[MS-NAPSO]:
Network Policy and Access Services System Overview

This document provides an overview of the Network Policy and Access Services System Overview Protocol Family. It is intended for use in conjunction with the Microsoft Protocol Technical Documents, publicly available standard specifications, network programming art, and Microsoft Windows distributed systems concepts. It assumes that the reader is either familiar with the aforementioned material or has immediate access to it.

A Protocol Family System Document does not require the use of Microsoft programming tools or programming environments in order to implement the Protocols in the System. Developers who have access to Microsoft programming tools and environments are free to take advantage of them.

Abstract

Describes a series of tasks required to allow NAP Clients to gain access to a NAP-protected network; also describes how various components work together to aid in ensuring the health and protection of networked systems.

Revision Summary

Date	Revision History	Revision Class	Comments
08/14/2009	0.1	Major	First Release.
09/25/2009	0.2	Minor	Updated the technical content.
11/06/2009	0.2.1	Editorial	Revised and edited the technical content.
12/18/2009	0.2.2	Editorial	Revised and edited the technical content.
01/29/2010	1.0	Major	Updated and revised the technical content.
03/12/2010	2.0	Major	Updated and revised the technical content.
04/23/2010	2.0.1	Editorial	Revised and edited the technical content.
06/04/2010	2.0.2	Editorial	Revised and edited the technical content.
07/16/2010	2.0.2	No change	No changes to the meaning, language, or formatting of the technical content.
08/27/2010	3.0	Major	Significantly changed the technical content.
10/08/2010	4.0	Major	Significantly changed the technical content.
11/19/2010	5.0	Major	Significantly changed the technical content.
01/07/2011	6.0	Major	Significantly changed the technical content.
02/11/2011	7.0	Major	Significantly changed the technical content.
03/25/2011	8.0	Major	Significantly changed the technical content.
05/06/2011	9.0	Major	Significantly changed the technical content.
06/17/2011	10.0	Major	Significantly changed the technical content.

Date	Revision History	Revision Class	Comments
09/23/2011	10.0	No change	No changes to the meaning, language, or formatting of the technical content.
12/16/2011	11.0	Major	Significantly changed the technical content.
03/30/2012	12.0	Major	Significantly changed the technical content.
07/12/2012	12.0	No change	No changes to the meaning, language, or formatting of the technical content.
10/25/2012	12.0	No change	No changes to the meaning, language, or formatting of the technical content.
01/31/2013	12.0	No change	No changes to the meaning, language, or formatting of the technical content.
08/08/2013	13.0	Major	Significantly changed the technical content.
11/14/2013	13.0	No change	No changes to the meaning, language, or formatting of the technical content.
02/13/2014	13.0	No change	No changes to the meaning, language, or formatting of the technical content.

Contents

1 Introduction

A "Defined Task" is a logical procedure that uses one or more protocols or systems to accomplish a specific goal. This Defined Task System Document describes the tasks that are part of the NAP System.

Reasonable knowledge of common networking protocols and network security protocols is required to understand this document.

In conjunction with Protocol Technical Documents, which are primarily intended to cover protocols, this Defined Task System Document presents and covers the rules for information exchange and the protocols relevant to the tasks that are used to interoperate or communicate with Windows client operating systems and selected Windows Server operating system scenarios (those covered in published technical documents).

This document describes the defined tasks to accomplish system health-validated access to enterprise resources and is organized as follows:

- This section, Introduction, describes what is covered in this document, provides a list of terms defined in this document, as well as terms used in this document but defined elsewhere in the documentation set, and provides a list of references that apply to the overall system.
- Section 2, Overview, provides a high-level overview of the NAP System Tasks and the protocols that participate in those tasks.
- Section 3, Background Knowledge and System-Specific Concepts, describes system-specific concepts required to understand this document. It provides references to other resources that provide more in-depth coverage of the background information described in this document.
- Section 4, Common Task Information, contains information that is common to the NAP System.
- Section 5, Update NAP Client Configuration, describes how the configuration of the Network Access Protection client (NAP client) is updated.
- Section 6, Create and Send SoH Task, describes how the NAP agent creates the statement of health (SoH) on a client computer when a change of health status occurs and how it sends it to the server.
- Section 7, Proxy SoH Task, describes how the policy enforcement point (PEP) sends SoH messages to the NAP health policy server.
- Section 8, Receive SoH Task, describes how the health policy server receives encapsulated SoH messages through either the Vendor-Specific RADIUS Attributes for Network Access Protection (NAP) Data Structure (RNAP) or Health Certificate Enrollment Protocol (HCEP).
- Section 9, Process SoH Task, describes how the health policy server evaluates health data in the SoH.
- Section 10, Create and Send SoHR Task, describes how the health policy server creates a statement of health response (SoHR) and transmits it through the RNAP or HCEP protocols.
- Section 11, Enforce NAP Policy Task, describes how the PEP enforces the NAP policy.
- Section 12, Proxy SoHR Task, describes how the health policy server sends SoHR messages to the PEP and how the PEP proxy the SoHR message to the NAP client.
- Section 13, Receive SoHR Task, describes how the NAP client receives SoHR messages.

- Section 14, Process SoHR Task, describes how the NAP client processes the SoHR messages.
- Section 15, Remediate Client Health Task, describes how NAP client remediates health when the NAP client is not compliant based on the health-evaluation results.
- Section 16, Security, describes system-wide security issues that are not otherwise described in the protocol documents related to the tasks covered in this document.
- Section 17, Appendix A: Product Behavior, lists the versions of Windows or other Microsoft products that implement tasks in this system.

1.1 Glossary

The following terms are defined in [MS-GLOS]:

authentication
authorization
Dynamic Host Configuration Protocol (DHCP)
enforcement client
Extensible Authentication Protocol (EAP)
Group Policy
Group Policy server
health messages
health policy server
health certificate enrollment agent (HCEA)
health registration authority (HRA)
health state
Internet Protocol version 4 (IPv4)
Internet Protocol version 6 (IPv6)
Network Access Protection (NAP)
network access server (NAS)
policy
remediation server
statement of health (SoH)
statement of health response (SoHR)

The following terms are defined in [MS-RNAP]:

vendor-specific attribute (VSA)

The following terms are defined in [MS-WSH]:

remediation
security updates

The following terms are specific to this document:

client computer: A computer capable of examining and reporting on its **health**, and requesting for and using network resources.

client user: The individual who requires access to network resources.

DHCP channel: A **policy enforcement point (PEP)** channel that uses **Dynamic Host Configuration Protocol (DHCP)** to transport the **SoH**/**SoHR** messages between the **NAP client** and the **PEP**.

health: The condition of a computer with respect to the state and configuration of security-related components, operating system updates, applications, and configuration settings. For example, whether the latest updates (especially security-related) for the operating system and prescribed security applications or tools are installed, as well as the use and configuration of security technologies such as anti-malware software and host-based firewalls.

Health Certificate Enrollment Protocol (HCEP): A protocol designed to accomplish health certificate enrollment. Health certificates encapsulate the client's compliance to **policy** in a way that can be presented to interested parties without requiring those parties to perform the validation themselves.

health requirement server: A computer that provides current system **health state** for **NAP health policy servers**. For example, a **health requirement server** for an antivirus program tracks the latest version of the antivirus signature file.

healthy: A system is deemed to be **healthy**, or compliant, when its **health** is compliant with the system **health** requirements of the enterprise. **Unhealthy**, or noncompliant, implies a system that is not compliant with system **health** requirements.

HTTP/S channel: A **PEP channel** that uses HTTP/S to transport the **SoH**/**SoHR** messages between the **NAP client** and the **PEP**.

NAP administration server: A component of the **NAP health policy server**. The **NAP administration server** facilitates communication between the **NAP health policy server** and the **system health validator (SHV)**. The **NAP administration server** component is provided with the NAP platform.

NAP agent: A component of the **Network Access Protection client (NAP client)**. The **NAP agent** maintains the current **health state** information of the **NAP client** and facilitates communication between the **NAP enforcement client (NAP EC)** components and system **health** agent components.

NAP enforcement client (NAP EC): The NAP **enforcement client** components are part of the **NAP client**. A **NAP EC** can be defined for different type of network access or communication.

NAP enforcement server (NAP ES): Components that are part of the NAP enforcement point.

NAP health policy server (NPS): Stores **health** requirement policies and provides **health state** validation for NAP clients.

Network Access Protection client (NAP client): The **NAP client** is the set of NAP components installed and running on a **client computer**. The **NAP client** is responsible for executing NAP related operations on the client side. The **NAP client** is also responsible for collecting **health** information on the **client computer**, composing the **health** information into an **SoH** [TNC-IF-TNCCSPBSoH], and sending the **SoH** to a **PEP**.

PEAP channel: A **PEP channel** that uses the **Protected Extensible Authentication Protocol (PEAP)** to transport the **SoH**/**SoHR** messages between the **NAP client** and the **PEP**.

PEP channel: An abstract interface that is used by the **NAP client** to transport **SoH** messages to and from the **PEP**. Examples of **PEP channels** are **DHCP**, HTTP/S and **PEAP channels** used to transport **SoH** messages.

policy decision point (PDP): The point where **policy** decisions are made. In the case of NAP, this is the **NAP health policy server**.

policy enforcement point (PEP): The point where the **policy** decisions are actually enforced.

Protected Extensible Authentication Protocol (PEAP): An extension to the **Extensible Authentication Protocol (EAP)** that adds security services to the **EAP** methods.

RADIUS: Remote Authentication Dial In User Service, as specified in [RFC2865]. For the Microsoft **vendor-specific attributes (VSAs)** that are passed over **RADIUS**, see [MS-RNAP].

restricted network: A network on which noncompliant systems are placed, preventing their access to compliant systems. The **restricted network** may contain **remediation servers** so that noncompliant clients can update their configurations to comply with system **health** requirements.

RNAP: See **Vendor-Specific RADIUS Attributes for Network Access Protection (NAP) Data Structure (RNAP)**.

security principal: An entity associated with a human user or a program that can be authenticated. At a minimum, it has two basic attributes, a name and an identifier, that uniquely identifies it and makes it meaningful to the system, administrators, and users. A **security principal** is also known as a principal or an account.

statement of health ReportEntry (SoH ReportEntry): A collection of data that represents a specific aspect of the **health state** of a client.

statement of health response ReportEntry (SoHR ReportEntry): A collection of data that represents the evaluation of a specific aspect of the **health state** of a client, according to network **policies**.

system health agent (SHA): The client components that make declarations on a specific aspect of the client **health state** and generate an **SoH ReportEntry**.

system health validator (SHV): The server counterpart to the **System Health Agent (SHA)**, which is responsible for verifying the declarations of client **health state** made by the respective **SHA**. The **SHV** generates an **SoHR ReportEntry**.

system statement of health (SSoH): Messages that are grouped together into a **system statement of health (SSoH)** message represent the aggregate of all known aspects of the client's **health**. **SSoH** messages are sent within other protocols to enable the evaluation of the overall client **health** by passing each **SoH** from the **SSoH** to the **SHV**.

Terminal Services Gateway (TSG) server: A server that allows authorized remote users from the Internet to connect to resources on a private network.

Terminal Services Gateway Server Protocol: A protocol that is primarily used for tunneling client to server traffic across firewalls when the **Terminal Services Gateway (TSG) server** is deployed in the perimeter network of an intranet, as specified in [MS-TSGU].

TSGU: See **Terminal Services Gateway Server Protocol**.

Vendor-Specific RADIUS Attributes for Network Access Protection (NAP) Data Structure (RNAP): The Microsoft **RADIUS VSAs** that are implemented in the Windows operating system as specified in [MS-RNAP].

virtual private network (VPN): A network that provides secure access to a private network over public infrastructure.

VPN client: A client that makes remote resources of another network available in a secure way.

VPN connection: A connection that transfers private data across the public network using the routing infrastructure of the Internet.

VPN server: A server that makes remote resources of another network available in a secure way.

Windows Client Certificate Enrollment Protocol (WCCE): A protocol used by a **health registration authority (HRA)** to obtain a signed **health** certificate for issuing to **client computers** that are compliant with **health policy** in an IPsec configuration, as specified in [MS-WCCE].

Windows Security Health Agent (WSHA): Reports the system security **health state** (Windows Security Center) to the **Windows Security Health Validator (WSHV)**, as specified in [MS-WSH].

Windows Security Health Validator (WSHV): Responds to the report received from the **Windows Security Health Agent (WSHA)**. If the status reported by the **WSHA** does not comply with the defined security **health policy**, the response from the **WSHV** includes quarantine and **remediation** instructions as specified in [MS-WSH].

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as described in [RFC2119]. Note that in [RFC2119] terms, most of these specifications should be imperative, to ensure interoperability. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

Any specification that does not explicitly use one of these terms is mandatory, exactly as if it used MUST.

1.2 References

References to Microsoft Open Specifications documentation do not include a publishing year because links are to the latest version of the documents, which are updated frequently. References to other documents include a publishing year when one is available.

A reference marked "(Archived)" means that the reference document was either retired and is no longer being maintained or was replaced with a new document that provides current implementation details. We archive our documents online [Windows Protocol].

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information.

[MS-CAESO] Microsoft Corporation, "Certificate Autoenrollment System Overview". (Archived)

[MS-DHCPE] Microsoft Corporation, "Dynamic Host Configuration Protocol (DHCP) Extensions".

[MS-DHCPM] Microsoft Corporation, "Microsoft Dynamic Host Configuration Protocol (DHCP) Server Management Protocol".

[MS-DHCPN] Microsoft Corporation, "Dynamic Host Configuration Protocol (DHCP) Extensions for Network Access Protection (NAP)".

[MS-GPNAP] Microsoft Corporation, "Group Policy: Network Access Protection (NAP) Extension".

[MS-GPREG] Microsoft Corporation, "Group Policy: Registry Extension Encoding".

[MS-GPSO] Microsoft Corporation, "Group Policy System Overview". (Archived)

[MS-HCEP] Microsoft Corporation, "Health Certificate Enrollment Protocol".

[MS-PEAP] Microsoft Corporation, "Protected Extensible Authentication Protocol (PEAP)".

[MS-RNAP] Microsoft Corporation, "Vendor-Specific RADIUS Attributes for Network Access Protection (NAP) Data Structure".

[MS-TLSP] Microsoft Corporation, "Transport Layer Security (TLS) Profile".

[MS-TSGU] Microsoft Corporation, "Terminal Services Gateway Server Protocol".

[MS-WCCE] Microsoft Corporation, "Windows Client Certificate Enrollment Protocol".

[MS-WSH] Microsoft Corporation, "Windows Security Health Agent (WSHA) and Windows Security Health Validator (WSHV) Protocol".

[MS-WSO] Microsoft Corporation, "Windows System Overview". (Archived)

[RFC1661] Simpson, W., Ed., "The Point-to-Point Protocol (PPP)", STD 51, RFC 1661, July 1994, http://www.ietf.org/rfc/rfc1661.txt

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, http://www.rfc-editor.org/rfc/rfc2119.txt

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997, http://www.ietf.org/rfc/rfc2131.txt

[RFC2132] Alexander, S., and Droms, R., "DHCP Options and BOOTP Vendor Extensions", RFC 2132, March 1997, http://www.ietf.org/rfc/rfc2132.txt

[RFC2315] Kaliski, B., "PKCS #7: Cryptographic Message Syntax Version 1.5", RFC 2315, March 1998, http://www.ietf.org/rfc/rfc2315.txt

[RFC2401] Kent, S., and Atkinson, R., "Security Architecture for the Internet Protocol", RFC 2401, November 1998, http://www.ietf.org/rfc/rfc2401.txt

[RFC2409] Harkins, D., and Carrel, D., "The Internet Key Exchange (IKE)", RFC 2409, November 1998, http://www.ietf.org/rfc/rfc2409.txt

[RFC2548] Zorn, G., "Microsoft Vendor-Specific RADIUS Attributes", RFC 2548, March 1999, http://www.ietf.org/rfc/rfc2548.txt

[RFC2616] Fielding, R., Gettys, J., Mogul, J., et al., "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999, http://www.ietf.org/rfc/rfc2616.txt

[RFC2716] Aboba, B., and Simon, D., "PPP EAP TLS Authentication Protocol", RFC 2716, October 1999, http://www.ietf.org/rfc/rfc2716.txt

[RFC2753] Yavatkar, R., Pendarakis, D., and Guerin, R., "A Framework for Policy-based Admission Control", RFC 2753, January 2000, http://www.ietf.org/rfc/rfc2753.txt

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000, http://www.ietf.org/rfc/rfc2818.txt

[RFC2865] Rigney, C., Willens, S., Rubens, A., and Simpson, W., "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000, http://www.ietf.org/rfc/rfc2865.txt

[RFC2866] Rigney, C., "RADIUS Accounting", RFC 2866, June 2000, http://www.ietf.org/rfc/rfc2866.txt

[RFC3280] Housley, R., Polk, W., Ford, W., and Solo, D., "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 3280, April 2002, http://www.ietf.org/rfc/rfc3280.txt

[RFC3548] Josefsson, S., Ed., "The Base16, Base32, and Base64 Data Encodings", RFC 3548, July 2003, http://www.ietf.org/rfc/rfc3548.txt

[RFC3579] Aboba, B., and Calhoun, P., "RADIUS (Remote Authentication Dial In User Service) Support For Extensible Authentication Protocol (EAP)", RFC 3579, September 2003, http://www.ietf.org/rfc/rfc3579.txt

[RFC3580] Congdon, P., Aboba, B., Smith, A., and et al., "IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines", RFC 3580, September 2003, http://www.ietf.org/rfc/rfc3580.txt

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., et al., "Extensible Authentication Protocol (EAP)", RFC 3748, June 2004, http://www.ietf.org/rfc/rfc3748.txt

[RFC3925] Littlefield, J., "Vendor-Identifying Vendor Options for Dynamic Host Configuration Protocol Version 4 (DHCPv4)", RFC 3925, October 2004, http://www.ietf.org/rfc/rfc3925.txt

[RFC4306] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", RFC 4306, December 2005, http://www.ietf.org/rfc/rfc4306.txt

[RFC4559] Jaganathan, K., Zhu, L., and Brezak, J., "SPNEGO-based Kerberos and NTLM HTTP Authentication in Microsoft Windows", RFC 4559, June 2006, http://www.ietf.org/rfc/rfc4559.txt

[TNC-IF-TNCCSPBSoH] TCG, "TNC IF-TNCCS: Protocol Bindings for SoH", version 1.0, May 2007, http://www.trustedcomputinggroup.org/resources/tnc_iftnccs_protocol_bindings_for_soh_version_10/

1.2.2 Informative References

[IEEE802.1X] Institute of Electrical and Electronics Engineers, "IEEE Standard for Local and Metropolitan Area Networks - Port-Based Network Access Control", December 2004, http://ieeexplore.ieee.org/iel5/9828/30983/01438730.pdf

[MS-ADA2] Microsoft Corporation, "Active Directory Schema Attributes M".

[MS-GLOS] Microsoft Corporation, "Windows Protocols Master Glossary".

[MS-RRASM] Microsoft Corporation, "Routing and Remote Access Server (RRAS) Management Protocol".

[MSDN-CorrelationId] Microsoft Corporation, "CorrelationId structure", http://msdn.microsoft.com/en-us/library/aa369150(v=vs.85).aspx

[MSDN-MGMTFUNCS] Microsoft Corporation, "Management Functions", http://msdn.microsoft.com/en-us/library/aa364943(v=VS.85).aspx

[MSDN-NAPAPI] Microsoft Corporation, "NAP Interfaces", http://msdn.microsoft.com/en-us/library/aa369705(v=VS.85).aspx

[MSFT-802.1XEnforceConfig] Microsoft Corporation, "802.1X Enforcement Configuration", http://technet.microsoft.com/es-es/library/dd125308(WS.10).aspx

[MSFT-CFGNAPTRCNG] Microsoft Corporation, "Configure NAP Tracing" http://technet.microsoft.com/en-us/library/cc771276.aspx

[MSFT-ConnReqPolicies] Microsoft Corporation, "Connection Request Policies", http://technet.microsoft.com/en-us/library/cc753603.aspx

[MSFT-HealthPolicies] Microsoft Corporation, "Health Policies", http://technet.microsoft.com/en-us/library/cc771934.aspx

[MSFT-NetworkPolicies] Microsoft Corporation, "Network Policies", http://technet.microsoft.com/en-us/library/cc754107.aspx

[X509] ITU-T, "Information Technology - Open Systems Interconnection - The Directory: Public-Key and Attribute Certificate Frameworks", Recommendation X.509, August 2005, http://www.itu.int/rec/T-REC-X.509/en

Note There is a charge to download the specification.

2 Overview

Section 1, "Introduction" primarily describes this Defined Task System Document per se. This section introduces the tasks that are being documented.

2.1 Summary

The **Network Access Protection (NAP)** System consists of software components that enable computers to obtain **health**-validated access to resources from a network.

The NAP System Tasks provide the means for computers to report system health and have it verified, and ensure that only compliant computers are given access to the network or resource. Noncompliant hosts can have their network access restricted or be denied access to a network resource. Most commonly, the point at which the health of a system is verified when a computer attempts to gain access to a private network or a resource on the network. Noncompliant computers can have their access limited to a **restricted network** that contains resources, known as **remediation servers**, which allow noncompliant computers to become compliant. Noncompliant computers can access the remediation servers on the restricted network to obtain the necessary updates, anti-virus signatures, and other software or instructions necessary to become compliant.

Note that the health evaluation that NAP performs can be invoked at any time. For example, the health of a computer can be checked at noon every day or only when it tries to connect to corporate email servers. Although NAP can be and generally is used as a health validation and network access control mechanism, it is much more flexible and could be used for other purposes.

NAP can be used to manage and enforce compliance with the system health requirements of the enterprise. NAP uses network infrastructure capabilities and the capabilities of other NAP components to restrict network access to computers that are not compliant with **policy**.

A NAP-enabled network infrastructure consists of the following:

- **Network Access Protection clients (NAP clients)** use **enforcement client** components to send and receive system health information.
- NAP policy enforcement points (PEPs), which are servers or network access devices that use NAP or can be used with NAP to require the evaluation of a NAP client's **health state** and provide restricted network access or communication. Examples of PEPs include the following: Health Registration Authority and **Terminal Services Gateway server (TSGU)** (HTTP/S **PEP**), **virtual private network (VPN) server** and 802.1X capable devices **Protected Extensible Authentication Protocol (PEAP)** (PEAP PEP), and **DHCP** server (DHCP PEP).
- **NAP health policy servers**, which are servers that are running Windows Server 2008 operating system or later, and the Network Policy Server service that evaluates NAP client health status.
- An optional, restricted network containing remediation servers.

The NAP System consists of NAP clients exchanging system health information with the NAP health policy server using the various protocols of the enforcement client components as transport mechanisms for NAP messages (for example DHCP and PEAP). Implementation of NAP is required if there are other systems or applications that need health-validated access to the network or resources. An example is the DHCP Server service as described in [MS-DHCPN]. System health information is created and consumed by **system health agents (SHAs)** and **system health validators (SHVs)** such as the **Windows Security Health Agent (WSHA)** and **Windows Security Health Validator (WSHV)** as described in [MS-WSH].

The NAP System uses 11 defined tasks to accomplish its goal of health-validated access to enterprise resources. The following diagram illustrates the interaction of the defined tasks to accomplish the goal.

