future stages for ECP such as a posture assessment being followed up by some action (e.g. remediation, alert, etc.). Ensuring that clients are taking instructions only from authorized parties will be critical. Inasmuch as it is practical, enterprises will want to use the same infrastructure and investment in PKI to send those instructions to a client.

Likewise, as more information with more value is gathered from endpoints, we will also want to ensure that this information is only released to authorized applications and parties. For the next stage of ECP, SACM may want to define an interface on the repository that can be queried by other security automation applications to make it easier to detect attacks and for other security automation applications. This interface has to be standards-based for enterprises to reap the benefits of innovation that can be achieved by making the enterprise's data available to other tools and services.

11.2. Threat Model

This section lists the attacks that can be mounted on an Endpoint Compliance Profile environment. The following section ([Section 11.3](#)) describes countermeasures.

Because the Endpoint Compliance Profile describes a specific use case for NEA components, many security considerations for these components are addressed in more detail in the technical specifications: [SWID-Message-for-PA-TNC], PC-TNC, [[RFC5793](#)], [[Server-Discovery](#)], [[RFC6876](#)], [[IF-IMV](#)].

11.2.1. Endpoint Attacks

While the Endpoint Compliance Profile provides substantial improvements in endpoint security as described in [Section 11.1](#), a certain percentage of endpoints will always get compromised. For this reason, all parties must regard data coming from endpoints as potentially unreliable or even malicious. An analogy can be drawn with human testimony in an investigation or trial. Human testimony is essential but must be regarded with suspicion.

- o Compromise of endpoint: A compromised endpoint may report false information to confuse or even provide maliciously crafted information with a goal of infecting others.
- o Putting bad information in SWID directory: Even if an endpoint is not completely compromised, some of the software running on it may be unreliable or even malicious. This software, potentially including the SWID generation or discovery tool, or malicious software pretending to be a SWID generation or discovery tool, can place incorrect or maliciously crafted information into the SWID directory. Endpoint users may even place such information in the directory, whether motivated by curiosity or confusion or a desire to bypass restrictions on their use of the endpoint.
- o Identity spoofing (impersonation): A compromised endpoint may attempt to impersonate another endpoint to gain its privileges or to besmirch the reputation of that other endpoint.

11.2.2. Network Attacks

A variety of attacks can be mounted using the network. Generally, the network cannot be trusted.

- o Eavesdropping, modification, injection, replay, deletion
- o Traffic analysis
- o Denial of service and blocking traffic

11.2.3. Server Attacks

The server is a critical security element and therefore merits considerable scrutiny.

- o Compromised trusted server: A compromised server or a malicious party that is able to impersonate a server can incorrectly grant or deny access to endpoints, place incorrect information into the repository, or send malicious messages to endpoints

- o Misconfiguration of trusted server: Accidental or purposeful misconfiguration of a trusted server can cause effects that are similar to those listed for compromised trusted server.
- o Malicious untrusted server: An untrusted server cannot mount any significant attacks because all properly implemented endpoints will refuse to engage in any meaningful dialog with such a server.

11.2.4. Repository Attacks

The repository is also an important security element and therefore merits careful scrutiny.

- o Putting bad information into trusted repository: An authorized repository client such as a server may be able to put incorrect information into a trusted repository or delete or modify historical information, causing incorrect decisions about endpoint security. Placing maliciously crafted data in the repository could even lead to compromise of repository clients, if they fail to carefully check such data.
- o Compromised trusted repository: A compromised trusted repository or a malicious untrusted repository that is able to impersonate a trusted repository can lead to effects similar to those listed for "Putting bad information into trusted repository". Further, a compromised trusted repository can report different results to different repository clients or deny access to the repository for selected repository clients.
- o Misconfiguration of trusted repository: Accidental or purposeful misconfiguration of a trusted repository can deny access to the repository or result in loss of historical data.
- o Malicious untrusted repository: An untrusted repository cannot mount any significant attacks because all properly implemented repository clients will refuse to engage in any meaningful dialog with such a repository.