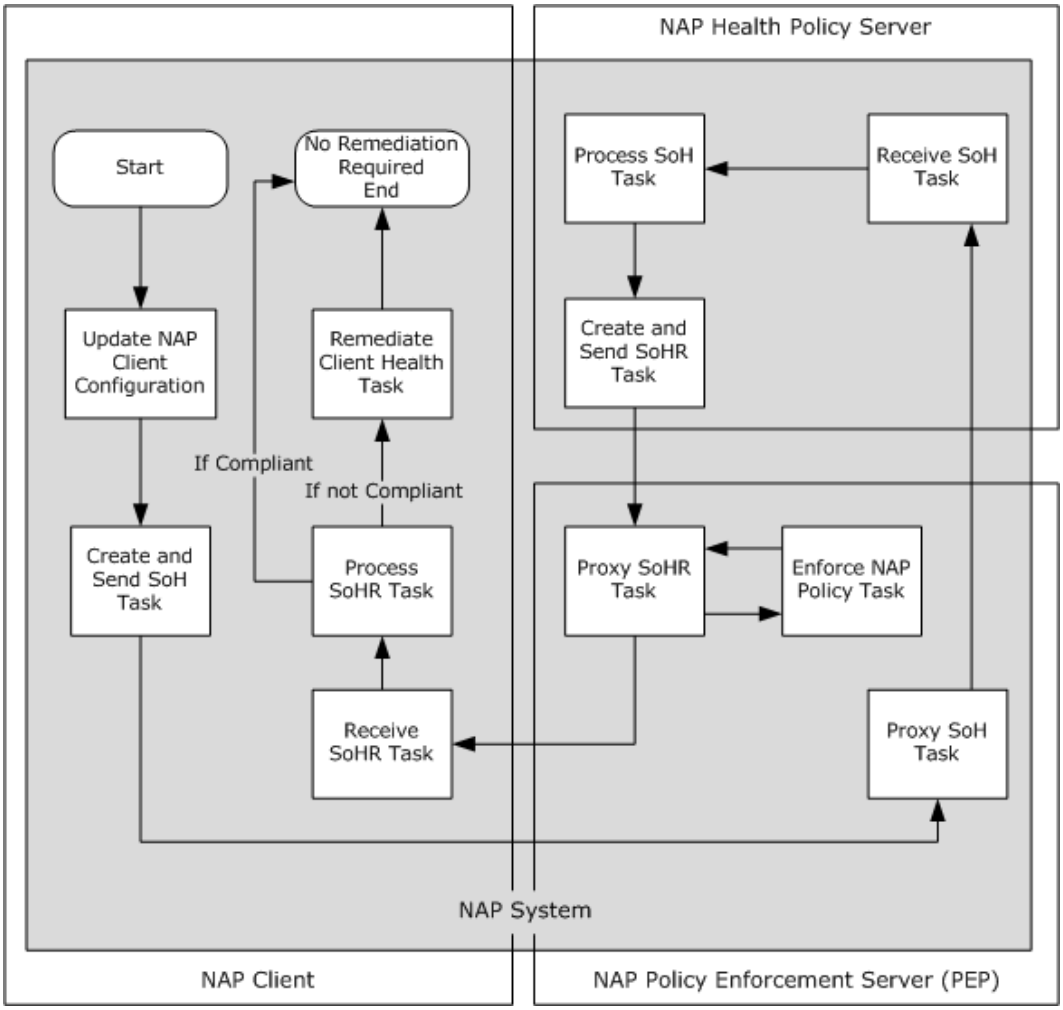

Figure 1: Process for health-validated access using ten defined tasks

The NAP client initiates the process by creating the **statement of health (SoH)**. It then sends it to the NAP health policy server for evaluation and the NAP health policy server receives the SoH and evaluates health based on the configured health policy.

The NAP health policy server creates the **statement of health response (SoHR)**, which includes the evaluation results and also steps to fix the client (if non-compliant), and sends it to the NAP client. The details of the tasks and their interactions are explained in sections 4 through 15.

2.2 List of Tasks

The NAP System Tasks described in this document are as follows:

Update NAP Client Configuration: This task describes the NAP client **Group Policy** configuration required for the NAP client to determine what EC to use and how to locate the policy enforcement point (PEP) if IPsec enforcement is used.

Create and Send SoH: This task describes the creation of the SoH on a client machine.

Proxy SoH: This task describes the process of sending the SoH from the client machine to the **policy decision point (PDP)** (the NAP health policy server).

Receive SoH: This task describes the process of receiving the SoH on the PDP (the NAP health policy server).

Process SoH: This task describes the process of evaluating health on the PDP (the NAP health policy server).

Create and Send SoHR: This task describes the creation of the SoHR on the PDP (the NAP health policy server).

Proxy SoHR: This task describes the process of sending the SoHR from the PDP to the client machine and the PEP.

Enforce NAP Policy: This task describes the process to enforce network restrictions on the NAP client based on the health evaluation.

Receive SoHR: This task describes the process of receiving the SoHR on the client machine.

Process SoHR: This task describes the process of evaluating the SoHR on the client machine.

Remediate Client Health: This task describes the steps to remediate the NAP client based on the health evaluation results.

2.3 Relevant Standards

Relevant Microsoft protocols are as follows:

Windows Security Health Agent (WSHA) and Windows Security Health Validator (WSHV) Protocol: As specified in [MS-WSH]. This protocol is included in the message payload specified in the SoH for the NAP Protocol.

Dynamic Host Configuration Protocol (DHCP) Extensions for Network Access Protection (NAP): As specified in [MS-DHCPN]. This protocol defines DHCP, which is designed to reduce the administrative burden and complexity of configuring hosts on a TCP/IP-based network, such as a private intranet. DHCP is an enforcement method supported by NAP.

Terminal Services Gateway Server Protocol Specification: As specified in [MS-TSGU]. This protocol is used primarily for tunneling client to server traffic across firewalls when the Terminal Services Gateway (TSG) server is deployed in the perimeter network of an intranet.

Vendor-Specific RADIUS Attributes for Network Access Protection (NAP) Data Structure: As specified in [MS-RNAP]. This protocol specifies the Microsoft **RADIUS vendor-specific attributes (VSAs)** that are implemented in the Windows operating system.

Protocol Bindings for SoH: As specified in [TNC-IF-TNCCSPBSoH]. This protocol specifies the SoH protocol in which a client and a server exchange SoH and SoHR messages. This protocol, and the appropriate **authentication** protocols, helps enterprises ensure that computers on their networks are compliant with corporate policies.

Health Certificate Enrollment Protocol Specification: As specified in [MS-HCEP]. This protocol allows a network endpoint to obtain digital certificates.

Protected Extensible Authentication Protocol (PEAP) Specification: As specified in [MS-PEAP]. This protocol adds security services to the **Extensible Authentication Protocol (EAP)** methods.

The relevant standards are as follows:

Remote Authentication Dial-In User Service: As specified in [RFC2865]. This standard defines a protocol for carrying authentication, **authorization**, and accounting information between a **network access server (NAS)** and a shared authentication server.

Extensible Authentication Protocol (EAP): As specified in [RFC3748]. This standard defines a framework that supports multiple authentication methods. The Protected Extensible Authentication Protocol (PEAP) is an extension of EAP that carries computer health information along with authentication information.

PPP EAP TLS Authentication Protocol: As specified in [RFC2716]. The standard defines an authentication method that uses transport layer security (TLS) within the framework of EAP. PEAP [MS-PEAP] is based on EAP-TLS, extended for NAP to carry computer health information.

RADIUS Support for Extensible Authentication Protocol (EAP): As specified in [RFC3579]. This standard defines how to carry EAP payloads within RADIUS messages.

Dynamic Host Configuration Protocol: As specified in [RFC2131]. This standard defines DHCP.

Vendor-Identifying Vendor Options for Dynamic Host Configuration Protocol Version 4 (DHCPv4): As specified in [RFC3925]. This standard defines the vendor options used for DHCP. NAP uses these options to exchange system health information in the DHCP enforcement method.

Internet Key Exchange (IKEv2) Protocol: As specified in [RFC4306]. This standard defines version 2 of the Internet Key Exchange (IKE) Protocol. IKE is a component of IPsec used for performing mutual authentication and establishing and maintaining security associations (SAs).

Security Architecture for the Internet Protocol: As specified in [RFC2401]. This standard defines the base architecture for IPsec support in hosts.

3 Background Knowledge and System-Specific Concepts

This section identifies the theoretical and practical information needed to understand this document and the tasks in this system, and summarizes:

- Background knowledge that is required to understand this document.
- Concepts that are specific to the tasks in this system.

It is assumed that the reader of this document has the following background knowledge:

- Authentication, authorization, and accounting (AAA) concepts and the EAP and RADIUS protocols as described in [RFC2865], [RFC2866], [RFC3748] and [RFC3580].
- Encapsulating EAP payloads in link layer and RADIUS protocols, as described in [IEEE802.1X] and [RFC3579].
- DHCP, as described in [RFC2131], and the structure of associated extensions, as described in [RFC2132].
- HTTP and HTTPS, as described in [RFC2616] and [RFC2818], along with extensions for Windows authentication, as described in [RFC4559].
- IPsec, as described in [RFC2401], and related key exchange, as described in [RFC2409].

The vast majority of malware infections occur to systems that are either improperly configured or do not have the latest **security updates** for the operating system or key applications installed. Attackers create malicious software that targets out-of-date computers simply because they are the easiest targets. Computers that do not have the most recent operating system, application, and anti-malware updates not only expose themselves to risk but become a risk and source of attack to other computers on the network.

IT administrators labor to ensure that the systems they are responsible for are correctly configured and updated. This, in practice, is an extremely difficult task to accomplish for large organizations. The diversity of systems, their applications, and their uses make it impossible to ensure that every system is configured correctly and updated. Enforcing requirements is even more difficult when computers not on the corporate network, such as home computers or laptops used when traveling, are exposed to malicious environments that are not under the administrator's control. The goal is to have as many systems as possible securely configured and updated to minimize malware outbreaks on the enterprise network. IT organizations expend enormous resources doing nothing more than ensuring systems are properly configured and updated. It is an error prone and difficult task that is one of the largest IT cost components. The goal of Network Access Protection (NAP) is to reduce the cost of this endeavor while substantially increasing the coverage and the likelihood of success.

3.1 System Context

This section describes the relationship between this system and its environment.

3.1.1 System Environment

The Network Policy and Access Services System provides an integrated way of validating and monitoring the health state of a network **client computers** and limiting the access of network clients until the health policy requirements have been satisfied.

3.1.1.1 Network Infrastructure

This system requires access to network services that support:

- TCP over IP (**IPv4** or **IPv6**)
- UDP over IP (IPv4 or IPv6)
- Name resolution services such as the Domain Name System (DNS) and Windows Internet Name Service (WIN)

To validate access to a network based on system health, a network infrastructure needs to provide the following areas of functionality:

- **Health state validation:** Determines whether the network client computers are compliant with health policy requirements.
- **Network access limitation:** Limits access for noncompliant network client computers.
- **Automatic remediation:** Provides necessary updates to allow a noncompliant, network client computer to become compliant without user intervention.
- **Ongoing compliance:** Automatically updates compliant network client computers so that they adhere to ongoing changes in health policy requirements.
- **Certificate services:** A **health registration authority (HRA)** requires X.509 certificates for issuing to client computers that are compliant with health policy.

The Windows Network Policy and Access Services System provide the following enforcement methods:

- Internet Protocol security (IPsec) enforcement for IPsec-protected communications
- 802.1X enforcement for IEEE 802.1X-authenticated connections
- Virtual private network (VPN) enforcement for remote access **VPN connections**
- Dynamic Host Configuration Protocol (DHCP) enforcement for DHCP-based address configuration
- Terminal Services Gateway (TSG) connections

The Network Policy and Access Services System provide an extendable client and server-side architecture through which policy validation, network access limitation, automatic **remediation**, and ongoing compliance can occur.

3.2 System Assumptions and Preconditions

The following assumptions and preconditions MUST be satisfied for the Network Policy and Access Services System to operate successfully:

Network configuration: In order for system components running on different computers to communicate with each other, the network services and infrastructure MUST be functional and configured such that required protocols, ports, and so on are remotely accessible. In order for the 802.1x enforcement to operate successfully in a NAP environment, wireless network infrastructure such as switches and access points MUST support 802.1x/Protected Extensible Authentication Protocol (PEAP). In order for dynamic VLAN ID assignment to work, network hardware MUST be configured as specified in [RFC3580].

Domain configuration: In a domain configuration, system components have access to directory services provided by the domain. Domain configuration is required for Group Policy support.

NAP Agent: In order for a client to deliver a SoH and operate successfully it MUST be configured with an EC-specific protocol. The EC-specific protocol can be one of the following: [MS-HCEP], [MS-DHCPN], [MS-TSGU], and [MS-PEAP].

3.2.1 Task Protocol Roles

The Windows Network Policy and Access Services System utilizes the protocols and data structures specified in the following documents:

- [MS-ADA2] Active Directory Schema Attributes M

  Specifies the msRADIUSFramedIPAddress attribute in Active Directory used by the NAP service.

- [MS-DHCPN] Dynamic Host Configuration Protocol (DHCP) Extensions for Network Access Protection (NAP)

  Specifies a set of vendor-class options defined for use by DHCP clients and DHCP servers to support NAP enforcement through DHCP.

- [MS-GPNAP] Group Policy: Network Access Protection (NAP) Extension

  Specifies how the behavior of a NAP client can be controlled through Group Policy by updating the client registry.

- [MS-HCEP] Health Certificate Enrollment Protocol Specification

  The Health Certificate Enrollment Protocol supports authentication of the server, client, or both. If the client's health state is compliant, the HRA requests a certificate authority (CA) to issue a certificate for the client.

- [MS-PEAP] Protected Extensible Authentication Protocol (PEAP) Specification

  EAP is an authentication framework that supports multiple authentication methods. PEAP supports the transmission of SoH and SoHR messages.

- [MS-RNAP] Vendor-Specific RADIUS Attributes for Network Access Protection (NAP) Data Structure

  Specifies the Microsoft VSAs that are passed over RADIUS between the network access server (NAS) and the RADIUS server to authenticate and authorize connection requests.

- [MS-RRASM] Routing and Remote Access Server (RRAS) Management Protocol Specification

  Specifies the registry information that can be used to specify the overall RRAS configuration including NAP configuration.

- [MS-TSGU] Terminal Services Gateway Server Protocol Specification

  Allows determination of the NAP capability of a Terminal Services Client.

- [MS-WCCE] Windows Client Certificate Enrollment Protocol Specification

  An HRA uses a **Windows Client Certificate Enrollment Protocol (WCCE)** to obtain a signed health certificate for issuing to client computers that are compliant with health policy in an IPsec configuration.

- [MS-WSH] Windows Security Health Agent (WSHA) and Windows Security Health Validator (WSHV) Protocol Specification

  The Windows Security Health Agent (WSHA) reports the system security health state (Windows Security Center) to the Windows Security Health Validator (WSHV), which responds with quarantine and remediation instructions if the status reported, is not compliant with the defined security health policy.

- [TNC-IF-TNCCSPBSoH] Protocol Bindings for SoH

  Specifies the format and message exchange of SoH and SoHR messages.

3.3 Architectural Details

This section contains specifications that are common to all of the other tasks described in this document.

3.3.1 NAP Client Architecture

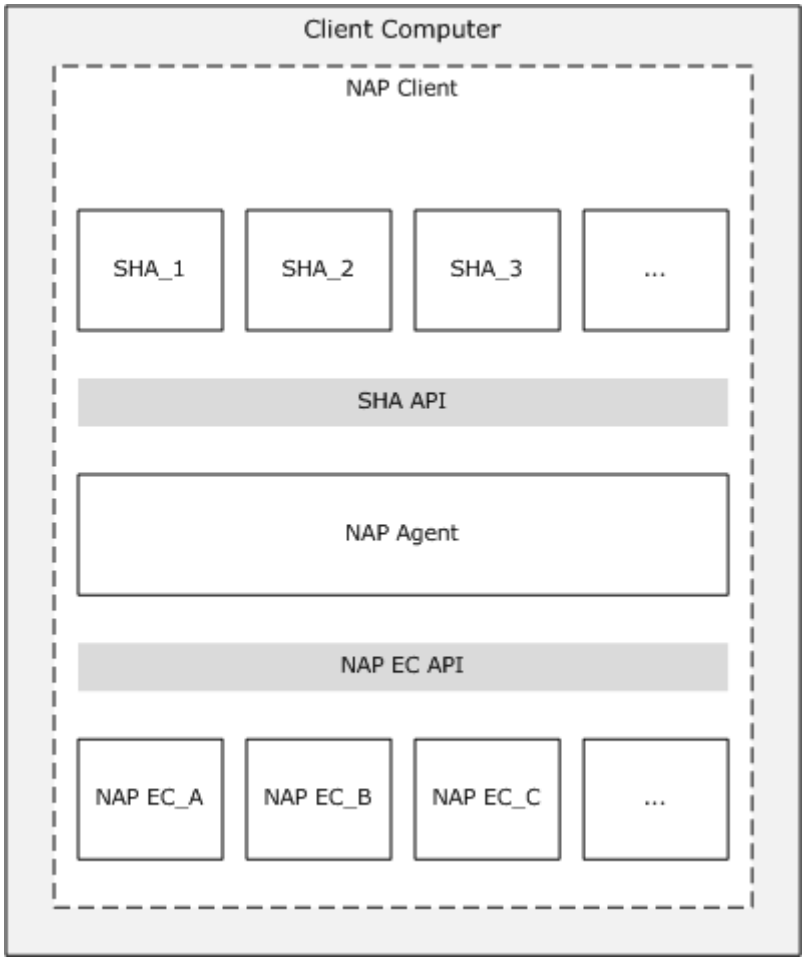

Figure 2: NAP client-side architecture

The NAP client architecture consists of the following:

- **A layer of NAP enforcement client (NAP EC) components**

  Each **NAP EC** is defined for a different type of network access or communication. For example, there is a NAP EC for remote access VPN connections and a NAP EC for DHCP configuration. The NAP EC can be matched to a specific type of NAP enforcement point. For example, the DHCP NAP EC is designed to work with a DHCP-based NAP enforcement point. Some NAP ECs are provided with the NAP platform while others are provided by third-party vendors.

- **A layer of system health agent (SHA) components**

  - An SHA is a component that maintains and reports one or multiple elements of system health. For example, there might be an SHA for antivirus signatures and an SHA for operating system updates.

  - An SHA can be matched to a remediation server. For example, an SHA for checking antivirus signatures is matched to the server that contains the latest antivirus signature file.

  - SHAs do not have to have a corresponding remediation server. For example, an SHA can just check local system settings to ensure that a host-based firewall is enabled.

  Starting in Windows XP operating system Service Pack 3 (SP3), Windows clients include a WSHA that monitors the settings of the Windows Security Center. Additional SHAs can be added to system.

- **NAP agent**

  Maintains the current health state information of the NAP client and facilitates communication between the NAP EC and SHA layers. The **NAP agent** is provided with the NAP platform.

- **SHA application programming interface (API)**

  Provides a set of function calls that allow SHAs to register with the NAP agent, to indicate system health status, respond to queries for system health status from the NAP agent, and for the NAP agent to pass system health remediation information to an SHA. The SHA API allows vendors to create and install additional SHAs. For information about the APIs, see [MSDN-NAPAPI].

- **NAP EC API**

  Provides a set of function calls that allow NAP ECs to register with the NAP agent, to request system health status, and pass system health remediation information to the NAP agent. The NAP EC API allows vendors to create and install additional NAP ECs.

  To indicate the health state of a specific SHA, an SHA creates an SoH message and passes it to the NAP agent. An SoH can contain one or multiple elements of system health. For example, the SHA for an antivirus program can create an SoH containing the state of the antivirus software running on the computer, its version, and the last antivirus signature update received. Whenever an SHA updates its status, it creates a new SoH and passes it to the NAP agent. To indicate the overall health state of a NAP client, the NAP agent uses a **system statement of health (SSoH)**, which includes version information for the NAP client and the set of SoHs for the installed SHAs.

  For information about the APIs, see [MSDN-NAPAPI].

- **NAP enforcement client**

  A NAP EC requests some level of access to a network, passes the computer's health status to a NAP enforcement point that is providing the network access, and indicates the limited or

unlimited network access status of the NAP client to other components of the NAP client architecture.

The NAP ECs for the NAP platform are the following:

- An IPsec NAP EC for IPsec-protected communications.
- An EAPHost NAP EC for 802.1X-authenticated connections.
- A VPN NAP EC for remote access VPN connections.
- A DHCP NAP EC for DHCP-based IPv4 address configuration.
- A TSG NAP EC for TSG connections.

- **IPsec NAP EC**

  The IPsec NAP EC is a component that obtains the SoH from the NAP agent and sends it to the HRA along with a request for a health certificate. The IPsec NAP EC is known as the IPsec Relying Party EC in the NAP Client Configuration snap-in. The IPsec NAP EC also interacts with the following:

  - The certificate store to store the health certificate.
  - The IPsec components in Windows to ensure that the health certificate is used for IPsec-protected communication.
  - The host-based firewall (such as Microsoft Windows Firewall) so that the IPsec-protected traffic is allowed by the firewall.

- **EAPHost NAP EC**

  The EAPHost NAP EC is a component that obtains the SoH from the NAP agent and sends it as a PEAP-Type-Length-Value (TLV) message for 802.1X-authenticated connections.

- **VPN NAP EC**

  The VPN NAP EC is new functionality in the Remote Access Connection Manager service that obtains the system statement of health (SSoH) message from the NAP agent and sends it as a PEAP-TLV message for remote access VPN connections. The VPN NAP EC is known as the Remote Access Quarantine EC in the NAP Client Configuration snap-in.

- **DHCP NAP EC**

  The DHCP NAP EC is new functionality in the DHCP Client service that uses industry standard DHCP messages to exchange system **health messages** and limited network access information. The DHCP NAP EC is known as the DHCP Quarantine EC in the NAP Client Configuration snap-in. The DHCP NAP EC obtains the SSoH from the NAP agent. The DHCP Client service fragments the SSoH, if required, and sets each fragment into a Microsoft vendor-specific DHCP option that is sent in DHCPDiscover, DHCPRequest or DHCPInform messages. DHCPDecline and DHCPRelease messages do not contain the SSoH.

- **System health agent (SHA)**

  An SHA performs system health updates and publishes its status in the form of an SoH to the NAP agent. The SoH contains information that the NAP health policy server can use to verify that the client computer is in the required state of health.

An SHA is matched to a system health validator (SHV) on the server-side of the NAP platform architecture. The corresponding SHV returns an SoHR to the NAP client, which is passed by the NAP EC and the NAP agent to the SHA, informing it of what to do if the SHA is not in a required state of health. For example, the SoHR sent by an antivirus SHV could instruct the corresponding antivirus SHA to request the latest version of the antivirus signature file from an antivirus signature server. The SoHR can also include the name or IP address of the antivirus signature server.

An SHA can use a locally installed system health component to assist in system health management functions in conjunction with a remediation server. For example, a software update SHA can use the locally installed software update client software to perform version checking and installation and update functions with the software update server (the remediation server).

- **NAP agent**

  The NAP agent provides the following services:

  - Collects the SoHs from each SHA and caches them. The SoH cache is updated whenever an SHA supplies a new or updated SoH.
  - Stores the SoH and supplies it to the NAP ECs upon request.
  - Passes notifications to SHAs when the limited network access state changes.
  - Passes SoHRs to the appropriate SHAs.

3.3.2 NAP Server Architecture

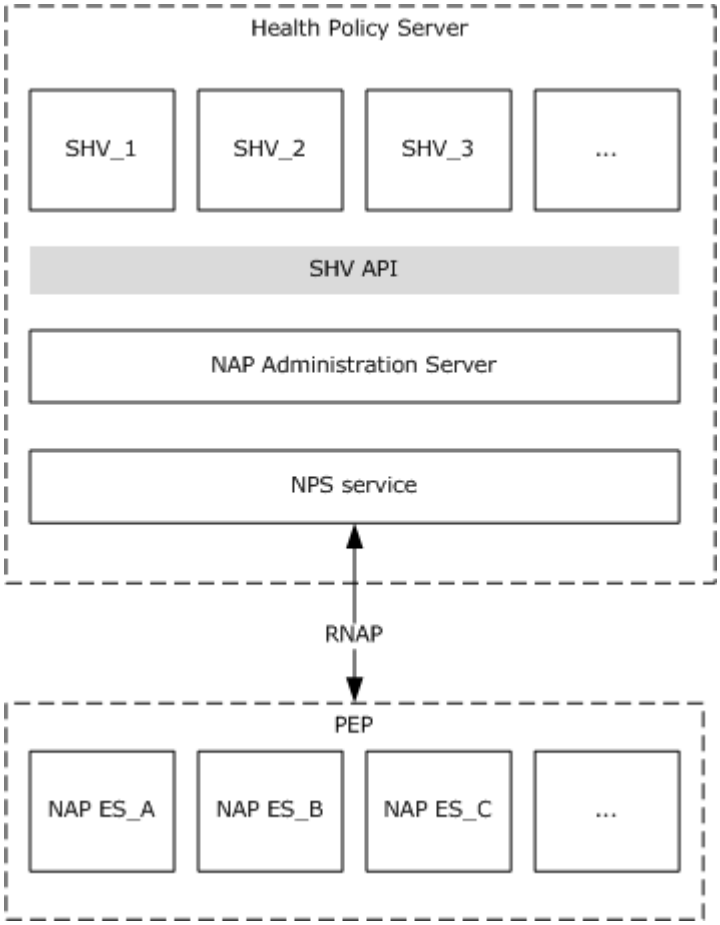

Figure 3: NAP server-side architecture

The NAP server-side architecture consists of NAP enforcement points and a NAP health policy server. A Windows-based NAP enforcement point has a layer of **NAP enforcement server (NAP ES)** components. Each NAP ES is defined for a different type of network access or communication. For example, there is a NAP ES for remote access VPN connections and a NAP ES for DHCP configuration. The NAP ES is typically matched to a specific type of NAP-capable client. For example, the DHCP NAP ES is designed to work with a DHCP-based NAP client. Third-party software vendors or Microsoft can provide additional NAP ESs for the NAP platform.

A NAP ES obtains the SoH from its corresponding NAP EC and sends it to a NAP health policy server transported as a RADIUS VSA of a RADIUS Access-Request message.

The NAP health policy server has the following components:

- The NAP health policy server (NPS) receives the RADIUS Access-Request message, extracts the SoH, and passes it to the **NAP administration server** component.
- The NAP administration server facilitates communication between the NPS and the SHVs. The NAP administration server component is provided with the NAP platform.

- A layer of system health validator (SHV) components, where each SHV is defined for one or multiple types of system health elements and can be matched to an SHA. For example, there might be an SHV for an antivirus program. An SHV can be matched to one or multiple **health requirement servers**. For example, an SHV for checking antivirus signatures is matched to the server that contains the latest signature file. SHVs do not have to have a corresponding health requirement server. For example, an SHV can just instruct NAP-capable clients to check local system settings to ensure that a host-based firewall is enabled.
- The SHV API provides a set of function calls that allow SHVs to register with the NAP administration server component, receive SoHs from the NAP administration server component, and send SoHRs to the NAP administration server component. The SHV API is provided with the NAP platform. For information about these APIs, see [MSDN-NAPAPI].

The most common configuration for NAP server-side infrastructure will consist of NAP enforcement points providing network access or communication of a specific type and separate NAP health policy servers providing system health validation and remediation. It is possible to install the NPS on individual Windows-based NAP enforcement points. However, in this configuration, each NAP enforcement point must then be separately configured with network access and health policies.

3.3.3 Interactions Between Computers and Devices in a NAP-Enabled Network

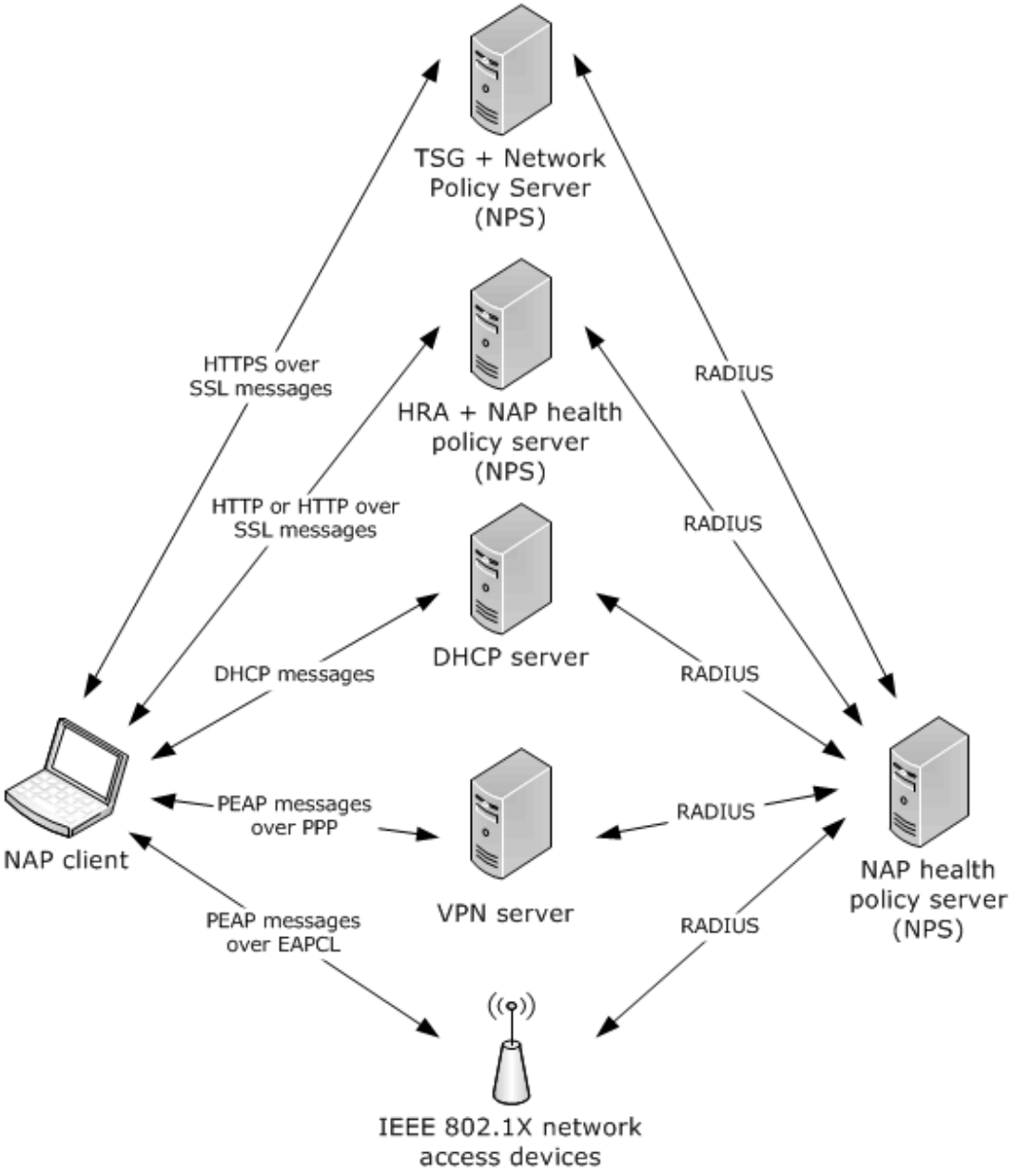

Figure 4: Interactions between NAP platform components

The interactions for the computers and devices of a NAP-enabled network infrastructure are the following:

- Between a NAP client and a **health policy server**

  The NAP client uses the Hypertext Transfer Protocol (HTTP) or an HTTP over Secure Sockets Layer (SSL) protected session to send its current system health state to the health policy server and request a health certificate. The HRA uses a **Health Certificate Enrollment Protocol**

(HCEP) protocol to send an SoHR message and remediation instructions (if the NAP client is noncompliant) and health certificate if the NAP client is compliant.

- Between a NAP client and an 802.1X network access device (an Ethernet switch or a wireless access point)

  The NAP client acting as an 802.1X client uses PEAP messages sent over EAP over LAN (EAPOL) to perform authentication of the 802.1X connection and to indicate its current system health state to the NAP health policy server. An 802.1X client is also known as an 802.1X supplicant. The NAP health policy server uses PEAP messages to either indicate remediation instructions (because the 802.1X client is noncompliant) or that the 802.1X client has unlimited access to the network. PEAP messages between the 802.1X client and NAP health policy server are routed through the 802.1X network access device.

- Between a NAP client and a VPN server

  The NAP client acting as a **VPN client** uses Point-to-Point Protocol (PPP) messages to establish a remote access VPN connection and PEAP messages over the PPP connection to indicate its current system health state to the NAP health policy server. The NAP health policy server uses PEAP messages to either indicate remediation instructions (because the VPN client is noncompliant) or that the VPN client has unlimited access to the intranet. PEAP messages between the VPN client and NAP health policy server are routed through the VPN server.

- Between a NAP client and a DHCP server

  The NAP client acting as a DHCP client uses DHCP messages to obtain a valid IPv4 address configuration and to indicate its current system health state. The DHCP server uses DHCP messages to allocate either an IPv4 address configuration for the restricted network and indicate remediation instructions (if the DHCP client is noncompliant), or an IPv4 address configuration for unlimited access (if the DHCP client is compliant).

- Between a NAP client and a TSG server

  The NAP client, acting as a TSG client, uses messages sent over HTTPS to obtain a connection to the server. The TSG server uses messages sent over HTTPS to allow the connection (if the TSG client is compliant) or deny the connection (if the TSG client is noncompliant).

- Between a NAP client and a remediation server

  While the NAP client has unlimited access to the intranet, it accesses the remediation server to ensure that it remains compliant. For example, the NAP client periodically checks an antivirus server to ensure that it has the latest antivirus signature file or a software update server, such as Windows Server Update Services, to ensure that it has the latest operating system updates.

  If the NAP client has limited access, it can communicate with the remediation server to become compliant, based on instructions from the NAP health policy server. For example, if during the health validation process the NAP health policy server determined that the NAP client does not have the most current antivirus signature file, the NAP health policy server instructs the NAP client to update its local signature file with the latest file that is stored on a specified antivirus server.

- Between one NAP health policy server and another NAP health policy server

  A NAP health policy server can forward messages using RADIUS to another NAP health policy server, i.e. it can act as a RADIUS proxy (this includes any authentication). The first NAP health policy server sends RADIUS messages to the second NAP health policy server which then processes the statement of health messages and then sends back Access-Accept or Accept-

Reject based on the outcome of RADIUS authentication which contain the corresponding SoHR message. The first NAP health policy server in the chain receives back a RADIUS message which includes both an SoHR message and a policy decision which it then forwards to the corresponding PEP.

- Between an HRA and a Certificate Authority

  A Health Registration Authority uses X.509 certificates obtained from a certificate authority to satisfy the request for a certificate using HCEP from compliant NAP clients.

- Between a NAP client and an HRA

  The NAP client uses the HyperText Transfer Protocol (HTTP) or an HTTP over Secure Sockets Layer (SSL) protected session to send its current system health state to the HRA and request a health certificate. The HRA uses HTTP or the protected HTTP over SSL session to send remediation instructions (if the NAP client is noncompliant) or a health certificate to the NAP client.

- Between an 802.1X network access device and a NAP health policy server

  The 802.1X network access device sends RADIUS messages to transfer PEAP messages sent by an 802.1X NAP client.

  The NAP health policy server sends RADIUS messages to:

  - Indicate that the 802.1X client has unlimited access because it is compliant.
  - Indicate a limited access profile to place the 802.1X client on the restricted network until it performs a set of remediation functions. A limited access profile can consist of a set of IP packet filters or a virtual LAN (VLAN) identifier (ID) to confine the traffic of a noncompliant 802.1X client.
  - Send PEAP messages to the 802.1X client.

- Between a VPN server and a NAP health policy server

  The VPN server sends RADIUS messages to transfer PEAP messages sent by a VPN-based NAP client. The NAP health policy server sends RADIUS messages to:

  - Indicate that the VPN client has unlimited access because it is compliant.
  - Indicate that the VPN client has limited access through a set of IP packet filters that are applied to the VPN connection.
  - Send PEAP messages to the VPN client.

  Like the HRA, the VPN server uses the NPS as a RADIUS proxy to exchange RADIUS messages with the NAP health policy server.

- Between a DHCP server and a NAP health policy server

  The DHCP server sends RADIUS messages to the NAP health policy server that contains the DHCP client's system health state.

  The NAP health policy server sends RADIUS messages to the DHCP server to:

  - Indicate that the DHCP client has unlimited access because it is compliant.

- Indicate that the DHCP client has limited access until it performs a set of remediation functions.

A DHCP server can use the NPS as a RADIUS proxy to exchange RADIUS messages with a NAP health policy server.

- Between a TSG server and a NAP health policy server

  The TSG server sends RADIUS messages to the NAP health policy server that contains the TSG client's system health state.

  The NAP health policy server sends RADIUS messages to the TSG server to:

  - Indicate that the TSG client has unlimited access because it is compliant.
  - Indicate that the TSG client has limited access until it performs a set of remediation functions.

  A TSG server can use the NPS as a RADIUS proxy to exchange RADIUS messages with a NAP health policy server.

- Between a DHCP server and a NAP health policy server

  When performing network access validation for a NAP client, the NAP health policy server might have to contact a health requirement server to obtain information about the current requirements for system health. For example, the NAP health policy server might have to contact an antivirus server to check for the version of the latest signature file or to contact a software update server to obtain the date of the last set of operating system updates. The following figure summarizes these interactions.

  The exception to this set of interactions is when a Windows-based NAP enforcement point (the HRA, the VPN server, or the DHCP server) is also acting as a NAP health policy server. In this case, the NAP enforcement point and the NAP health policy server is the same computer. This configuration is appropriate for a small network configuration in conjunction with a single-server networking infrastructure device. However, on an enterprise network, there are usually multiple DHCP servers and typically multiple VPN servers. In this case, using a separate NAP health policy server allows centralization of the configuration of network access and system health requirement policies, rather than configuring them at each NAP enforcement point.

4 Common Task Information

This section contains information that is common to many of the tasks described in this document.

4.1 Common Architectural Details

This section contains information that is common to many of the tasks described in this document.

4.1.1 Common Abstract Data Model

This section describes state established, used, and maintained by processing rules for many tasks described in this document.

NAP Available SHAs List: The NAP agent compiles a list of available SHAs in the system. The mapping is a tuple: a 32-bit unsigned integer representing the SHV **System-Health-ID** and a reference to the SHV. The **System-Health-ID** consists of constants that never change for a given SHV/SHA pair and which represent the IANA SMI Code for the vendor of the SHA/SHV pair.

ShaTimeoutInMsec: A **DWORD** that specifies the timeout value for the call by the NAP agent to the SHA, in milliseconds. The default value for this ADM element is 2000 milliseconds.

PEP Channel Used: An enumeration { HCEP, DHCP, TSG, PEAP } that specifies the **PEP channel** protocol used.

4.2 Common Task: Proxy EAP Payload to RADIUS

This section describes the Proxy EAP Payload to RADIUS Task. This task is described only as a documentation convenience, to clarify the descriptions of other tasks in this System.

The Proxy EAP Payload to RADIUS Task is used when it is necessary to transfer EAP payloads transported by PPP [RFC1661] or 802.1X [IEEE802.1X] to RADIUS. This task performs the actions described in [RFC3748] section 2.3.

4.2.1 Task Overview

4.2.1.1 Task Purpose

The purpose of the task is to build a RADIUS message containing an EAP-Message with the payload of an EAP message, and as a result, make the PEP act as a pass-through device.

4.2.1.2 Task Applicability

This task is used when an EAP message is received to the NAP enforcement server (NAP ES) on the PEP that manages EAP over PPP or EAPOL.

4.2.1.3 Task Use Cases

4.2.1.3.1 Stakeholders and Interests Summary

The stakeholders for the task are as follows:

PEP: The primary interest of the PEP is to pass-through the EAP messages from the EAP peer to the RADIUS server.

4.2.1.3.2 Supporting Actors and Task Interests Summary

None.

4.2.1.3.3 Use Case Diagrams

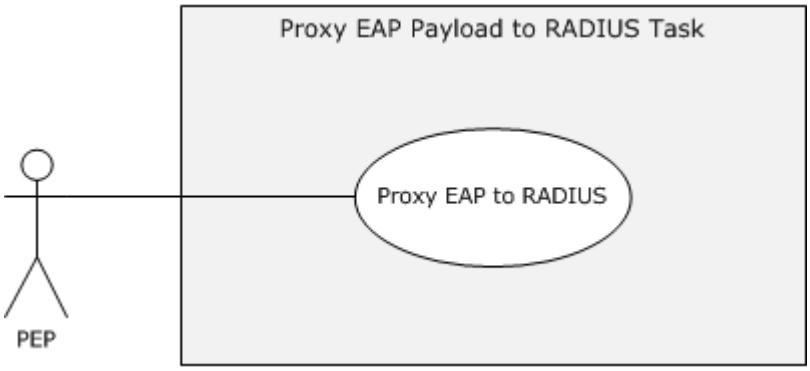

Figure 5: Proxy EAP Payload to RADIUS Task use case diagram

4.2.1.3.4 Use Case: Receive EAP message

This use case is associated with the use case diagram in section 4.2.1.3.3.

Goal: Pass-through an EAP message over RADIUS from the EAP peer to the RADIUS server.

Context of use: An EAP message is received by the PEP.

Direct actor: The PEP.

Primary actor: The PEP.

Supporting Actors: None.

Stakeholders and Interests: All stakeholders are as specified in section 4.2.1.3.1.

Preconditions: None.

Minimal Guarantees: None.

Success Guarantee: A RADIUS message is built.

Trigger: An EAP message is received by the NAP enforcement server (NAP ES) on the PEP that manages EAP.

Main Success Scenario: An EAP message is transferred from the EAP peer to the RADIUS server.

Extensions: None.

4.2.2 Task Context

This section describes the relationship between this task and its environment.

4.2.2.1 Task Environment

This task is accomplished by the PEP. The task does not depend on any external entity.

4.2.2.2 Task Relationships

4.2.2.2.1 Black-Box Relationship Diagrams

Figure 6: Proxy EAP Payload to RADIUS Task black-box relationships

4.2.2.2.2 Task Dependencies

The dependencies are as follows:

The Proxy SoH Task (section 7) provides to this task the EAP message and uses the resulting RADIUS message.

4.2.2.2.3 Task Influences

None.

4.2.2.3 Task Assumptions and Preconditions

None.

4.2.2.4 Task Versioning and Capability Negotiation

The system does not define any versioning or capability negotiation beyond those described in the specifications of the protocols supported by the system.

4.2.3 Task Architecture

This section describes the structure of the task and the interrelationships among its parts.

4.2.3.1 Task Architectural Constraints

None.

4.2.3.2 Task Abstract Data Model

This section describes the state established, used, and maintained by processing rules of this task. State may be volatile or persisted and may pertain to one, some, or all instances of the task. The task's state consists of the values of the named data elements (also called state variables) presented in this section. The overall organization of the data elements, with their names, is the Abstract Data Model. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Data Model, such association can be achieved in other ways. Implementations may depart from this model so long as their external behavior remains consistent with that described in this document.

RADIUS-message: A RADIUS message as specified in [RFC2865] section 3.

4.2.3.3 Task Abstract Parameters

This section describes data passed to an instance of this task at the time it is invoked or triggered. The parameters consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Parameter. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Parameters, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

T-Protocol: An abstract parameter of type Transport Protocol which is used to encapsulate an EAP packet using PPP [RFC1661] or 802.1X [IEEE802.1X].

temp-EAP: An EAP packet as specified in [MS-PEAP] section 2.2.1.

4.2.3.4 Task Abstract Results

This section describes data returned by an instance of this task to its caller. The results consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Result. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Results, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

After the processing rules have been performed according to [RFC3748] section 2.3, the returned result is the **RADIUS-message** ADM element encapsulating the packet in the **temp-EAP** abstract parameter for transport over RADIUS. During pass-through processing, the EAP packet is encrypted.

4.2.3.5 White-Box Relationships

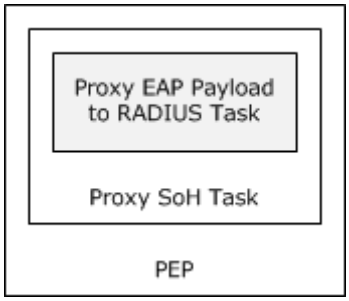

Figure 7: Proxy EAP Payload to RADIUS Task white-box relationships

4.2.3.6 Task Events

4.2.3.6.1 Task Timers

None.

4.2.3.6.2 Task Non-Timer Events

None.

4.2.3.7 Task Architecture and Communication

The task communicates only with the PEP.

4.2.3.8 Task Processing Rules

Build a RADIUS message according to [RFC3579] section 3.1 and store it in the **RADIUS-message** ADM element.

4.2.3.9 Task Failure Scenarios

A malformed EAP message may raise an error. The caller of this task will not receive a RADIUS message and should discard the received EAP message and abort any further processing of the EAP message.

4.2.4 Task Details

4.2.4.1 Task Precondition Details

The caller of the task successfully obtained the EAP message.

4.2.4.2 Task Initialization of External Entities

None.

4.2.4.3 Task Event Details

4.2.4.3.1 Task Timer Details

None.

4.2.4.3.2 Task Non-Timer Event Details

None.

4.2.4.4 Task Architectural Details

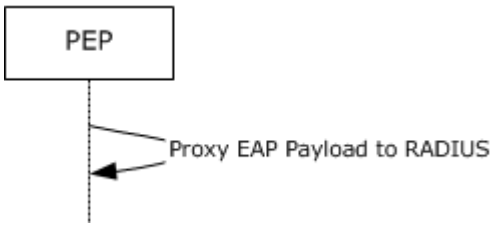

Figure 8: Sequence diagram for the Proxy EAP Payload to RADIUS Task main success scenario

4.2.4.5 Task Processing Rule Details

Build a RADIUS message according to [RFC3579] section 3.1 and store it in the **RADIUS-message** ADM element.

4.2.5 Task Security

For additional information about security considerations, see section 16, as well as the Security sections of the referenced protocol Technical Documents.

4.3 Common Task: Proxy EAP Payload from RADIUS

This section describes the Proxy EAP Payload from RADIUS Task. This task is described only as a documentation convenience, to clarify the descriptions of other tasks in this System.

The Proxy EAP Payload from RADIUS Task is used when it is necessary to transfer EAP payloads transported by RADIUS to PPP [RFC1661] or 802.1X [IEEE802.1X]. This task performs the pass-through action described in [RFC3748] section 2.3.

4.3.1 Task Overview

4.3.1.1 Task Purpose

The purpose of the task is to build an EAP message based on a RADIUS message containing the EAP-message attribute, and as a result, make the PEP act as a pass-through device.

4.3.1.2 Task Applicability

This task is used when a RADIUS message is received by the **NAP enforcement server (NAP ES)** on the PEP that manages EAP over PPP or EAPOL.

4.3.1.3 Task Use Cases

4.3.1.3.1 Stakeholders and Interests Summary

The stakeholders for the task are as follows:

PEP: The primary interest of the PEP is to pass-through the EAP messages from the RADIUS server to the EAP peer.

4.3.1.3.2 Supporting Actors and Task Interests Summary

None.