11.3. Countermeasures

This section lists the countermeasures that can be used in an Endpoint Compliance Profile environment.

11.3.1. Countermeasures for Endpoint Attacks

This profile is in and of itself a countermeasure for a compromised endpoint. A primary defense for an endpoint is to run up to date software configured to be run as safely as possible.

Ensuring that anti-virus signatures are up to date and that a firewall is configured are also protections for an endpoint that are supported by the current NEA specifications.

Endpoints that have hardware cryptographic modules that are provisioned by the enterprise, in accordance with [[IEEE-802-1ar](#)], can protect the private keys used for authentication and help prevent adversaries from stealing credentials that can be used for impersonation. Future versions of the Endpoint Compliance Profile may want to discuss in greater detail how to use a hardware cryptographic module, in accordance with [[IEEE-802-1ar](#)], to protect credentials and to protect the integrity of the code that executes during the bootstrap process.

11.3.2. Countermeasures for Network Attacks

To address network attacks, [[RFC6876](#)] includes required encryption, authentication, integrity protection, and replay protection. [[Server-Discovery](#)] also includes authorization checks to ensure that only authorized servers are trusted by endpoints. Any unspecified or not yet specified network protocols employed in the Endpoint Compliance Profile (e.g. the protocol used to interface with the repository) should include similar protections.

These protections reduce the scope of the network threat to traffic analysis and denial of service. Countermeasures for traffic analysis (e.g. masking) are usually impractical but may be employed. Countermeasures for denial of service (e.g. detecting and blocking particular sources) SHOULD be used when appropriate to detect and block denial of service attacks. These are routine practices in network security.

11.3.3. Countermeasures for Server Attacks

Because of the serious consequences of server compromise, servers SHOULD be especially well hardened against attack and minimized to reduce their attack surface. They SHOULD be monitored using the NEA protocols to ensure the integrity of the behavior and analysis data stored on the server and SHOULD utilize a [[IEEE-802-1ar](#)] compliant hardware cryptographic module for identity and/or integrity measurements of the server. They should be well managed to minimize vulnerabilities in the underlying platform and in systems upon which the server depends. Network security measures such as firewalls or intrusion detection systems may be used to monitor and limit traffic to and from the server. Personnel with administrative access to the server should be carefully screened and monitored to detect problems as soon as possible. Server administrators should not use password-based authentication but should instead use non-reusable credentials

and multi-factor authentication (where available). Physical security measures should be employed to prevent physical attacks on servers.

To ease detection of server compromise should it occur, server behavior should be monitored to detect unusual behavior (such as a server reboot, unusual traffic patterns, or other odd behavior). Endpoints should log and/or notify users and/or administrators when peculiar server behavior is detected. To aid forensic investigation, permanent read-only audit logs of security-relevant information pertaining to servers (especially administrative actions) should be maintained. If server compromise is detected, the server's certificate should be revoked and careful analysis should be performed of the source and impact of this compromise. Any reusable credentials that may have been compromised should be reissued.

Endpoints can reduce the threat of server compromise by minimizing the number of trusted servers, using the mechanisms described in [[Server-Discovery](#)].

11.3.4. Countermeasures for Repository Attacks

If the host for the repository is located on its own endpoint, it should be protected with the same measures taken to protect the server. In this circumstance, all messages between the server and repository should be protected with a mature security protocol such as TLS or IPsec.

The repository can aid in the detection of compromised endpoints if an adversary cannot tamper with its contents. For instance, if an endpoint reports that it does not have an application with a known vulnerability installed, an administrator can check whether the endpoint might be lying by querying the repository for the history of what applications were installed on the endpoint.

To help prevent tampering with the information in the repository:

1. Only authorized parties should have privilege to run code on the endpoint and to change the repository.
2. If a separate endpoint hosts the repository, then the functionality of that endpoint should be limited to hosting the repository. The firewall on the repository should only allow access to the server and to any endpoint authorized for administration.
3. The repository should ideally use "write once" media to archive the history of what was placed in the repository, to include a snapshot of the current status of applications on endpoints.