4.3.1.3.3 Use Case Diagrams

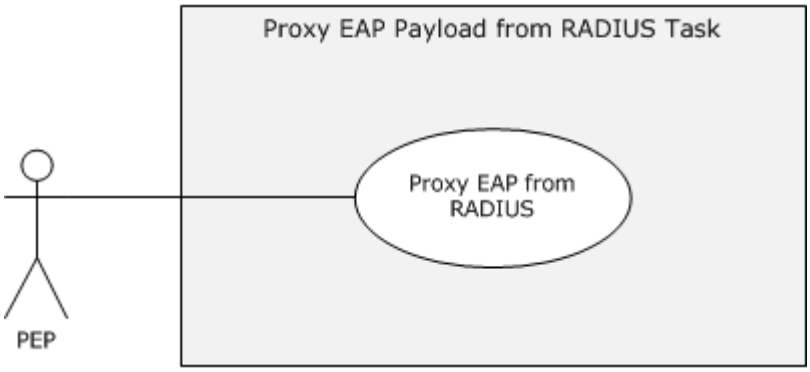

Figure 9: Proxy EAP Payload from RADIUS Task use case diagram

4.3.1.3.4 Use Case: Receive RADIUS message

This use case is associated with the use case diagram in section 4.3.1.3.3.

Goal: Pass-through an EAP message over RADIUS from the RADIUS server to the EAP peer.

Context of use: A RADIUS message is received by the PEP.

Direct actor: The PEP.

Primary actor: The PEP.

Supporting Actors: None.

Stakeholders and Interests: All stakeholders are as specified in section 4.3.1.3.1.

Preconditions: None.

Minimal Guarantees: None.

Success Guarantee: An EAP message is built.

Trigger: A RADIUS message is received by the **NAP enforcement server (NAP ES)** on the PEP that manages EAP.

Main Success Scenario: An EAP message is transferred from the RADIUS server to the EAP peer.

Extensions: None.

4.3.2 Task Context

This section describes the relationship between this task and its environment.

4.3.2.1 Task Environment

This task is accomplished by the PEP. The task does not depend on any external entity.

4.3.2.2 Task Relationships

4.3.2.2.1 Black-Box Relationship Diagrams

Figure 10: Proxy EAP Payload from RADIUS Task black-box relationships

4.3.2.2.2 Task Dependencies

The dependencies are as follows:

The Proxy SoHR Task (section 12) provides to this task the RADIUS message and uses the resulting EAP message.

4.3.2.2.3 Task Influences

None.

4.3.2.3 Task Assumptions and Preconditions

None.

4.3.2.4 Task Versioning and Capability Negotiation

The system does not define any versioning or capability negotiation beyond those described in the specifications of the protocols supported by the system.

4.3.3 Task Architecture

This section describes the structure of the task and the interrelationships among its parts.

4.3.3.1 Task Architectural Constraints

None.

4.3.3.2 Task Abstract Data Model

This section describes the state established, used, and maintained by processing rules of this task. State may be volatile or persisted and may pertain to one, some, or all instances of the task. The task's state consists of the values of the named data elements (also called state variables) presented in this section. The overall organization of the data elements, with their names, is the Abstract Data Model. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Data Model, such association can be achieved in other ways. Implementations may depart from this model so long as their external behavior remains consistent with that described in this document.

EAP-message: An EAP packet as specified in [MS-PEAP] section 2.2.1.

4.3.3.3 Task Abstract Parameters

This section describes data passed to an instance of this task at the time it is invoked or triggered. The parameters consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Parameter. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Parameters, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

T-Protocol: An abstract parameter of type Transport Protocol, which is an EAP packet encapsulated in PPP [RFC1661] or 802.1X [IEEE802.1X].

temp-RADIUS: A RADIUS message as specified in [RFC2865] section 3.

4.3.3.4 Task Abstract Results

This section describes data returned by an instance of this task to its caller. The results consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Result. It is intended to facilitate the reader's conceptual

understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Results, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

After the processing rules have been performed according to [RFC3748] section 2.3, the returned result is the **EAP-message** ADM element containing the EAP packet which was encapsulated in the **temp-RADIUS** ADM element.

4.3.3.5 White-Box Relationships

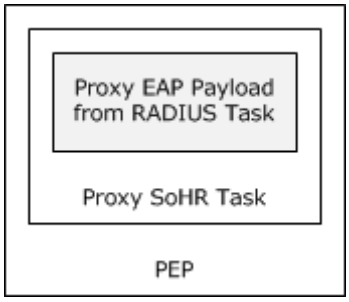

Figure 11: Proxy EAP Payload from RADIUS Task white-box relationships

4.3.3.6 Task Events

4.3.3.6.1 Task Timers

None.

4.3.3.6.2 Task Non-Timer Events

None.

4.3.3.7 Task Architecture and Communication

The task communicates only with the PEP.

4.3.3.8 Task Processing Rules

Build an EAP message according to [RFC3579] section 3.1 and store it in the **EAP-message** ADM element.

4.3.3.9 Task Failure Scenarios

A malformed RADIUS message may raise an error. The caller of this task will not receive an EAP message and should discard the received RADIUS message and abort any further processing of the RADIUS message.

4.3.4 Task Details

4.3.4.1 Task Precondition Details

The caller of the task successfully obtained the RADIUS message.

4.3.4.2 Task Initialization of External Entities

None.

4.3.4.3 Task Event Details

4.3.4.3.1 Task Timer Details

None.

4.3.4.3.2 Task Non-Timer Event Details

None.

4.3.4.4 Task Architectural Details

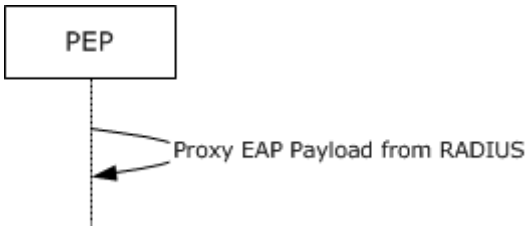

Figure 12: Sequence diagram for the Proxy EAP Payload from RADIUS Task main success scenario

4.3.4.5 Task Processing Rule Details

Build an EAP message according to [RFC3579] section 3.1 and store it in the **EAP-message** ADM element.

4.3.5 Task Security

For additional information about security considerations, see section 16, as well as the Security sections of the referenced protocol Technical Documents.

5 Update NAP Client Configuration Task

This section describes how the NAP agent receives configuration updates from the Configuration Manager and updates the NAP-specific configuration on the client computer. This task updates the local configuration on a domain-joined or non-domain-joined client computer according to changes applied by a Group Policy System [MS-GPSO], as specified in the Network Access Protection (NAP) Extension [MS-GPNAP] or by a client administrator.

Note This task uses the **NAP Client Config** ADM element (section 4.1.1). All other common information defined in section 4 is not applicable to this task.

5.1 Task Overview

5.1.1 Task Purpose

The purpose of this task is to ensure that the client configuration is correctly collected, the client computer is correctly configured, and that the enabled enforcement client (EC) and HRA URLs are updated. The client configuration can be updated manually by the administrator or automatically through a Group Policy client.

5.1.2 Task Applicability

This task is used when the client computer reboots, a timer triggers the NAP agent to poll the Configuration Manager, or when there is a configuration update event.

This task is not applicable if the NAP system is not deployed.

5.1.3 Task Use Cases

5.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Update NAP Client Configuration Task (section 5) are:

NAP agent: The main software component on the NAP client computer. It is responsible for fetching the NAP configuration. The primary interest of the NAP agent is to receive a NAP-specific configuration from the Configuration Manager.

Create and Send SoH Task: The primary purpose of the Create and Send SoH Task (section 6) is to fetch the current NAP-specific configuration from this use case.

NAP Event Handler: A component on the NAP client computer that receives events from underlying layers and passes them to the NAP agent. It ensures that the use case will process the events in this task.

5.1.3.2 Supporting Actors and Task Interests Summary

The supporting actors are:

Configuration Manager: Stores the NAP-specific configuration information on the client computer and sends events to the NAP Event Handler. The task interest is that the NAP agent is able to read the configuration information from the Configuration Manager.

GPNAP client: A Group Policy client as described in [MS-GPSO] that enables a client computer to retrieve NAP-specific policy settings from a **Group Policy server** and update them in the

Configuration Manager, as specified in [MS-GPNAP]. The task employs the GPNAP client to receive NAP-specific policy settings from the Group Policy server.

Configuration Manager Utility: The NAP client user interface. The Configuration Manager Utility provides the client administration tools to retrieve, create, update, and delete NAP-specific configuration information that is stored in the Configuration Manager. The task employs this actor to define the NAP-specific configuration to be stored in the Configuration Manager.

Client administrator: The individual who configures the client computer. The client administrator modifies the NAP-specific configuration by using the Configuration Manager Utility. The task employs this actor to update the configuration information as required.

5.1.3.3 Use Case Diagrams

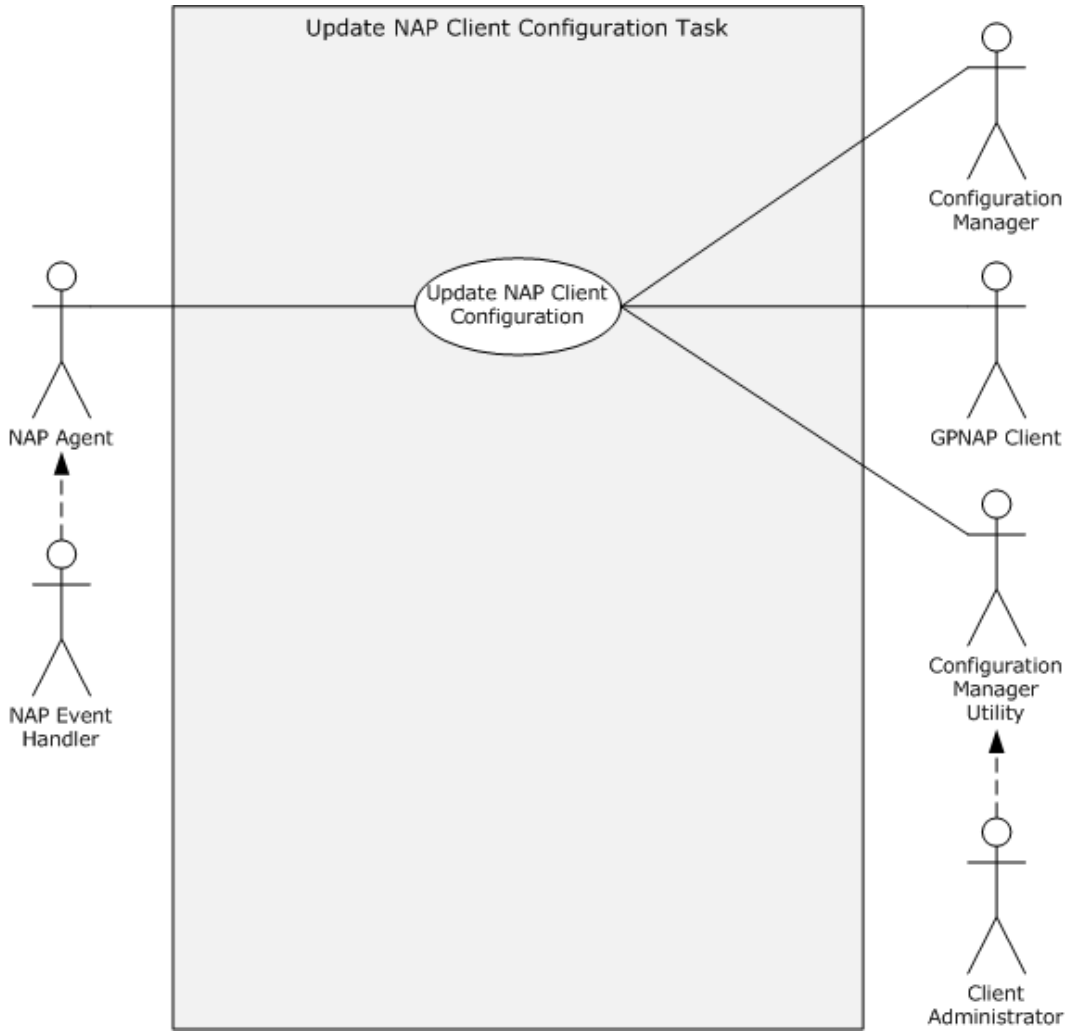

Figure 13: Update NAP Client Configuration use case diagram

5.1.3.4 Use Case: Update NAP Client Configuration -- NAP Agent

Goal: To update the NAP-specific configuration on the NAP agent based on updates to the Configuration Manager.

Context of Use: This use case is initiated when the client computer restarts, when the internal timer in the NAP agent triggers the task, or when there is a configuration update event.

Direct Actor: The direct actor is the NAP agent.

Primary Actor: The primary actor is the NAP Event Handler.

Supporting Actors: The supporting actors are as specified in section 5.1.3.2.

Stakeholders and Interests: The stakeholder for this use case is the Create and Send SoH Task (section 6). The primary interest of this stakeholder is to fetch the most recent NAP-specific configuration from this use case.

Precondition: The NAP client components on the client computer are deployed and configured correctly by the client administrator.

Minimal Guarantees:

1. The NAP Event Handler will always receive the events.
2. The use case will always process the received events.
3. The Configuration Manager maintains the current NAP-specific configuration.
4. The use case will always deliver the current NAP-specific configuration.

Success Guarantee: The NAP client configuration is up-to-date.

Trigger: The Update NAP Client Configuration Task (section 5) can be triggered by any of the following:

- The client computer restarts.
- An update in the NAP-specific configuration.
- A timer triggers the NAP agent to poll the Configuration Manager.

Main Success Scenario:

1. A configuration update to the Configuration Manager occurs as follows:
    1. The GPNAP client retrieves the NAP-specific configuration from the Group Policy System, as specified in [MS-GPSO], and updates the Configuration Manager, which will send an event to the NAP Event Handler.
    2. The client administrator updates the Configuration Manager via the Configuration Manager Utility.
2. Or, one of the following occurs on the client computer:
    - The NAP Event Handler receives an event indicating that the client computer has restarted.
    - A timer triggers the NAP agent to poll the Configuration Manager.

3. The NAP agent reads the NAP configuration information from the Configuration Manager.
4. The NAP agent updates the client configuration information in the ADM elements specified in section 4.1.1 if it detects a change.
5. The NAP client configuration is up-to-date.

Extensions: None.

5.2 Task Context

This section describes the relationship between this task and its environment.

5.2.1 Task Environment

This task is accomplished by the NAP client in an environment where the NAP system configuration may be changed. The environment should meet several requirements to support this task.

- **Requirement:** The client computer has network access to the Group Policy server.
  - **Reason for requirement:** To retrieve Group Policy: NAP Extension values (as specified in [MS-GPNAP]) via the Group Policy: Registry Extension Encoding (specified in [MS-GPREG]), access to the Group Policy server via the network is required.
  - **Satisfying the requirement:**
    1. The network interface of the client computer is configured to operate on the local subnet.
    2. The physical network path (network devices, Ethernet cables, and so on) between the local subnet and the Group Policy server is connected.
    3. All network devices between the local subnet and the Group Policy server are configured to allow packet flow between the two entities.
    4. The routing tables in the client computer are configured to enable correct packet routing between the client computer and the Group Policy server.
  - **Verifying requirement is satisfied:** The client computer can successfully ping the Group Policy server over the network.
  - **Consequences of not satisfying requirement:** The task is unable to receive Group Policy: NAP Extension values. Only manual updates to the Configuration Manager are possible.
- **Requirement:** The Configuration Manager has access to Windows registry.
  - **Reason for requirement:** To update the ADM elements specified in section 4.1.1, access to Windows registry is required.
  - **Satisfying the requirement:** The Configuration Manager is able to access the client registry using registry access APIs.
  - **Verifying requirement is satisfied:** No log entries regarding a failure to access the registry.
  - **Consequences of not satisfying requirement:** If the NAP agent is unable to read configuration information from the Configuration Manager, this will prevent the NAP agent from functioning.

Unless explicitly specified otherwise, the task implementation may assume its environment is properly configured and is not expected to verify that every requirement is satisfied. On the other hand, the task implementation should be able to gracefully handle errors that may be caused by environment misconfiguration or temporary dysfunction. The implementation should log these errors along with relevant information to allow for troubleshooting.

5.2.2 Task Relationships

5.2.2.1 Black-Box Relationship Diagrams

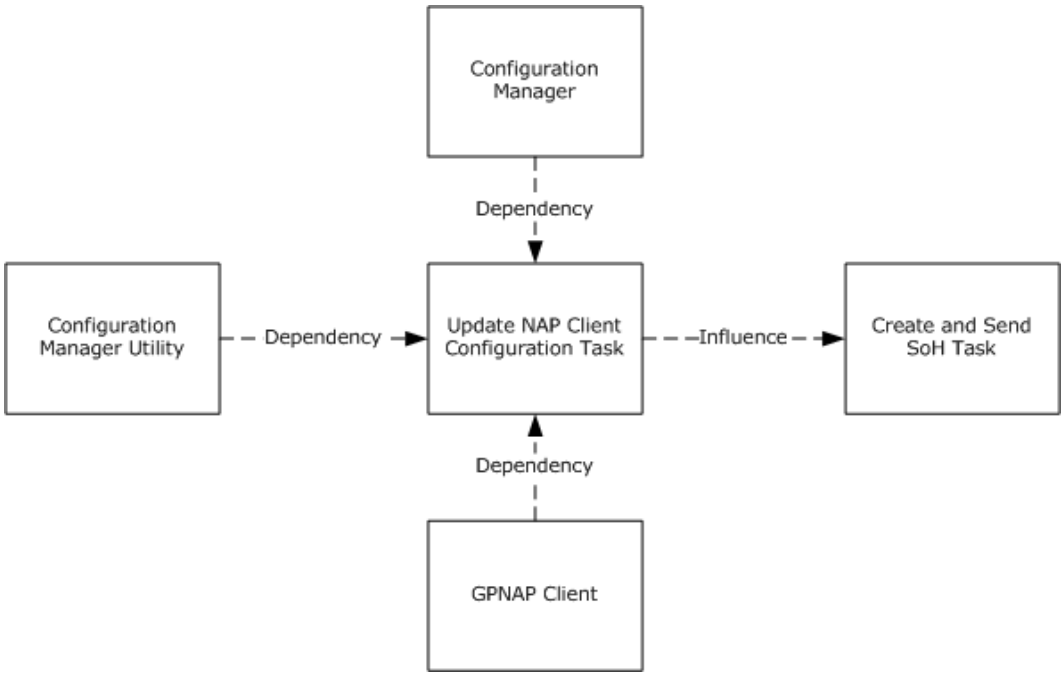

Figure 14: Update NAP Client Configuration Task black-box relationships

The NAP agent updates its configuration from the Configuration Manager on the client computer which was updated manually by the client administrator or the GPNAP client.

5.2.2.2 Task Dependencies

The Update NAP Client Configuration Task (section 5) is dependent on the Group Policy server or the Configuration Manager Utility to have the updated configuration saved to the Configuration Manager.

The Update NAP Client Configuration Task is dependent on the Configuration Manager to be able to read the updated configuration.

5.2.2.3 Task Influences

The Update NAP Client Configuration Task influences which transport protocol is used by the Create and Send SoH Task to send the SoH message. If the Update NAP Client Configuration Task fails, the NAP client configuration may be incorrect. In other words, the client will not have any ECs, or the correct EC will not be enabled. As a result, the Create and Send SoH Task may not be able to send the SoH or it could use the wrong configuration for sending. For information about possible failure scenarios, see section 5.3.9.

5.2.3 Task Assumptions and Preconditions

To accomplish this task, the NAP client has the following preconditions and assumptions:

- The operating system and hardware comprising on the client computer is trustworthy.
- The NAP client is enabled on the client computer.
- The NAP client can access local storage on the client computer.

5.2.4 Task Versioning and Capability Negotiation

The Update NAP Client Configuration Task does not define any versioning and capability negotiation beyond those described in the specifications of the protocols supported or used by the task, as listed in section 2.3.

5.3 Task Architecture

This section describes the structure of the Update NAP Client Configuration Task and the interrelationships among its parts.

5.3.1 Task Architectural Constraints

There is only one instance of the Update NAP Client Configuration Task on each client computer and this instance initializes itself each time it starts. Different instances of this task on different client computers can run independently.

5.3.2 Task Abstract Data Model

This section describes the state established, used, and maintained by processing rules of this task. State may be volatile or persisted and may pertain to one, some, or all instances of the task. The task's state consists of the values of the named data elements (also called state variables) presented in this section. The overall organization of the data elements, with their names, is the Abstract Data Model. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Data Model, such association can be achieved in other ways. Implementations may depart from this model so long as their external behavior remains consistent with that described in this document.

ConfigurationReadInterval: A timer that is used by this task to periodically check the configuration. The timer is a DWORD that represents the time intervals between consecutive retrievals of information from the Configuration Manager and is initialized to 5 minutes.

Enabled EC List: A list of objects, in which each contains the ID (DWORD) of an EC to use and a Boolean representing whether or not the EC is enabled. An instance of this ADM element is created for each enabled enforcement client received in Enforcement Client Settings ([MS-GPNAP] section 2.3). Possible values for the enforcement types include:

- DHCP Enforcement - Indicates that health policies should be enforced when a client computer attempts to obtain an IP address from a DHCP server, as described in [MS-GPNAP] section 2.3.1.
- Remote Access Enforcement - Indicates that health policies should be enforced when a client computer attempts to gain access to the network through a virtual private network (VPN) connection, as described in [MS-GPNAP] section 2.3.2.

- IPsec Enforcement - Indicates that health policies should be enforced when a client computer attempts to communicate with another computer using IPsec, as described in [MS-GPNAP] section 2.3.3.
- RDG Enforcement - Indicates that health policies should be enforced when a client computer attempts to gain access to an RDG, as described in [MS-GPNAP] section 2.3.4.
- Wireless EAPOL Enforcement / EAP Enforcement - Indicates that health policies should be enforced when a client computer attempts to access a network through an 802.1X wireless connection or an authenticating switch connection, as described in [MS-GPNAP] section 2.3.5.

This ADM element is initialized as an empty list. When an EC is detected in the Configuration Manager, the ID of the EC is added to this ADM element along with its state. When the NAP agent cannot access the Configuration Manager or the Configuration Manager is deleted, no EC will be available, and therefore, this ADM element is not initialized. The information in this ADM element is used by the NAP agent to create an ADM element that is used in the Create and Send SoH Task (section 6.3.3).

IPsec HRA List: This ADM is initialized to an empty string. When IPsec enforcement client is enabled, this ADM element is created to contain a list of strings, where each string indicates an available IPsec HRA server URL as specified in [MS-GPNAP] section 2.4.4. When the NAP agent cannot access the Configuration Manager or the Configuration Manager is deleted, this ADM element will not be initialized and the IPsec EC will not be able to send the SoH. The information in this ADM element is used by the NAP agent to send HCEP messages in the Create and Send SoH Task (section 6.3.3).

NAP Client Config: Contains configuration information used by the NAP client. The ADM element is initialized with default values. When the NAP agent reads the configuration from local storage, the NAP agent stores the configuration information in this ADM element, as described in section 5. The following NAP configuration information is stored in this ADM element:

- Trace settings - The NAP client tracing functionality settings that are compounded from the following settings:
  - Enable Tracing - An indicator that specifies whether tracing is enabled on the NAP client. The data type and range of values are specified in [MS-GPNAP] section 2.1.1.
  - Tracing Level - Holds the tracing level on the NAP client. The data type and range of values are specified in [MS-GPNAP] section 2.1.2.
- User Interface settings - UI display information settings that are used by the NAP client UI and are compounded from the following settings:
  - SmallText - Specifies the user notification title displayed to the user. The data type and valid values are specified in [MS-GPNAP] section 2.2.1.
  - LargeText - Specifies the user notification subtitle displayed to the user. The data type and valid values are specified in [MS-GPNAP] section 2.2.2.
  - ImageFile - Represents an image that is displayed in the NAP client user interface. The data type and valid values are specified in [MS-GPNAP] section 2.2.3.
  - ImageFileName - Used to determine the format of the image data specified in ImageFile. The data type and valid values are specified in [MS-GPNAP] section 2.2.4.

- Health Registration Authority (HRA) settings - The NAP HRA settings are used by the NAP client in the Create and Send SoH Task (section 6) for HCEA EC and are compounded from the following settings:
  - PKCS#10 Certificate settings - Security parameters to construct the Public Key Cryptography Standards (PKCS) #10 certificate request, as specified in [MS-HCEP] section 2.2.1.4. The following parameters are used when the IPsec enforcement client is enabled:
    - Cryptographic Service Provider (CSP) - The name of the cryptographic service provider (CSP) that is used to generate the key pair on the HCEA. Data type and valid values are specified in [MS-GPNAP] section 2.4.1.1.
    - Cryptographic Provider Type - The type of the cryptographic service provider (CSP) that is used to generate the key pair on the HCEA. The data type and range of values are specified in [MS-GPNAP] section 2.4.1.2.
    - Public Key OID - An object identifier (OID) that identifies the algorithm of the public-private key pair associated with the certificate. The data type and valid values are specified in [MS-GPNAP] section 2.4.1.3.
    - Public Key Length - The key length of the public-private key pair associated with the certificate. The data type and range of values are specified in [MS-GPNAP] section 2.4.1.4.
    - Public Key Spec - The public key associated with the certificate. The data type and range of values are specified in [MS-GPNAP] section 2.4.1.5.
    - Hash Algorithm OID - An OID identifying the hash algorithm used to sign the certificate request. The data type and valid values are specified in [MS-GPNAP] section 2.4.1.6.
  - HRA Auto-Discovery - An indicator that specifies whether HRA Auto-Discovery is enabled in the NAP client. The data type and range of values are specified in [MS-GPNAP] section 2.4.2.
  - Use SSL - An indicator that specifies whether SSL is enabled for communication with the HRA. The data type and range of values are specified in [MS-GPNAP] section 2.4.3.
  - Reconnect Attempts settings - The time (in minutes) that the client should wait before attempting to reconnect to an HRA in the event of a connection failure. The data type and range of values are specified in [MS-GPNAP] section 2.4.5.
- SoH settings - These settings include a Backward Compatible indicator which is used for backward compatibility by the NAP client in the Create and Send SoH Task (section 6). The data type and range of values are specified in [MS-GPNAP] section 2.5.

5.3.3 Task Abstract Parameters

None.

5.3.4 Task Abstract Results

This section describes data returned by an instance of this task to its caller. The results consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Result. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Results, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

The Update NAP Client Configuration Task does not return data to its caller.

5.3.5 White-Box Relationships

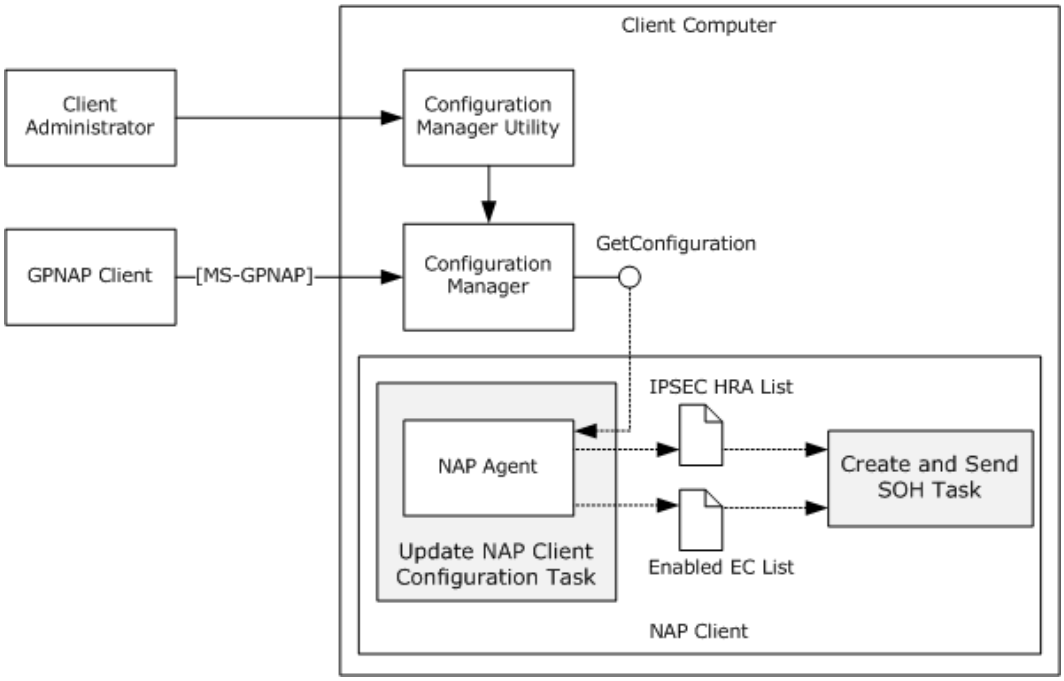

Figure 15: Update NAP Client Configuration Task white-box relationships

The NAP agent contacts the Configuration Manager on the client computer to obtain the latest configuration. The Configuration Manager maintains the configuration in the registry and accesses it using registry access APIs. The configuration is stored in the registry in two locations, one that is updated by the client administrator using the Configuration Manager Utility, and another that is updated by the GPNAP client (see [MS-GPNAP] section 1.4). The GPNAP client has priority over the manual configuration. The **IPSEC HRA List** and **Enabled EC List** ADM elements specified in section 5.3.2 are shared between this task and the Create and Send SoH Task (section 6).

5.3.6 Task Events

5.3.6.1 Task Timers

The Update NAP Client Configuration Task implements the timer specified in the **ConfigurationReadInterval** ADM element (section 5.3.2).

5.3.6.2 Task Non-Timer Events

This task uses one non-timer event: Update Configuration.

5.3.7 Task Architecture and Communication

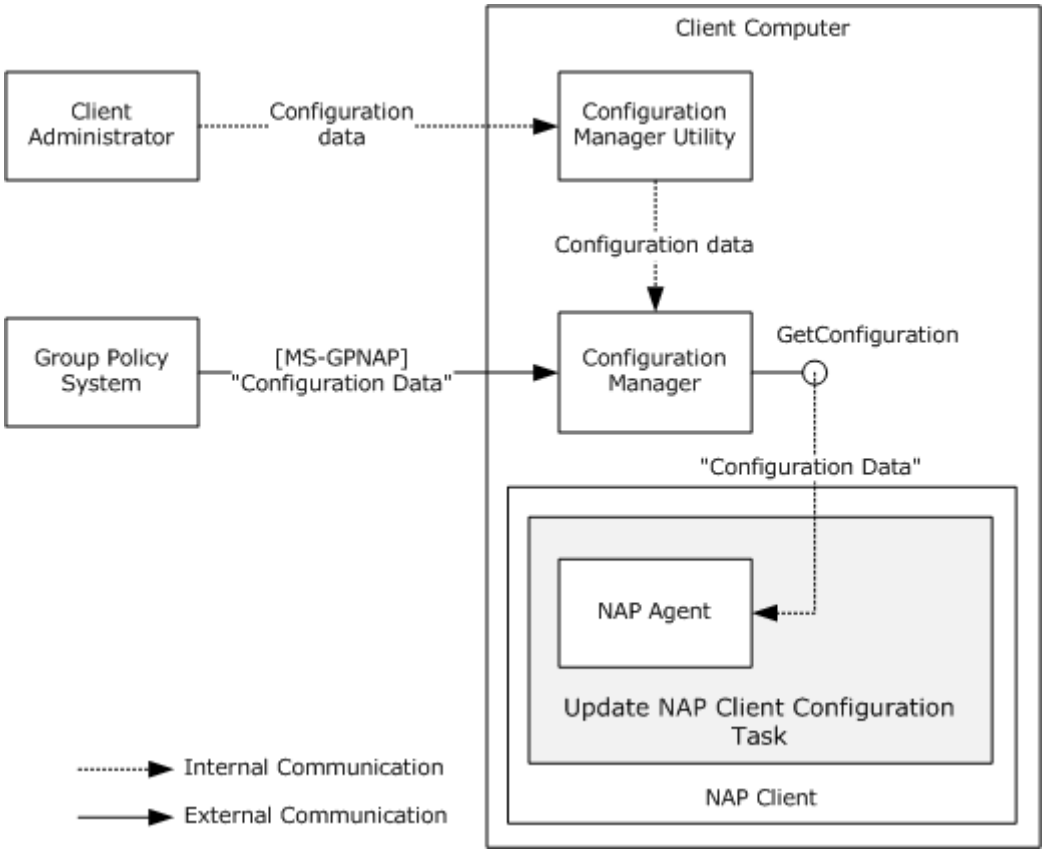

Figure 16: Update NAP Client Configuration Task architecture and communication overview

5.3.8 Task Processing Rules

The following describes the operational flow of the Update NAP Client Configuration Task:

1. A configuration update to the Configuration Manager occurs in one of the following ways:
   - The GPNAP client retrieves the NAP client configuration from the Group Policy System [MS-GPSO] and updates the Configuration Manager, as specified in [MS-GPNAP].
   - The client administrator updates the Configuration Manager via a Configuration Manager Utility.
2. The NAP agent contacts the Configuration Manager every period of time based on the value of the timer interval specified in the **ConfigurationReadInterval** ADM element as specified in section 5.3.6.1 or when an event is received from the NAP Event Handler.
3. The NAP agent obtains the latest configuration information and loads it into memory.
4. The NAP agent compares the values in the cached configuration from the previous step with the values stored in the **NAP Client Config** (section 4.1.1), and **Enabled EC List** and **IPsec HRA**

List ADM elements (section 5.3.2). If changes are detected, the NAP agent updates the values stored in **NAP Client Config**, **Enabled EC List**, and **IPsec HRA List**.

5. The NAP agent waits until the next configuration is read.

5.3.9 Task Failure Scenarios

5.3.9.1 Tasks Fail to Receive System Configuration

If the NAP agent fails to receive the configuration from the Connection Manager, either because the Connection Manager cannot access the registry or because registry values were deleted, the ADM elements passed to the Create and Send SoH Task may contain the wrong configuration for sending the SoH.

5.4 Task Details

This section contains the details that complete the descriptions in earlier sections of the document. These details are needed to understand and implement this task.

5.4.1 Task Precondition Details

For the complete list of task preconditions and assumptions, see section 5.2.3. Details for some of the preconditions are as follows:

- The NAP client is enabled on the client computer.
- The NAP client can access the registry on the client computer.
- If the client computer is configured to receive Group Policy updates, the client computer must be a member of a domain.

5.4.2 Task Initialization of External Entities

None.

5.4.3 Task Event Details

5.4.3.1 Task Timer Details

The Update NAP Client Configuration Task uses the timer stored in the **ConfigurationReadInterval** ADM element (section 5.3.2).

5.4.3.2 Task Non-Timer Event Details

When there is an update in the NAP-specific configuration on the client computer, the task is triggered to update the NAP-specific ADM elements (section 4.1.1) maintained by the NAP agent.

5.4.4 Task Architectural Details

This section illustrates an example of how the NAP agent reads from the Configuration Manager and updates its configuration at the same time that the Configuration Manager is being updated externally by the Group Policy.

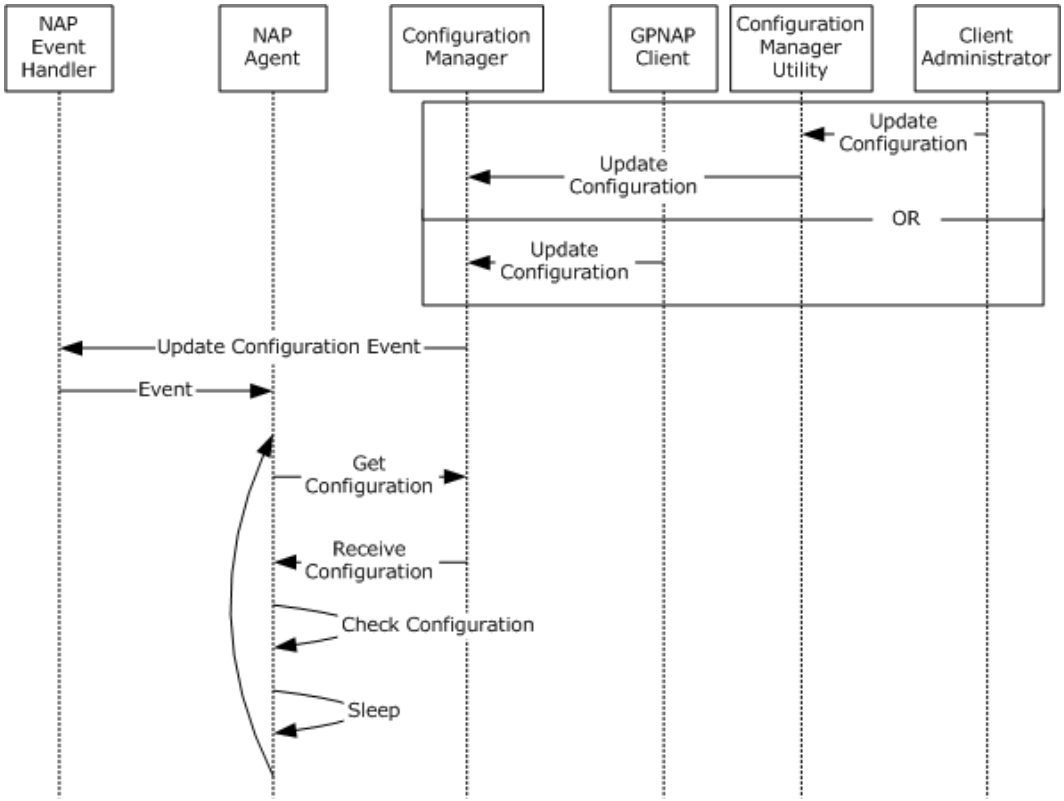

Figure 17: Sequence diagram for the main success scenario of the Update NAP Client Configuration Task

1. The NAP client configuration is updated in the Configuration Manager by the GPNAP client or by the Configuration Manager Utility.
2. An Update Configuration event is sent from the Configuration Manager to the NAP Event Handler. This event will trigger the NAP agent to fetch the configuration from the Configuration Manager.
3. The NAP agent reads the configuration data from the Configuration Manager.
4. The NAP agent reviews the configuration for changes, and updates its configuration if necessary.
5. The NAP agent configuration read process waits until the next configuration read based on the timer interval value stored in the **ConfigurationReadInterval** ADM element.

5.4.5 Task Processing Rule Details

1. A configuration update to the Configuration Manager occurs in one of the following ways:
   - The GPNAP retrieves the NAP client configuration from the Group Policy System [MS-GPSO] and updates registry entries in the Configuration Manager, as specified in [MS-GPNAP].
   - The client administrator updates registry entries in the Configuration Manager via the Configuration Manager Utility.
2. The Configuration Manager sends an event to the NAP Event Handler to notify it about the change.

3. The NAP Event Handler triggers the NAP agent.

4. The NAP agent contacts the Configuration Manager to obtain an updated configuration. The NAP agent also polls the Configuration Manager at certain time intervals to obtain an updated configuration, based on the value of the timer specified in the **ConfigurationReadInterval** ADM element (section 5.3.2).

5. The Configuration Manager reads the registry entries specified in [MS-GPNAP] section 2 using a public registry API and returns the information to the NAP agent.

6. For each registry entry, the NAP agent updates the following ADM elements:

   - The Trace settings specified in [MS-GPNAP] section 2.1, including Enable Tracing and Tracing Level, are stored in the **NAP Client Config** ADM element (section 4.1.1) to be used by the NAP agent. Tracing records NAP events in a log file used for troubleshooting and maintenance. By default, NAP tracing is disabled, so NAP events are recorded in the NAP tracing log file. For more information see [MSFT-CFGNAPTRCNG].

     Tracing level determines the extent of logging, 0x00000003 results in a detailed log.

     Note This NAP configuration information will not affect the protocol's wire behavior.

   - The User Interface Settings specified in [MS-GPNAP] section 2.2, including SmallText, LargeText, ImageFile, and ImageFileName, are stored in the **NAP Client Config** ADM element (section 4.1.1) to be used by the NAP client UI. These settings are used by the NAP client UI to display the following information:

     - **SmallText:** User notification title.
     - **LargeText:** User notification subtitle.
     - **ImageFile:** Image.
     - **ImageFileName:** Specifies the format of the image data specified in **ImageFile**.

       Note The **ImageFile** and **ImageFileName** settings do not affect protocol wire behavior.

   - For each Enforcement Client specified in [MS-GPNAP] section 2.3, the ID of the ECs and their state (enabled or disabled) are added to the **Enabled EC List** ADM element (section 5.3.2). The ID of the EC is used by the NAP client in the Create and Send SoH Task (section 6), where possible values include DHCP Enforcement, Remote Access Enforcement, IPsec Enforcement, Wireless EAPOL Enforcement, RDG Enforcement, and EAP Enforcement.

   - The Health Registration Authority (HRA) settings specified in [MS-GPNAP] section 2.4, including Cryptographic Service Provider (CSP), Cryptographic Provider Type, Public Key OID, Public Key Length, Public Key Spec, Hash Algorithm OID, HRA Auto-Discovery, Use SSL, and Reconnect Attempts, are stored in the **NAP Client Config** ADM element (section 4.1.1).

   - When the IPsec enforcement client is enabled, the **IPsec HRA List** ADM element (section 5.3.2) is created and filled with strings, where each string indicates an available IPsec HRA server URL as specified in [MS-GPNAP] section 2.4.4. If HRA Auto-Discovery, stored in the **NAP Client Config** ADM element (section 4.1.1), is enabled, HRA URLs are discovered automatically by the NAP client using DNS SRV lookup, as described in [MS-GPNAP] section 2.4.2. The information in the **IPsec HRA List** ADM element is used by the NAP client to send HCEP messages in the Create and Send SoH Task (section 6).

7. The NAP agent sleeps according to the value of the timer interval specified in **ConfigurationReadInterval**.

5.5 Task Security

This section documents security issues specific to this task that are not otherwise described in the Technical Documents (TDs) for the protocols used in the task. It does not duplicate what is already in the protocol TDs unless there is some unique aspect that applies to the system as a whole.

The only security considerations for this task is that the NAP agent must be allowed to access the registry on the client computer and only an administrator can change the configuration manually.

6 Create and Send SoH Task

This section describes how the NAP agent creates an SoH on a client computer. This task is initiated when there is a trigger for new health information on a client computer where the NAP components are deployed. For example, when a change of health state occurs, when a client attempts to access a NAP protected network resource, and so on. For more information about possible triggers, see section 6.1.3.4. The NAP Agent requests an SoH message from the SoH Client. The SoH Client uses the services provided by the system health agent (SHA) to create an SoH. The SoH Client collects health evaluation information from each SHA and caches them. The SoH cache is updated whenever an SHA supplies a new or updated SoH. The NAP agent sends the SoH message to the enforcement client which sends it to the Policy Enforcement Server.

The health collection in Windows Security Health Agent (WSHA) follows the protocol defined in [MS-WSH]. The protocols that can be used in this task are specified in the following documents: [TNC-IF-TNCCSPBSoH], [MS-WSH], [MS-HCEP], [MS-DHCPN], [MS-TSGU], [IEEE802.1X], and [MS-PEAP].

Note This task uses the **NAP Client Config** and **NAP Available SHAs List** ADM elements (section 4.1.1). All other common information defined in section 4 is not applicable to this task.

6.1 Task Overview

6.1.1 Task Purpose

The purpose of this task is to ensure that the health information is correctly collected and that the SoH is correctly created on a client computer when a health state change occurs. This task includes but is not limited to health status assessment and SoH composition.

6.1.2 Task Applicability

This task is used when a client computer attempts to access network-based resources and the NAP System is deployed on the client computer. This task is not applicable if the NAP System is not deployed.

6.1.3 Task Use Cases

6.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Create and Send SoH Task are as follows:

NAP Agent: The main software component on the NAP client computer. It is responsible for executing NAP-related operations, such as fetching the NAP configuration, creating the correlation ID, determining which transport protocol to use, and so on. The primary interest of the NAP Agent in this task is that all transport protocol messages are sent by the task with a correlation ID, an SoH packet, and the authentication data.

NAP Event Handler: A component on the NAP client computer that receives events from the underlying layers and passes them to the NAP Agent. The primary interest of the NAP Event Handler in this task is that the use case will always process the received events.

Proxy SoH Task: The purpose of this stakeholder is to proxy the correlation ID, the SoH packet, and the authentication data from the DHCPN, HCEP, TSGU, or PEAP servers to the RNAP client. As such, the primary interest of this stakeholder is to ensure that the Create and Send SoH Task only sends protocol messages that contain all three components.

6.1.3.2 Supporting Actors and Task Interests Summary

The supporting actors are:

SOH Client: The purpose of this actor is to utilize the processing rules defined in [TNC-IF-TNCCSPBSoH] to create SoH packets. This is accomplished by first creating the SSoH header, which is prepended to the SoH packet. The actor then calls the abstract interface of each registered and enabled SHA in turn. Each SHA returns a SoHReportEntry which is appended to the SoH packet. The use case employs this actor whenever a new SoH packet is required.

HCEP HCEA: This protocol client is used to send [MS-HCEP] messages to an HCEP server on the PEP computer. In this use case, when HCEP is used, the HCEP HCEA acts in the role of an enforcement client (EC). This is typically done when the client computer requires an X.509 certificate for use by an IPSec connection to the corporate network. An [MS-HCEP] protocol message exchange can be triggered in other ways, such as an IP Address change. The use case employs this actor whenever an SoH packet has to be sent to the HCEP server.

DHCP Client: This protocol client is used to send [MS-DHCPN] messages to a DHCPN server on the PEP computer. In this use case, when DHCPN is used, the DHCP Client acts in the role of an enforcement client (EC). This is typically done when the client computer uses dynamic addressing and first boots up. The [MS-DHCPN] message is carried as part of the payload in the DHCP messages used to retrieve an IP Address. A [MS-DHCPN] protocol message exchange is also triggered when the lease record expires. The use case employs this actor whenever an SoH packet has to be sent to the DHCPN server.

TSGU Client: This protocol client is used to send [MS-TSGU] messages to a TSGU server on the PEP computer. In this use case, when TSGU is used, the TSGU Client acts in the role of an enforcement client (EC). This is typically done when a user on the client computer attempts to use RDP to access a computer located on the corporate network, which requires going through a terminal services gateway. The use case employs this actor whenever an SoH packet has to be sent to the TSGU server.

PEAP Peer: This protocol client is used to send [MS-PEAP] messages to a PEAP server on the PEP computer. In this use case, when PEAP is used, the PEAP Peer acts in the role of an enforcement client (EC). There are two typical scenarios where a [MS-PEAP] protocol message exchange is triggered. The first scenario occurs when the client computer first boots up and authenticates against an 802.1X device, such as a gateway router. The second scenario occurs when a user on the client computer attempts to VPN into a resource located on the corporate network. A [MS-PEAP] protocol message exchange can be triggered in other ways, such as a change in the NAP configuration. The use case employs this actor whenever an SoH packet has to be sent to the PEAP server.

Configuration Manager: The configuration manager maintains the data store containing the NAP configuration and notifies the **NAP Agent** about configuration changes. The NAP configuration contains elements representing all the settings found in [MS-GPNAP]. The use case contacts the Configuration Manager whenever it is required to get NAP configuration values.