12. Privacy-Considerations

The Endpoint Compliance Profile specifically addresses the collection of posture data from enterprise endpoints by an enterprise network. As such, privacy is not going to often arise as a concern for those deploying this solution.

A possible exception may be the concerns a user may have when attempting to connect a personal endpoint (such as a phone or mobile endpoint) to an enterprise network. The user may not want to share certain details, such as an endpoint identifier or SWID tags, with the enterprise. The user can configure their NEAC to reject requests for this information; however, it is possible that the enterprise policy will not allow the user's endpoint to connect to the network without providing the requested data.

13. References

13.1. Informative References

- [DSD] http://www.dsd.gov.au/publications/csocprotect/top_4_mitigations.htm, "Top 4 Mitigation Strategies to Protect Your ICT System", November 2012.
- [IEEE-802-1ar]
Institute of Electrical and Electronics Engineers, "IEEE 802.1ar", December 2009.
- [RFC5209] Sangster, P., Khosravi, H., Mani, M., Narayan, K., and J. Tardo, "Network Endpoint Assessment (NEA): Overview and Requirements", [RFC 5209](#), DOI 10.17487/RFC5209, June 2008, <<http://www.rfc-editor.org/info/rfc5209>>.
- [SANS] <http://www.sans.org/critical-security-controls/>, "CIS Critical Security Controls".
- [TNC] Trusted Computing Group, "TCG Trusted Network Connect TNC Architecture for Interoperability, Version 1.5", February 2012.

13.2. Normative References

- [I-D.ietf-sacm-terminology]
Waltermire, D., Montville, A., Harrington, D., and N. Cam-Winget, "Terminology for Security Assessment", [draft-ietf-sacm-terminology-05](#) (work in progress), August 2014.

- [IF-IMC] Trusted Computing Group, "TCG Trusted Network Connect TNC IF-IMC, Version 1.3", February 2013.
- [IF-IMV] Trusted Computing Group, "TCG Trusted Network Connect TNC IF-IMV, Version 1.4", December 2014.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC5792] Sangster, P. and K. Narayan, "PA-TNC: A Posture Attribute (PA) Protocol Compatible with Trusted Network Connect (TNC)", [RFC 5792](#), DOI 10.17487/RFC5792, March 2010, <<http://www.rfc-editor.org/info/rfc5792>>.
- [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](#), DOI 10.17487/RFC5793, March 2010, <<http://www.rfc-editor.org/info/rfc5793>>.
- [RFC6876] Sangster, P., Cam-Winget, N., and J. Salowey, "A Posture Transport Protocol over TLS (PT-TLS)", [RFC 6876](#), DOI 10.17487/RFC6876, February 2013, <<http://www.rfc-editor.org/info/rfc6876>>.
- [RFC7632] Waltermire, D. and D. Harrington, "Endpoint Security Posture Assessment: Enterprise Use Cases", [RFC 7632](#), DOI 10.17487/RFC7632, September 2015, <<http://www.rfc-editor.org/info/rfc7632>>.
- [Server-Discovery] Trusted Computing Group, "DRAFT: TCG Trusted Network Connect PDP Discovery and Validation, Version 1.0", October 2015.
- [SWID] "Information technology--Software asset management--Part 2: Software identification tag", ISO/IEC 9899:1999, 2009.

Authors' Addresses

Danny Haynes
The MITRE Corporation
202 Burlington Road
Bedford, MA 01730
USA

Email: dhaynes@mitre.org

Jessica Fitzgerald-McKay
Department of Defense
9800 Savage Road
Ft. Meade, Maryland
USA

Email: jmfitz2@nsa.gov

Lisa Lorenzin
Pulse Secure
2700 Zanker Rd., Suite 200
San Jose, CA 95134
US

Email: llorenzin@pulsesecure.net