6.1.3.3 Use Case Diagrams

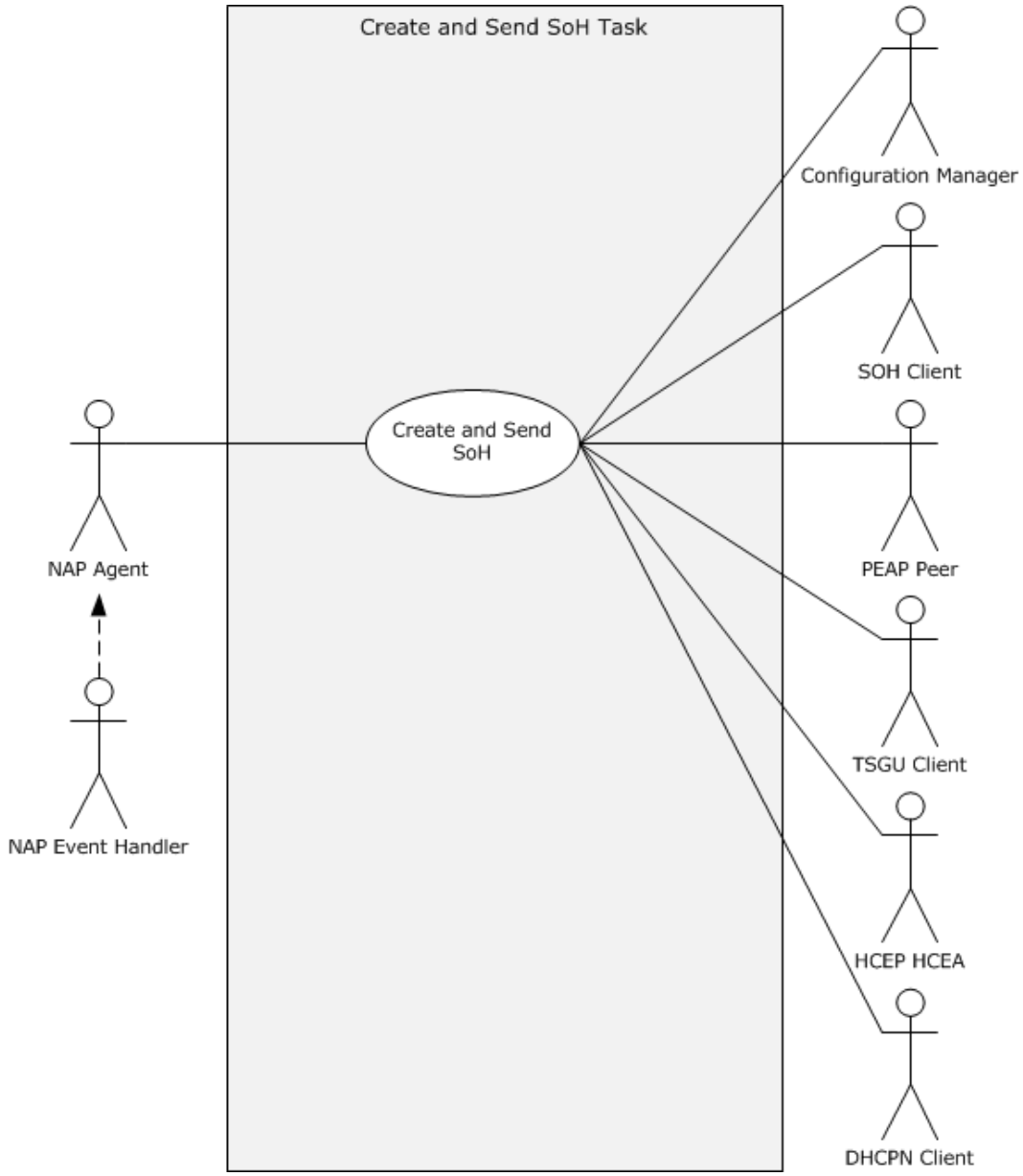

Figure 18: Create and Send SoH Task use case diagram

6.1.3.4 Use Case: Create and Send SoH -- NAP Agent

This use case is associated with the use case diagram in section 6.1.3.3.

Goal: To create an SoH message [TNC-IF-TNCCSPBSoH] containing health information about the client computer and send the SoH message to the PEP computer by using one of the transport protocols: [MS-HCEP], [MS-DHCPN], [MS-TSGU], or [MS-PEAP].

Context of Use: This use case is initiated when there is a trigger for new health information on a client computer where the NAP components are deployed.

Direct Actor: This role is performed by the **NAP Agent**. For more information, see section 3.3.1. The **NAP Agent** is the main software component on the NAP client computer. It is responsible for executing NAP-related operations, such as fetching the NAP configuration, creating the correlation ID, determining which transport protocol to use, and so on. Its interest in this task is that all transport protocol messages are sent by the task with a correlation ID, an SoH packet, and the authentication data.

Primary Actor: This role is performed by the **NAP Event Handler**. The **NAP Event Handler** is a component on the NAP client computer that receives events from the underlying protocol layers and passes them to the **NAP Agent**. Its interest in this task is that the use case will always process the received events.

Supporting Actors: The supporting actors are as specified in section 6.1.3.2.

Stakeholders and Interests:

- **Proxy SoH Task**: The purpose of this stakeholder is to proxy the correlation ID, the SoH packet, and the authentication data from the DHCPN, HCEP, TSGU, or PEAP servers to the RNAP client. As such, the primary interest of this stakeholder is to ensure that the Create and Send SoH Task only sends protocol messages that contain all three components.

Precondition: The NAP client components on the client computer are deployed and configured correctly by the client administrator.

Minimal Guarantees:

- The **NAP Event Handler** will always receive the events.
- The use case will always process the received events.
- No transport protocol messages are sent by the task without a correlation ID, an SoH packet, and the authentication data.

Success Guarantee: The health status on the client computer is correctly assessed and an SoH message is created based on the health assessment and sent to the NAP health policy server.

Trigger: The Create and Send SoH Task can be triggered by any of the following:

- The **NAP Event Handler** receives an event indicating that the network status of the client computer has changed.
- The **NAP Event Handler** receives an event indicating that the DHCP lease expires and has to be renewed.
- The **NAP Event Handler** receives an event indicating that the IPSec certificate has expired.
- The **NAP Event Handler** receives an event indicating that the client computer reboots or awakens from hibernation.
- The **NAP Event Handler** receives an event indicating that the client computer attempts to access a NAP-protected network resource by using IPSec, VPN, or RDP.
- The **NAP Event Handler** receives an event indicating that the client computer completes remediation.

- The **NAP Event Handler** receives an event indicating that the SoH values change due to an internal or external event. (A state change occurs in the client computer firewall, Windows Server Update Service, antivirus or antispyware software, and so on.)
- The **NAP Event Handler** receives an event indicating that there is a change in the NAP configuration of the client computer, for example a NAP Group Policy change.

Main Success Scenario:

1. The **NAP Event Handler** triggers the use case when one of the following events occurs:
   - The **NAP Event Handler** receives an event indicating that the network status of the client computer has changed.
   - The **NAP Event Handler** receives an event indicating that current DHCP lease expires.
   - The **NAP Event Handler** receives an event indicating that the IPSec certificate received via [MS-HCEP] expires.
   - The **NAP Event Handler** receives an event indicating that the client computer reboots or awakens from hibernation.
   - The **NAP Event Handler** receives an event indicating that the client computer accesses a NAP-protected network resource by using IPSec, VPN, or RDP.
   - The **NAP Event Handler** receives an event indicating that the client computer completes remediation.
   - The **NAP Event Handler** receives an event indicating that a state change occurs on the client computer modifying the current SoH values.
   - The **NAP Event Handler** receives an event indicating that the NAP configuration is modified.

   The **NAP Event Handler** passes the event to the **NAP Agent**.
2. The **NAP Agent** creates a unique correlation ID.
3. The **NAP Agent** requests and receives the current NAP configuration from the **Configuration Manager**.
4. The **NAP Agent** obtains an SoH packet in the following manner:
   - The **NAP Agent** calls the GetSoHRequest abstract interface ([TNC-IF-TNCCSPBSoH]), passing the correlation ID as an input. The Task Timer and Backward Compatible settings from the NAP configuration are also passed as inputs.
   - The **SoH Client** creates an SoH Packet with valid health information using the process described in [TNC-IF-TNCCSPBSoH].
   - The **SoH Client** returns the SoH Packet to the **NAP Agent** using the *SoHRequest* output parameter of the GetSoHRequest abstract interface.
5. The **NAP Agent** determines which transport protocol processing is required, according to the **Enabled EC List** and the trigger type.
6. If the **NAP Agent** determines the trigger is applicable to DHCPN, DHCPN is enabled and the client computer has new or existing DHCP connections or is configured to use DHCP:

- The **NAP Agent** notifies the **DHCP Client** about the health change using the abstract interface DhcpClientNotifySoHChange (see [MS-DHCPN] section 3.1.7.3).
- The **DHCP Client** composes a DHCPN message containing the SoH packet which it retrieves using NAP EC APIs (see [MSDN-NAPAPI]) and sends it to the PEP using the process described in [MS-DHCPN] section 3.1.4.2.

7. If the **NAP Agent** determines the trigger is applicable to HCEP, HCEP is enabled and the client computer has new or existing IPSec connections:
   - The **NAP Agent** sends the correlation ID, the SoH packet, and authentication data to the **HCEP HCEA** using the [MS-HCEP] client abstract interface. The Cryptographic Service Provider, Cryptographic Provider Type, Public Key OID, Public Key Length, Public Key Spec, Hash Algorithm OID, HRA Auto-Discovery, Use SSL, HRA URLs, and Reconnect Attempts settings from the NAP configuration are also passed as inputs.
   - The **HCEP HCEA** composes an HCEP message containing the SoH packet and sends it to the PEP using the process described in [MS-HCEP] section 3.1.5.1.
8. If the **NAP Agent** determines the trigger is applicable to TSGU, TSGU is enabled, and the client computer has new or existing RDP connections:
   - The **NAP Agent** sends the correlation ID, the SoH packet, and authentication data to the **TSGU Client** using the [MS-TSGU] client abstract interface.
   - The **TSGU Client** composes a TSGU message containing the SoH packet and sends it to the PEP using the process described in [MS-TSGU] section 3.6.4.
9. If the **NAP Agent** determines the trigger is applicable to PEAP, PEAP is enabled, and the client computer has new or existing VPN connections or is configured to use EAPOL:
   - The **NAP Agent** sends the correlation ID, the SoH packet, and authentication data to the **PEAP Peer** using the [MS-PEAP] client abstract interface.
   - The **PEAP Peer** composes a PEAP message containing the SoH packet and sends it to the PEP using the process described in [MS-PEAP] section 3.2.5.4.5.

Extensions: None.

6.2 Task Context

This section describes the relationship between this task and its environment.

6.2.1 Task Environment

This task is accomplished by the NAP client in an environment where client computer request access to network resources under the control of devices or servers acting as PEPs [RFC2753].

To accomplish this task, the NAP client requires the following from its environment:

- **Requirement:** The SOH Client, NAP Agent, DHCP Client, TSGU Client, PEAP Peer, and HCEP ECEA are correctly configured as specified in section 4.1.1.
  - **Reason for requirement:** Correct configuration is required for the SOH Client to collect health information from SHAs, compose the SoH message and send it to NAP Agent, for the NAP Agent to send the SoH message to the DHCP Client, TSGU Client, PEAP Peer, and HCEP

ECEA, and for the DHCP Client, TSGU Client, PEAP Peer, and HCEP ECEA to send the message to the PEP server.

- **Satisfying the requirement:** The Update NAP Client Configuration Task completes successfully.
- **Verifying requirement is satisfied:**

  1. A network capture performed during the task execution shows SoH encapsulated within the messages of the transport protocol selected for this task, as specified in the Main Success Scenario. The request MUST contain an SoHReportEntry for each enabled SHA.

  2. No errors related to configuration are logged by the SOH Client, NAP Agent, DHCP Client, TSGU Client, PEAP Peer, or HCEP ECEA.

- **Consequences of not satisfying requirement:** The task is unable to create and send the SoH.

- **Requirement:** The task triggers described in section 6.1.3.4 are functioning correctly.
  - **Reason for requirement:** The triggers initiate the task.
  - **Satisfying the requirement:**

    1. The Update NAP Client Configuration Task executes successfully.

    2. The SHAs and NAP EC are enabled.

  - **Verifying requirement is satisfied:** A network capture performed immediately after any triggering event shows the SoH encapsulated within the messages of the transport protocol selected for this task, as specified in the main success scenario.
  - **Consequences of not satisfying requirement:** The task does not start; the SoH is not created and not sent.
- **Requirement:** The enabled SHAs are running and able to provide correct health evaluation of the client computer.
  - **Reason for requirement:** The SHAs provide health information which is encapsulated in the SoH.
  - **Satisfying the requirement:** The SHAs are enabled.
  - **Verifying requirement is satisfied:**

    1. A network capture performed during the task execution shows SoH encapsulated within messages of the transport protocol selected for this task, as specified in the main success scenario. The health information contained in the request must be complete and correct.

    2. No errors are logged by SHAs.

  - **Consequences of not satisfying requirement:** The task is unable to create the SoH, or the health information in the SoH is incorrect or missing.
- **Requirement:** There is network connectivity between the NAP client computer and PEP servers.
  - **Reason for requirement:** The NAP EC communicates with the PEP.

- **Satisfying the requirement:**

  1. The network interface of the client computer is configured to operate on the local subnet.

  2. The physical network path (network devices, Ethernet cables, and so on) between the local subnet and the PEP is connected.

  3. All network devices between the local subnet and the PEP are configured to allow packet flow between the two entities.

  4. The network infrastructure that provides name and address resolution and routing services is functional.

- **Verifying requirement is satisfied:** The NAP client computer can successfully ping the PEP over the network.

- **Consequences of not satisfying requirement:** The SoH cannot be sent.

- **Requirement:** The **HCEP HCEA** is running and can communicate with the HCEP HRA on the PEP by using the [MS-HCEP] protocol.

  - **Reason for requirement:** The **HCEP HCEA** is used to send SoH encapsulated within the [MS-HCEP] packets to the PEP.

  - **Satisfying the requirement:**

    1. The **HCEP HCEA** component is enabled, as described in [MS-GPNAP] section 2.3.

    2. The **HCEP HCEA** is configured with the HCEP HRA settings described in [MS-GPNAP] section 2.4.

  - **Verifying requirement is satisfied:**

    1. Monitoring tools show the **HCEP HCEA** is active.

    2. A network capture performed during health certificate enrollment shows SoH encapsulated within [MS-HCEP] messages. The message exchange must reflect the HRA settings.

    3. No errors are logged by the HCEP client.

  - **Consequences of not satisfying requirement:** The SoH cannot be sent using the Health Certificate Enrollment Protocol [MS-HCEP].

- **Requirement:** The **DHCPN Client** is running and can communicate with the DHCPN server on the PEP using the Dynamic Host Configuration Protocol (DHCP) Extensions for Network Access Protection (NAP) [MS-DHCPN].

  - **Reason for requirement:** The **DHCPN Client** is used to send SoH encapsulated within [MS-DHCPN] packets to the PEP.

  - **Satisfying the requirement:** The **DHCPN Client** component is enabled, as described in [MS-GPNAP] section 2.3.

  - **Verifying requirement is satisfied:**

    1. Monitoring tools show the **DHCPN Client** is active.

2. A network capture performed during the task execution shows SoH encapsulated within the [MS-DHCPN] messages.

3. No errors are logged by the **DHCPN Client**.

- **Consequences of not satisfying requirement:** The SoH cannot be sent using the Dynamic Host Configuration Protocol (DHCP) Extensions for Network Access Protection (NAP) [MS-DHCPN] protocol.

- **Requirement:** The **TSGU Client** is running and has the ability to communicate with the TSGU server on the PEP using the Terminal Services Gateway Server Protocol [MS-TSGU].

  - **Reason for requirement:** The **TSGU Client** is used to send SoH encapsulated within the [MS-TSGU] packets to the PEP.

  - **Satisfying the requirement:** The **TSGU Client** component is enabled, as described in [MS-GPNAP] section 2.3.

  - **Verifying requirement is satisfied:**

    1. Monitoring tools show the **TSGU Client** is active.

    2. A network capture performed during the task execution shows SoH encapsulated within the [MS-TSGU] messages.

    3. No errors are logged by the **TSGU Client**.

  - **Consequences of not satisfying requirement:** The SoH cannot be sent using the Terminal Services Gateway Server Protocol [MS-TSGU].

- **Requirement:** The **PEAP Peer** is running and can communicate with the PEAP server on the PEP using the Protected Extensible Authentication Protocol (PEAP) [MS-PEAP].

  - **Reason for requirement:** The **PEAP Peer** is used to send SoH encapsulated within [MS-PEAP] packets to the PEP.

  - **Satisfying the requirement:** The **PEAP Peer** component is enabled, as described in [MS-GPNAP] section 2.3.

  - **Verifying requirement is satisfied:**

    1. Monitoring tools show the **PEAP Peer** is active.

    2. A network capture performed during the task execution shows SoH encapsulated within the [MS-PEAP] messages.

    3. No errors are logged by the **PEAP Peer**.

  - **Consequences of not satisfying requirement:** The SoH cannot be sent using the Protected Extensible Authentication Protocol (PEAP) [MS-PEAP].

6.2.2 Task Relationships

6.2.2.1 Black-Box Relationship Diagrams

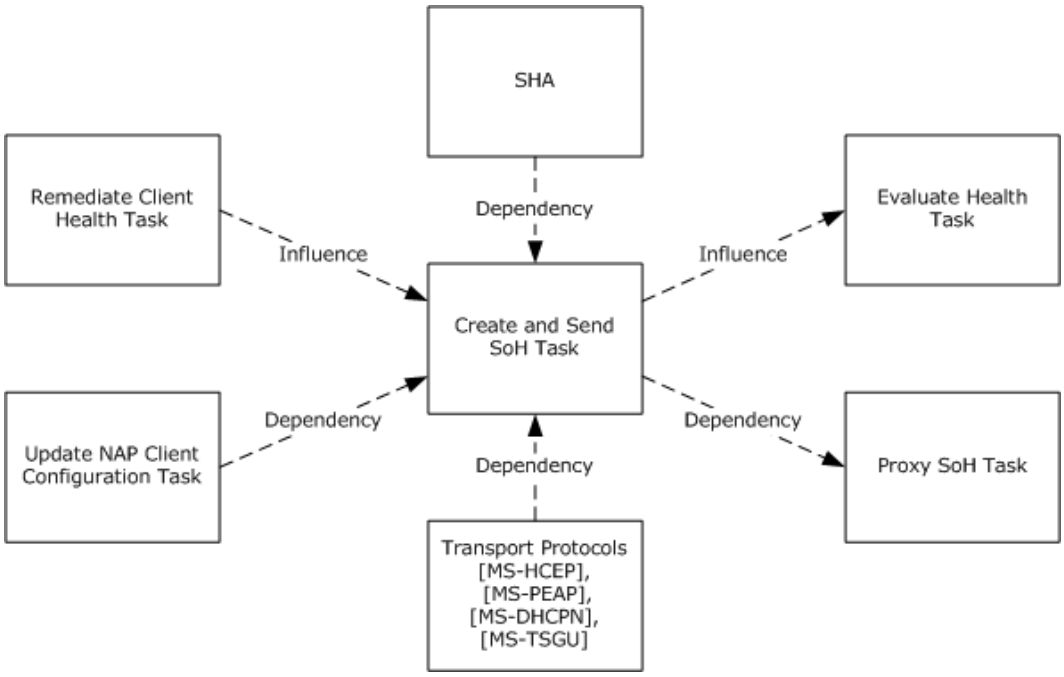

Figure 19: Create and Send SoH Task black-box relationships

The NAP client collects health information and an assessment, and creates SoH messages and sends them to the NAP health policy server via PEP. The SoH messages transported on the wire from the NAP client to the PEP are encapsulated by the underlying communication protocols: see [MS-HCEP], [MS-DHCPN], [MS-TSGU], and [MS-PEAP].

6.2.2.2 Task Dependencies

The dependencies in the relationship diagram are as follows:

- The Proxy SoH Task depends on the Create and Send SoH Task because the Proxy SoH Task relies on the Create and Send SoH Task to generate an SoH message so that the SoH message can be transferred to the NAP health policy server.
- This task depends on the SHA to perform the health state evaluation on the client computer in order for it to be able to create an SoH.
- This task depends on the Update NAP Client Configuration Task to set the Enabled EC List, and in the case of IPsec, it also depends on the Update NAP Client Configuration Task to provide the location of the HRA server.
- This task depends on the transport protocols listed in section 6.2.2.1 to transport the created SoH to the PEP.

6.2.2.3 Task Influences

The Create and Send SoH Task influences the Process SoH Task (section 9) as the latter is the consumer of the SoH generated by the Create and Send SoH Task.

The Create and Send SoH Task is influenced by the Remediate Client Health Task (section 15) as the latter can change the client computer firewall, Windows Server Update Service, or antivirus or antispyware software, and as a result, trigger the Create and Send SoH Task.

6.2.3 Task Assumptions and Preconditions

To accomplish this task, the NAP client has the following preconditions and assumptions:

- The operating system and hardware comprising on the client computer is trustworthy.
- The client administrators are trustworthy. The client administrators are responsible for enabling and configuring the NAP client correctly. They are also responsible for the integrity of executables that provide NAP client services.
- The underlying network infrastructures, such as the EC PEP channels, name and address resolution, and routing services, are configured correctly.
- The underlying task triggers, such as the EC connection to NAP protected network, the Configuration change task, and the networking modules are functioning correctly.
- The NAP client is enabled and correctly configured by the client administrator.

6.2.4 Task Versioning and Capability Negotiation

The Create and Send SoH Task does not define any versioning and capability negotiation beyond those described in the specifications of the protocols supported or used by the task, as listed in section 2.3.

6.3 Task Architecture

This section describes the structure of the Create and Send SoH Task and the interrelationships among its parts.

6.3.1 Task Architectural Constraints

There is only one instance of the Create and Send SoH Task on each client computer and this instance initializes itself each time it starts. Different instances of this task on different client computers can run independently.

6.3.2 Task Abstract Data Model

This section describes the state established, used, and maintained by processing rules of this task. State may be volatile or persisted and may pertain to one, some, or all instances of the task. The Task's state consists of the values of the named data elements (also called state variables) presented in this section. The overall organization of the data elements, with their names, is the Abstract Data Model. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Data Model, such association can be achieved in other ways. Implementations may depart from this model so long as their external behavior remains consistent with that described in this document.

Correlation Id: A structure that is used to pair SoH messages with SoHR messages, and which uniquely describes an SoH exchange. The data type and values are described in [MSDN-CorrelationId].

Create and Send SoH State: Indicates whether or not the task is active, where Active is represented by a value of 0x00000001 and Inactive by a value of 0x00000000.

NAP Available SHA List (Common): A list of available SHAs, as specified in section 4.1.1. When initialized, this ADM element will contain zero or more elements. If initialization fails, the list will contain zero elements.

6.3.3 Task Abstract Parameters

This section describes data passed to an instance of this task at the time it is invoked or triggered. The parameters consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Parameter. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Parameters, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

Enabled EC List: A data type and range of values as specified in section 5.3.2.

IPsec HRA List: A data type and range of values as specified in section 5.3.2.

6.3.4 Task Abstract Results

This section describes data returned by an instance of this task to its caller. The results consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Result. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Results, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

This task is expected to deliver an SoH message each time it is called. The sent SoH message follows the format defined in [TNC-IF-TNCCSPBSoH].

6.3.5 White-Box Relationships

The white-box relationships between the Create and Send SoH Task and other tasks are shown in the following figure.

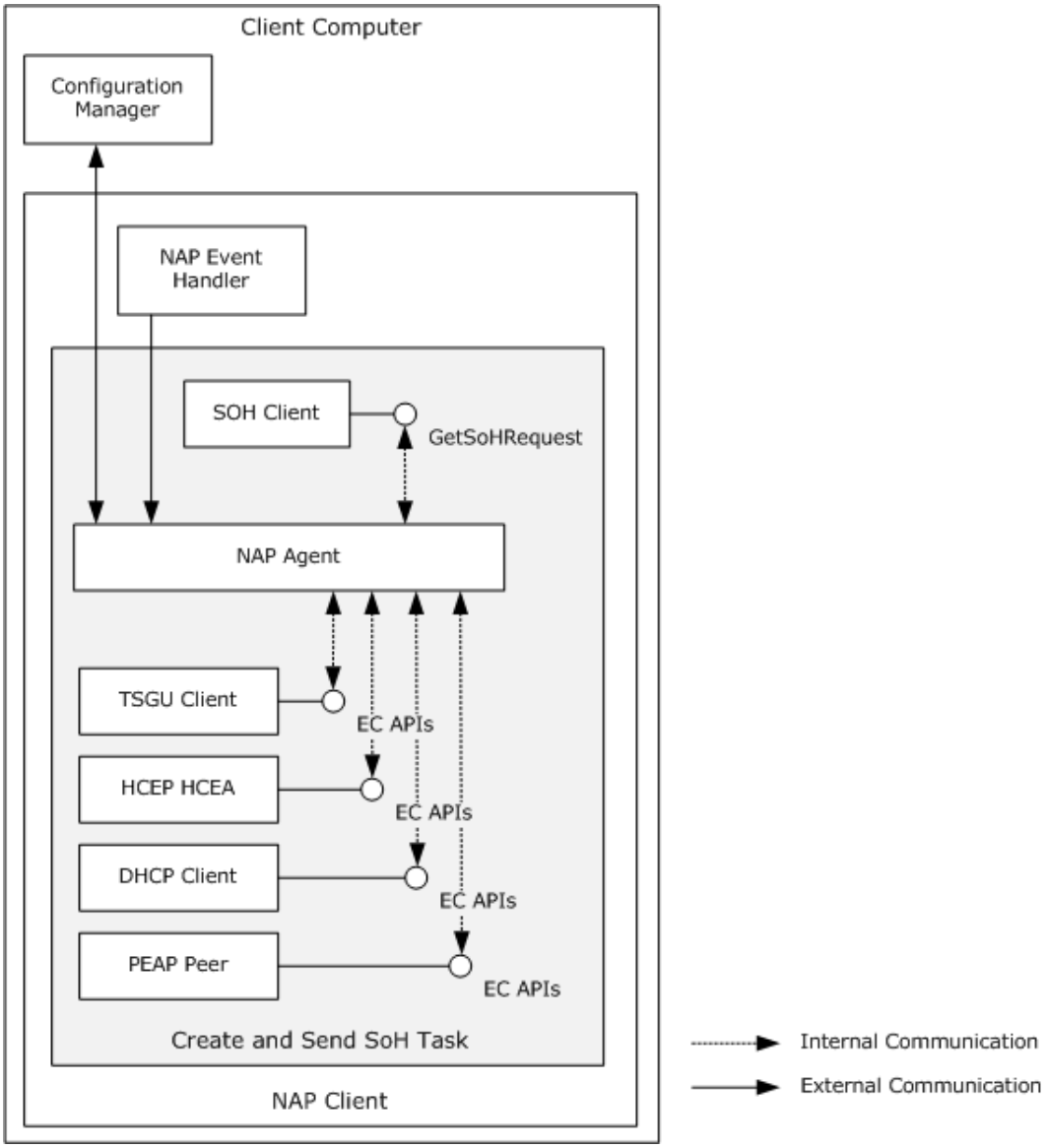

Figure 20: Create and Send SoH Task white-box relationships

The Create and Send SoH Task contains three major NAP client components: SoH Client (SHA), NAP agent, and NAP EC.

The diagram represents the relationships between the different components on the Client machine. The Create and Send SoH Task provides services related to the collection of health evaluations from the different SHA(s) and the packaging into SoH messages, as described in [TNC-IF-TNCCSPBSoH] and the encapsulating of the SoH messages by the EC into PEP-specific transport protocol messages as described in ([MS-HCEP], [MS-DHCPN], [MS-TSGU], and [MS-PEAP]). These encapsulated SoH messages are consumed by the Send SoH Task on a NAP policy enforcement server (PEP).

6.3.6 Task Events

6.3.6.1 Task Timers

There are no additional timers on outside entities imposed by this task other than the timers in the underlying transport system. However, inside this task there is a timer associated with all function calls that the SoH Client makes into SHAs. This timer, known as **ShaTimeoutInMsec** (see section 4.1.1), determines how soon these function calls must return.

6.3.6.2 Task Non-Timer Events

This task uses four non-timer events: operating system wake-up, health state change, connection state change, and Enabled EC List change. For more details, see section 6.4.3.2.

6.3.7 Task Architecture and Communication

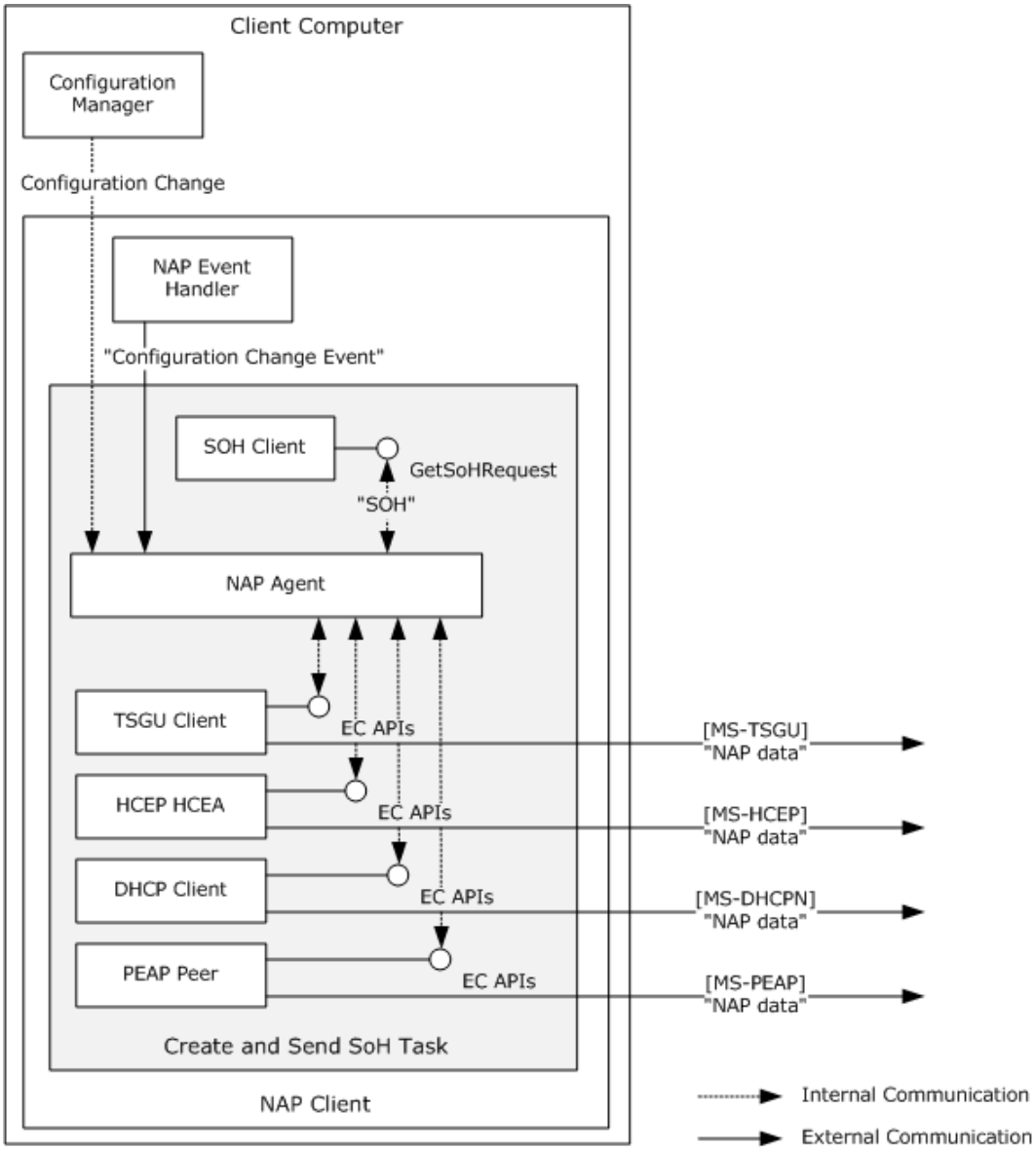

Figure 21: Create and Send SoH Task architecture and communication

6.3.8 Task Processing Rules

The following describes the operational flow of the Create and Send SoH Task:

1. The **NAP Event Handler** triggers the use case when one of the following events occurs:

   - The **NAP Event Handler** receives an event indicating that the network status of the client computer has changed.
   - The **NAP Event Handler** receives an event indicating that current DHCP lease expires.

- The **NAP Event Handler** receives an event indicating that the IPsec certificate received via [MS-HCEP] expires.
- The **NAP Event Handler** receives an event indicating that the client computer reboots or wakes up from hibernation.
- The **NAP Event Handler** receives an event indicating that the client computer accesses a NAP protected network resource using IPsec, VPN, or RDP.
- The **NAP Event Handler** receives an event indicating that the client computer completes remediation.
- The **NAP Event Handler** receives an event indicating that a state change occurs on the client computer that would modify the current SoH values.
- The **NAP Event Handler** receives an event indicating that the NAP configuration is modified.

The **NAP Event Handler** passes the event to the **NAP Agent**.

2. The **NAP Agent** creates a unique correlation Id as specified in [TNC-IF-TNCCSPBSoH].
3. The **NAP Agent** requests and receives the current NAP configuration from the **Configuration Manager**.
4. The **NAP Agent** obtains an SoH packet in the following manner:
    1. The **NAP Agent** calls the GetSoHRequest abstract interface in [TNC-IF-TNCCSPBSoH], passing the correlation Id as an input. The Task Timer and Backward Compatible settings from the NAP Client Configuration ADM element are also passed as inputs.
    2. The **SoH Client** creates an SoH Packet with valid health information using the process described in [TNC-IF-TNCCSPBSoH].
    3. The **SoH Client** returns the SoH Packet to the **NAP Agent** using the *SoHRequest* output parameter of the GetSoHRequest abstract interface.
5. The **NAP Agent** determines which transport protocol the trigger is applicable to, according to **Enabled EC List** and the type of the trigger, as follows:
    - If DHCP Enforcement is enabled and any triggering event except the following occurred, the **NAP Agent** notes that DHCPN processing is required.
        1. The IPsec certificate received via the Health Certificate Enrollment Protocol (HCEP) [MS-HCEP] expires.
        2. The client computer accesses a NAP protected network resource.
    - If Remote Access Enforcement is enabled and any triggering event except the following occurred, the **NAP Agent** notes that PEAP processing is required.
        1. The current DHCP lease expires.
        2. The IPsec certificate received via the Health Certificate Enrollment Protocol (HCEP) [MS-HCEP] expires.
        3. The client computer reboots or wakes up from hibernation.

- If IPsec Enforcement is enabled and any triggering event except the following occurred, the **NAP Agent** notes that HCEP processing is required.

  1. The current DHCP lease expires.

  2. The client computer reboots or wakes up from hibernation.

- If Wireless EAPOL Enforcement is enabled and any triggering event except the following occurred, the **NAP Agent** notes that PEAP processing is required.

  1. The current DHCP lease expires.

  2. The IPsec certificate received via the Health Certificate Enrollment Protocol (HCEP) [MS-HCEP] expires.

  3. The client computer accesses a NAP protected network resource.

- If RDG Enforcement is enabled and any triggering event except the following occurred, the **NAP Agent** notes that TSGU processing is required.

  1. The current DHCP lease expires.

  2. The IPsec certificate received via the Health Certificate Enrollment Protocol (HCEP) [MS-HCEP] expires.

  3. The client computer reboots or wakes up from hibernation.

- If EAP Enforcement is enabled and any triggering event except the following occurred, the **NAP Agent** notes that PEAP processing is required.

  1. The current DHCP lease expires.

  2. The IPsec certificate received via the Health Certificate Enrollment Protocol (HCEP) [MS-HCEP] expires.

  3. The client computer accesses a NAP protected network resource.

6. If the **NAP Agent** determines that the trigger is applicable to DHCPN and DHCP is enabled:

   - The **NAP Agent** notifies the **DHCP Client** about the health change using the abstract interface DhcpClientNotifySoHChange (see [MS-DHCPN] section 3.1.7.3).

   - The **DHCP Client** composes a DHCPN message containing the SoH packet which it retrieves using NAP EC APIs (see [MSDN-NAPAPI]) and sends it to the PEP using the process described in [MS-DHCPN] section 3.1.4.2.

7. If the **NAP Agent** determines that the trigger is applicable to HCEP, HCEP is enabled, and the client computer has new or existing IPsec connections:

   - The **NAP Agent** sends the correlation Id, the SoH packet and authentication data to the **HCEP HCEA** using the Health Certificate Enrollment Protocol (HCEP) [MS-HCEP] client abstract interface. The Cryptographic Service Provider, Cryptographic Provider Type, Public Key OID, Public Key Length, Public Key Spec, Hash Algorithm OID, HRA Auto-Discovery, Use SSL, HRA URLs, and Reconnect Attempts settings from the NAP configuration are also passed as inputs.

   - The **HCEP HCEA** composes an HCEP message containing the SoH packet and sends it to the PEP using the process described in [MS-HCEP] section 3.1.5.1.

8. If the **NAP Agent** determines that the trigger is applicable to TSGU, TSGU is enabled, and the client computer has new or existing RDP connections:

   - The **NAP Agent** sends the correlation Id, the SoH packet, and authentication data to the **TSGU Client** using the Terminal Services Gateway Server Protocol [MS-TSGU] client abstract interface.

   - The **TSGU Client** composes a TSGU message containing the SoH packet and sends it to the PEP using the process described in [MS-TSGU] section 3.6.4.

9. If the **NAP Agent** determines that the trigger is applicable to PEAP, PEAP is enabled, and the client computer has new or existing VPN connections or is configured to use EAPOL:

   - The **NAP Agent** sends the correlation Id, the SoH packet, and authentication data to the **PEAP Peer** using the Protected Extensible Authentication Protocol (PEAP) [MS-PEAP] client abstract interface.

   - The **PEAP Peer** composes a PEAP message containing the SoH packet and sends it to the PEP using the process described in [MS-PEAP] section 3.2.5.4.5.

10. If the connection to PEP fails, the SoH message is not transported.

6.3.9 Task Failure Scenarios

6.3.9.1 Failures in SHA and SoH Client Communication with SHA

These failures are caused by an error with the initialization, registration, or binding of the SHA. The SoH Client relies on its ability to communicate with the registered SHAs in order to retrieve the health status that is monitored and reported by a SHA. In this failure scenario either the health information collection fails on the SHA or SHA fails to communicate the health status of the properties that are monitored by the SHA to SoH Client. The SoH Client experiencing this failure will not be able to create SoH messages and EC will not send a SoH to PEP, which may make the client machine **unhealthy**. The failures are detected by a timer monitored by a SoH Client (section 6.3.6.1). The NAP System provides an error code enabling the administrator to configure fragility settings to detect and override the health policy decision on the policy decision point (PDP).

6.3.9.2 NAP Agent Communication with EC

These failures are caused by an error with the initialization or registration of the enforcement client. In this task, the NAP client relies on the communication between the NAP agent service and an enforcement client to get the task network change triggers specified in section 6.1.3.4. A client experiencing this failure will not be able to listen to the network change triggers and make the NAP agent miss requests to create SoH messages. These failures are not detected by the NAP agent. The NAP System cannot recover from such a failure.

6.3.9.3 EC and PEP Communication

These failures can be caused by:

- Misconfigurations on the EC and/or PEP.

- Network connectivity issues in which the EC cannot communicate with the PEP.

If the EC cannot communicate with the PEP, the client machine may not have access to the network resources. The system may recover from certain types of failures (for example, the DHCP EC can attempt to connect to secondary DHCP server if there is no response from the primary server) and

cannot recover from various other failures (for example, if the EC cannot communicate with an 802.1X switch or VPN server then the NAP System cannot recover from this failure). The failures can be detected by the timers on the ECs.

6.4 Task Details

This section contains the details that complete the descriptions in earlier sections of the document. These are needed to understand and implement this task.

6.4.1 Task Precondition Details

For the complete list of task preconditions and assumptions, see section 6.2.3. Details for some of the preconditions are as follows:

- The NAP agent service is started and initialized correctly on the client computer.
- SHAs are correctly registered and bound to the SoH Client so that the SoH Client has a complete SHA list.
- ECs are correctly configured, enabled, and bound to the NAP agent so that the NAP agent has a complete enabled EC list.
- Depending on the specific configuration, any of the required PEP channels (the **HTTP/S channel**, the **PEAP channel**, or the **DHCP channel**) are functioning correctly.
- The networking modules (for example, TCP/IP modules) are functioning correctly so that notifications can be sent to the NAP agent in time when there are network status changes that the NAP agent is interested in.

6.4.2 Task Initialization of External Entities

None.

6.4.3 Task Event Details

6.4.3.1 Task Timer Details

Inside this task, there is a timer associated with all function calls that the SoH Client makes into SHAs. When the SoH Client calls into a SHA to perform a task, such as getting a new health statement from the SHA, a timeout is enforced. The SHA is expected to complete the call within the timeout. Otherwise, the call is canceled and an error is reported by the SoH Client. The timeout value from the **ShaTimeoutInMsec** ADM element described in section 4.1.1.

This task does not impose any additional timers besides the timers related to the underlying transports; they are described in [MS-DHCPN], [MS-PEAP], and [MS-HCEP].

6.4.3.2 Task Non-Timer Event Details

This task uses and responds to the following non-timer events:

- Operating system wake-up from sleep or hibernation: When a client computer wakes up from sleep or hibernation, the NAP Event Handler on the client computer receives such events and triggers the **NAP Agent** that starts a new SoH transaction by calling SoH Client for new health statements.

- Health state changes on the client computer: Depending on the health state that a SHA is monitoring, if changes occur in this health state (for example, if an installed SHA is monitoring the Windows Server Update Services status, and if the Windows Server Update Services is turned off), the SHA notifies the NAP Event Handler, which triggers the **NAP Agent** that starts a new SoH transaction.
- Connection state changes: An EC component may monitor the state of a connection that it manages. When it decides that the connection state has changed and the health of the client computer needs to be re-evaluated, the EC component calls into the NAP agent directly to ask for new health statements, which triggers a new SoH transaction.
- Configuration changes: The Update client configuration task completion triggers this task. When this is triggered by the Update client configuration task, it calls into all registered SHAs to get new health statements, triggering this task.

6.4.4 Task Architectural Details

This section illustrates an example of a NAP client creating an SoH and sending it to the PEP. The **NAP Agent** requests that the SoH Client perform a health evaluation and create an SoH by calling the **INapSystemHealthAgentRequest** API. After the SoH is created, the **NAP Agent** passes the health information to the EC by calling the **INapEnforcementClientConnection** API. A complete list of SOH Client and EC APIs are specified in [MSDN-NAPAPI].

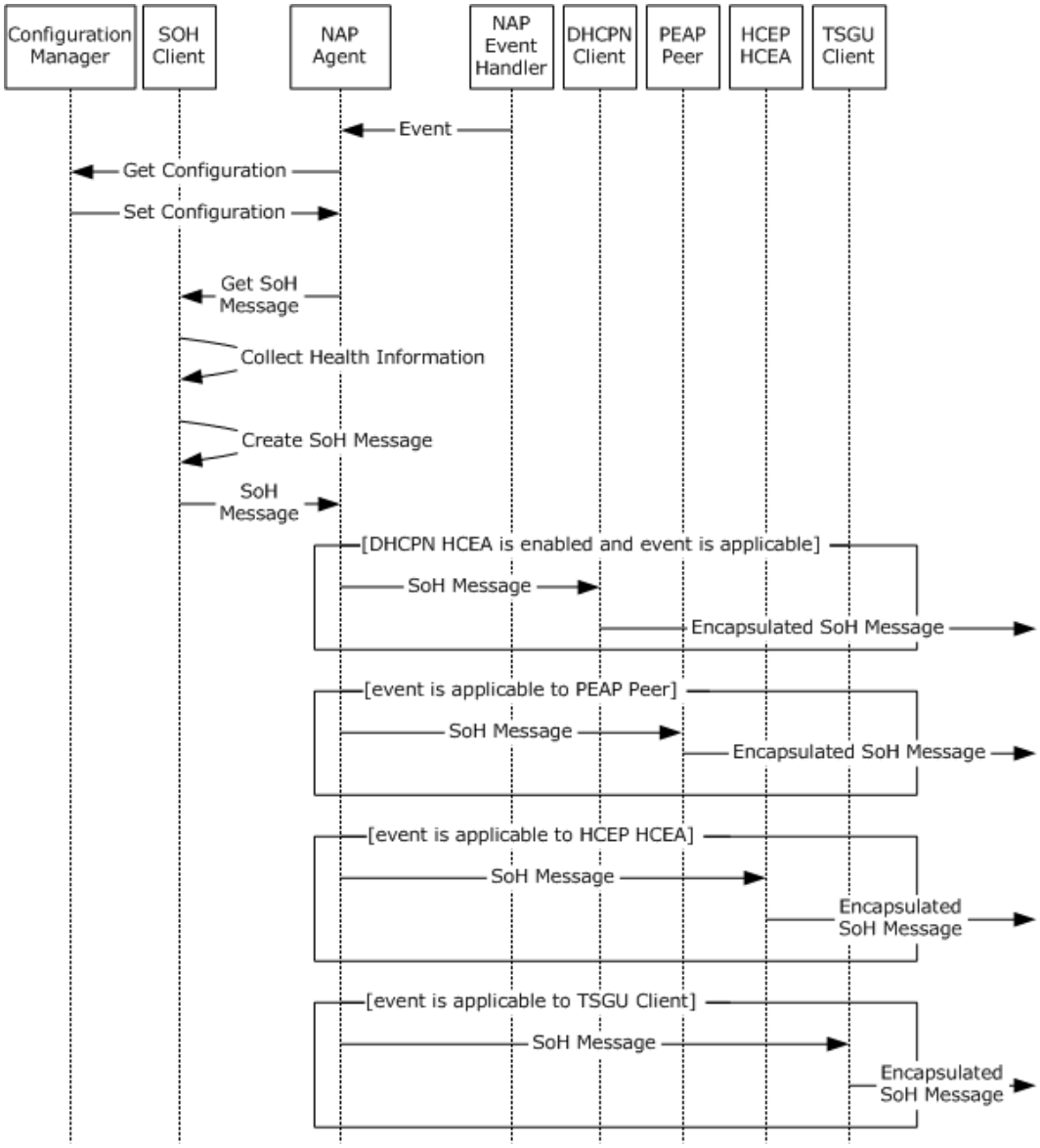

Figure 22: Sequence diagram for the main success scenario of the Create and Send SoH Task

1. The **NAP Event Handler** receives an event that requires the creation of an SOH, and notifies the **NAP Agent** of this event.

2. The **NAP Agent** retrieves the current configuration from the **Configuration Manager**.

3. The **NAP Agent** requests an SoH from the SoH Client. The SoH Client collects the health information (such as the status of anti-virus software or the status of Windows Server Update Services) from the SHAs in the **NAP Available SHAs List** ADM element (section 4.1.1).

4. The SoH Client composes an SoH message with the health information collected from the SHAs [TNC-IF-TNCCSPBSoH] and passes it back to the **NAP Agent**.

5. The **NAP Agent** forwards the SoH to those ECs in the **Enabled EC List** that are applicable to the event received by the **NAP Event Handler**.

6. The **DHCP Client**, **TSGU Client**, **PEAP Peer**, and **HCEP ECEA**, when applicable to the event, encapsulate the SoH in the transport protocol.

7. The **DHCP Client**, **TSGU Client**, **PEAP Peer**, and **HCEP ECEA**, when applicable to the event, send the SoH message to the PEP.

6.4.5 Task Processing Rule Details

The following describes the operational flow details of the Create and Send SoH Task:

1. The Create and Send SoH Task is triggered as specified in the processing rules defined in section 6.3.8. When the NAP agent is triggered, the value of the **Create and Send SoH State** ADM element changes to active as specified in section 6.3.2.

2. The NAP agent creates a unique **correlation Id** as specified in [TNC-IF-TNCCSPBSoH].

3. The NAP agent requests and receives the current **NAP** configuration from the Configuration Manager.

4. The NAP agent obtains an **SoH Packet** in the following manner:

   - The NAP agent calls the **GetSoHRequest** abstract interface in [TNC-IF-TNCCSPBSoH], passing the **correlation Id** as an input. The *ShaTimeoutInMsec* ADM Element and the Backward Compatible field from the **NAP Client Config** ADM element are also passed as inputs.

   - The **SoH Client** performs an **API call** (see section 3.3.1 and [MSDN-NAPAPI]) to each **SHA** listed in **NAP Available SHAs List**, requesting it to perform health evaluation of its monitored resources. After the API is called, the NAP agent waits for the evaluation to complete. If the evaluation does not complete within the time defined in the *ShaTimeoutInMsec* parameter, the task fails and the **SoH** will not be delivered. The task can also fail if the **SOH Client** is unable to connect to the SHA using the API. For information about possible failure scenarios, see section 6.3.9.1.

   - The **SoH Client** creates an **SoH Packet** with valid health information using the process described in [TNC-IF-TNCCSPBSoH].

   - The **SoH Client** returns the **SoH Packet** to the NAP agent using the *SoHRequest* output parameter of the **GetSoHRequest** abstract interface.

5. The NAP agent determines which transport protocol the trigger is applicable to, based on the value of the **Enabled EC** List ADM element (section 5.3.2) that stores the value of Enforcement Client settings specified in [MS-GPNAP] section 2.3 and the type of the trigger. For details about the possible trigger types and the determination of the applicable transport protocol, see section 6.3.8.

6. If the NAP agent determines that the trigger is applicable to DHCPN, DHCPN is enabled, and the client computer is configured to use DHCP:

   - The NAP agent notifies the DHCP Client about the health change using the abstract interface DhcpClientNotifySoHChange (see [MS-DHCPN] section 3.1.7.3).

   - The DHCP Client composes a DHCPN message containing the **SoH Packet** which it retrieves using NAP EC APIs (see [MSDN-NAPAPI]) and sends it to the PEP using the process described

in [MS-DHCPN] section 3.1.4.2, Creating and Transmitting a DHCPREQUEST Message During Lease Renewal.

7. If the NAP agent determines that the trigger is applicable to HCEP, HCEP is enabled, and the client computer has new or existing **IPsec** connections:

   - The NAP agent sends the **correlation Id**, the **SoH Packet**, and authentication data to the HCEP HCEA using the [MS-HCEP] client abstract interface. The Cryptographic Service Provider (CSP), Cryptographic Provider Type, Public Key OID, Public Key Length, Public Key Spec, Hash Algorithm OID, HRA Auto-Discovery, Use SSL, HRA URLs, and Reconnect Attempts values stored in the **NAP Client Config** ADM element (section 4.1.1) are also passed as inputs.

   - The HCEP HCEA uses PKCS #10 certificate settings sent as parameters for creating a PKCS #10 certificate request, as specified in [MS-HCEP] section 2.2.1.4. Then the HCEP HCEA composes an HCEP message containing the **SoH Packet** and sends it to the PEP using the process described in [MS-HCEP] section 3.1.5.1 (Sending an HCEP Request).

   - If the connection to the PEP fails, the Reconnect Attempts parameter is used to decide how long the HCEP HCEA should wait before attempting to reconnect to the PEP. If the connection cannot be set, the SoH message is not transported.

8. If the NAP agent determines that the trigger is applicable to TSGU, TSGU is enabled, and the client computer has new or existing RDP connections:

   - The NAP agent sends the **correlation Id**, the **SoH Packet**, and authentication data to the TSGU Client using the [MS-TSGU] client abstract interface.

   - The TSGU Client composes a TSGU message containing the **SoH Packet** and sends it to the PEP using the process described in [MS-TSGU] section 3.6.4 (Message Processing Events and Sequencing Rules).

9. If the NAP agent determines that the trigger is applicable to PEAP, PEAP is enabled, and the client computer has new or existing VPN connections or is configured to use EAPOL:

   - The NAP agent sends the **correlation Id**, the **SoH Packet**, and authentication data to the PEAP PEER using the [MS-PEAP] client abstract interface.

   - The PEAP PEER composes a PEAP message containing the **SoH Packet** and sends it to the PEP using the process described in [MS-PEAP] section 3.2.5.4.5 (Received SoH Request TLV).

10. If the connection to the PEP fails, the SoH message is not transported.

6.5 Task Security

The only security consideration for this task is in the case of HCEP EC Enabled and the client computer requires that the X.509 certificate use SSL as specified in [MS-TLSP]. For additional information about security considerations, see section 16, as well as the Security sections of the referenced protocol Technical Documents.

7 Proxy SoH Task

This section describes the task of sending SoH messages from the NAP Enforcement Proxy (PEP) to the NAP health policy server. The SoH messages are transported on PEP channels (transport protocols). The SoH messages can arrive on a number of different transport protocols. The SoH message is received from the incoming transport protocol and passed to the NAP Enforcement Proxy. The SoH message is then passed by the NAP Enforcement Proxy to the health policy server via Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] or EAP-supporting RADIUS [RFC3579]. The format of the SoH message is specified in [TNC-IF-TNCCSPBSoH].

Note This task uses the **PEP Channel Used** ADM element (section 4.1.1). All other common information defined in section 4 is not applicable to this task.

7.1 Task Overview

7.1.1 Task Purpose

The purpose of this task is to ensure that an SoH message is correctly sent from the NAP Enforcement Proxy (PEP) to the NAP health policy server after the SoH has been constructed and sent to the NAP Enforcement Proxy in the Create and Send SoH Task (section 6).

7.1.2 Task Applicability

This task is used when an SoH message is received by the NAP Enforcement Proxy. This task is not applicable if a NAP System is not deployed.

7.1.3 Task Use Cases

7.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Proxy SoH Task are as follows:

HCEP HRA: This protocol server is used to receive HCEP Protocol [MS-HCEP] messages from an HCEP client on the EC computer. It acts in the role of an enforcement server (NAP ES) in this use case when HCEP is used. The interest of this actor in this task is that the use case will always process the received HCEP messages.

DHCPN Server: This protocol server is used to receive (DHCP) Extensions for NAP [MS-DHCPN] messages from a DHCPN client on the EC computer. It acts in the role of an enforcement server (NAP ES) in this use case when DHCPN is used. The interest of this actor in this task is that the use case will always process the received DHCPN messages.

TSGU Server: This protocol server is used to receive TSGU Protocol [MS-TSGU] messages from a TSGU client on the EC computer. It acts in the role of an enforcement server (NAP ES) in this use case when TSGU is used. The interest of this actor in this task is that the use case will always process the received TSGU messages.

PEAP Pass-through Server: This protocol server is used to receive PEAP Protocol [MS-PEAP] messages from a PEAP peer on the EC computer. It acts in the role of an enforcement server (NAP ES) in this use case when PEAP is used. The interest of this actor in this task is that the use case will always process the received PEAP message.

NAP Enforcement Proxy: The NAP Enforcement Proxy is used to proxy SoH messages to the NPS. The interest of this actor in this task is that it will attempt to send any SoH that it receives via Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] or EAP-supporting RADIUS [RFC3579].

Receive SoH Task: The purpose of this task is to ensure the reception of SoH messages forwarded by a NAP Enforcement Proxy computer. The main interest of the task in this use case is to receive SoH messages only via Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] or EAP-supporting RADIUS [RFC3579].

7.1.3.2 Supporting Actors and Task Interests Summary

RNAP client: A client that uses Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] to transport the SoH/SoHR messages between the NAP Enforcement Proxy and the NAP health policy server. The Proxy SoH Task uses the RNAP client to send the SoH from the PEP to the NAP health policy server when HCEP, DHCPN, or TSGU enforcement is used.

EAP-supporting RADIUS client: A client that uses RADIUS [RFC2865] and RADIUS support for EAP [RFC3579] to transport PEAP messages that contain the SoH/SoHR messages between the NAP ES and the NAP health policy server. The Proxy SoH Task uses the EAP-supporting RADIUS client to send the SoH from the NAP Enforcement Proxy to the NAP health policy server when PEAP enforcement is used.

7.1.3.3 Use Case Diagrams

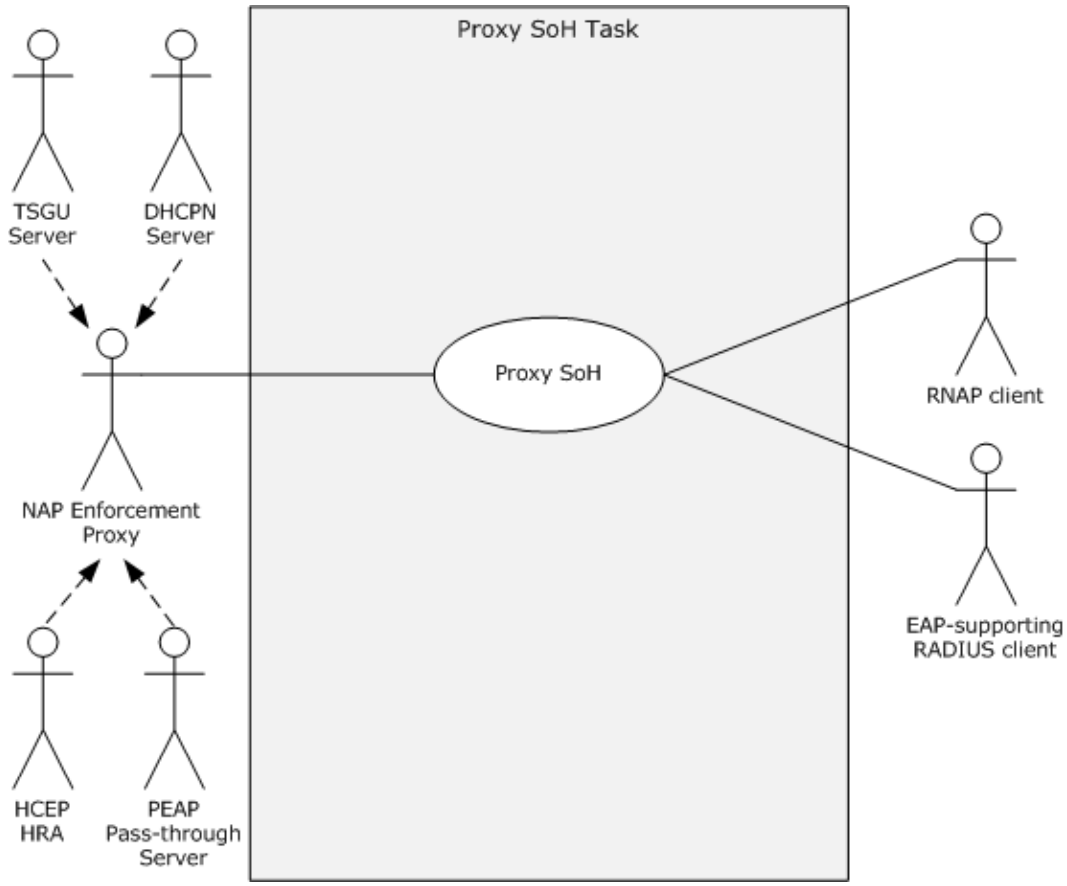

Figure 23: Proxy SoH Task use case diagram

7.1.3.4 Use Case: Proxy SoH - NAP Enforcement Proxy

Goal: To proxy the SoH message [TNC-IF-TNCCSPBSoH] received by the NAP enforcement server (NAP ES) (HCEP HRA, DHCPN server, TSGU server, or PEAP Pass-through server) to the NAP health policy server.

Context of Use: This use case is used when an SoH message is passed by the NAP ES to the NAP Enforcement Proxy.

Direct Actor: This role is performed by the NAP Enforcement Proxy. It is used to proxy SoH messages to the NPS. The interest of this actor in this task is in the ability to integrally proxy SoH messages via RADIUS-based channels, either Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] or EAP-supporting RADIUS [RFC3579].

Primary Actor: This role is performed by the NAP ES. The interest of the primary actor in this task is that the use case will always process SoH messages received by the NAP ES.

Supporting Actors: The supporting actors are the **RNAP** channel and EAP-supporting RADIUS.

Stakeholders and Interests: The stakeholders are defined as follows:

Receive SoH Task: The purpose of this task is to receive SoH messages. The main interest of the task in this use case is to receive SoH messages only via Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] or EAP-supporting RADIUS [RFC3579].

Precondition: The Create and Send SoH Task was completed successfully, any NAP ES received payload containing an SoH message from the NAP client, and the message was passed to the NAP Enforcement Proxy.

Minimal Guarantees:

- The use case will always process the SoH messages received from the NAP ES.
- The use case will always attempt to send the SoH messages that it receives via Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] or EAP-supporting RADIUS [RFC3579].

Success Guarantee: All the Minimal Guarantees are satisfied. Additionally, the payload that contains an SoH message is sent successfully by Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] or EAP-supporting RADIUS [RFC3579].

Trigger: Any NAP ES triggers this task when a message from the corresponding supported protocol arrives.

Main Success Scenario:

1. The NAP Enforcement Proxy receives an encapsulated SoH via one of the following transport protocols: [MS-HCEP], [MS-DHCPN], [MS-TSGU], or [MS-PEAP].
2. If PEAP enforcement is used:
    1. The NAP Enforcement Proxy passes the entire incoming message to the EAP-supporting RADIUS client.
    2. The EAP-supporting RADIUS client sends the EAP-encapsulated SoH message to the NAP policy health server.
3. If HCEP, DHCPN, or TSGU enforcement is used:

1. The payload that contains the SoH is passed to the RNAP client.

2. The RNAP client sends the SoH message encapsulated using Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] to the NAP policy health server.

4. The RNAP client or EAP-supporting RADIUS client successfully sends a Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] or EAP-supporting RADIUS [RFC3579] message containing the SoH.

Extensions: None.

7.2 Task Context

This section describes the relationship between this task and its environment.

7.2.1 Task Environment

This task is accomplished by the NAP ES in an environment where an encapsulated SoH is received over a PEP channel from a NAP EC and forwarded to a NAP health policy server over a RADIUS channel. The environment should meet the following requirement to support this task.

- **Requirement:** All NAP enforcement servers (HCEP HRA, DHCP server, TSGU server, PEAP Pass-through server) are running and have the ability to receive messages in the corresponding supported protocol ([MS-HCEP], [MS-DHCPN], [MS-TSGU], and [MS-PEAP] respectively) from their counterpart client's NAP EC.

  - **Reason for requirement:** The NAP enforcement servers are used to receive SoH messages from the NAP EC, where the messages are encapsulated within a message of the protocol which the NAP ES supports.

  - **Satisfying the requirement:**

    1. The NAP enforcement servers have network access from the counterpart client at the NAP EC:

       - The network interface of the server computer is configured to operate on the local subnet.

       - The physical network path (network devices, Ethernet cables, and so on) between the local subnet and the NAP EC is connected.

       - All network devices between the local subnet and the NAP EC are configured to allow packet flow between the two entities.

       - The routing tables in the client computer are configured to enable correct packet routing between the client computer and the NAP Enforcement Proxy.

    2. The NAP ES services have been started.

  - **Verifying requirement is satisfied:**

    1. The client computer can successfully ping the NAP ES computer over the network.

    2. A sniffer trace performed during the SoH validation event shows packets of either of the supported protocols traveling between the NAP EC computer and the NAP ES computer.

    3. The NAP ES server is shown as running within the list of services.

4. No errors are logged by the NAP ES.

- **Consequences of not satisfying requirement:** The task is unable to receive encapsulated SoH messages and proxy them to the NPS.

Unless explicitly specified otherwise, the task implementation may assume its environment is properly configured and is not expected to verify that the requirement described in this section is satisfied. On the other hand, the task implementation should be able to gracefully handle errors which may be caused by environment misconfiguration or temporary dysfunction. The implementation should log these errors along with relevant information to allow troubleshooting.

7.2.2 Task Relationships

7.2.2.1 Black-Box Relationship Diagrams

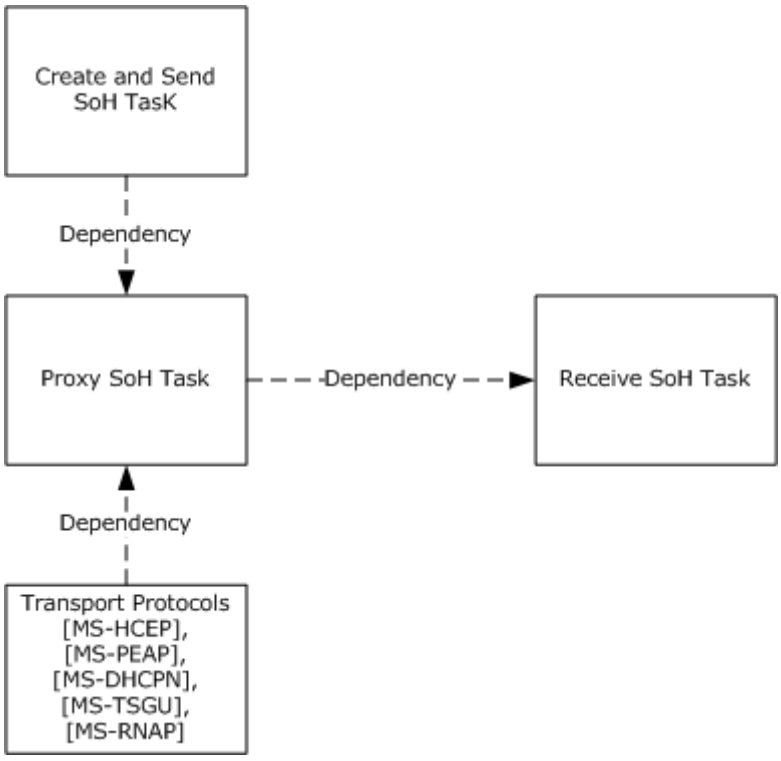

Figure 24: Proxy SoH Task black-box relationships

In this task, the NAP EC sends encapsulated SoH messages to the NAP health policy server via a PEP transport channel.

7.2.2.2 Task Dependencies

The Proxy SoH Task depends on the Create and Send SoH Task. This is because the Proxy SoH Task must rely on the Create and Send SoH Task to generate an SoH message and send it to the NAP Enforcement Proxy so that the NAP Enforcement Proxy can transfer this SoH message to the NAP health policy server.

The Receive SoH Task has a dependency on the Proxy SoH Task. Without the SoH message transferred by the Proxy SoH Task, there is no use of the Receive SoH Task.

This task also depends on the various underlying transport protocols that govern the PEP channels (such as [MS-HCEP], [MS-DHCPN], [MS-TSGU], [MS-PEAP], and [MS-RNAP]).

7.2.2.3 Task Influences

None.

7.2.3 Task Assumptions and Preconditions

To accomplish this task, the NAP client has the following preconditions and assumptions:

- The operating system and hardware of the NAP Enforcement Proxy computer are trustworthy.
- All NAP enforcement servers are available, correctly configured, and functioning correctly.
- Authentication information was transferred successfully by the underlying transport protocols.

7.2.4 Task Versioning and Capability Negotiation

The Proxy SoH Task does not define any versioning and capability negotiation beyond those described in the specifications of the protocols supported or used by the task.

7.3 Task Architecture

This section describes the structure of the Proxy SoH Task and the interrelationships among its parts.

Note This task uses the **PEP Channel Used** ADM element (section 4.1.1). All other common information defined in section 4 is not applicable to this task.

7.3.1 Task Architectural Constraints

There can be more than one instance of the Proxy SoH Task if multiple PEP channels are deployed. These task instances initialize themselves each time they start and run independently. Different instances of this task on different PEPs also run independently.

7.3.2 Task Abstract Data Model

This section describes state established, used, and maintained by processing rules of this task. State may be volatile or persisted. State may pertain to one, some, or all instances of the task. The task's state consists of the values of the named data elements (also called state variables) presented in this section. The overall organization of the data elements, with their names, is the Abstract Data Model. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Data Model, such association can be achieved in other ways. Implementations may depart from this model so long as their external behavior remains consistent with that described in this document.

PEP Channel Used: A common ADM element, as specified in section 4.1.1.

User Ipv4 Address: The **IPv4** address of the endpoint that is requesting network access. This ADM element is initialized to 0 and set to the IPv4 address by the internet protocol layer.

User Ipv6 Address: The **IPv6** address of the endpoint that is requesting network access. This ADM element is initialized to 0, and when the network access server supports IPv6, the value is set to the IPv6 address by the internet protocol layer.

Version: A string representing the Remote Access Service (RAS) version. This ADM element is initialized when PEP is loaded.

RAS Correlation Id: A GUID that is sent in the RADIUS [RFC2865] Access-Request or Accounting-Request message to uniquely identify a RADIUS session. For more information, see [MS-RNAP] section 3.2.5.1.9.

7.3.3 Task Abstract Parameters

This section describes data passed to an instance of this task at the time it is invoked or triggered. The parameters consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Parameter. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Parameters, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

SoH: A buffer that contains either an SoH message, as specified in [TNC-IF-TNCCSPBSoH], or a PEAP message encapsulating an SoH message, as described in [MS-PEAP] section 2.2.8.2.2.

Authentication data: A collection of RADIUS attributes where each entry contains an attribute type number and a value. The attributes and their type numbers are defined in [RFC2865] section 5.44. The list can be empty. Depending on the transport protocol used, the list of sent parameters is as follows:

- For DHCP:
  - **Machine Name:** A string representing the machine name of the endpoint that is requesting network access, as specified in [MS-DHCPM] section 2.2.1.2.19.
  - **Identity SID:** An account SID of the user requesting access in the format of a binary SID used to authenticate a remote access client, as specified in [MS-DHCPM] section 3.5.1.
  - **Identity Type:** A Boolean value indicating whether the RADIUS server is to perform authentication or only health evaluation.
  - **User Class:** A string representing the user class that is used, as specified in [MS-DHCPN] section 2.2.2.
- For HCEP:
  - **Machine Name:** A string representing the machine name of the endpoint that is requesting network access, as specified in [MS-HCEP] section 2.2.1.4.
- For TSGU:
  - **User Name:** A string representing the name of the Remote Access Service (RAS) client endpoint machine.
  - **Machine Name:** A string representing the machine name of the endpoint that is requesting network access, as specified in [MS-TSGU] section 3.5.1.

7.3.4 Task Abstract Results

This section describes data returned by an instance of this task to its caller. The results consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Result. It is intended to facilitate the reader's conceptual

understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Results, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

None.

7.3.5 White-Box Relationships

The following diagram shows the white-box relationships for the Proxy SoH Task.

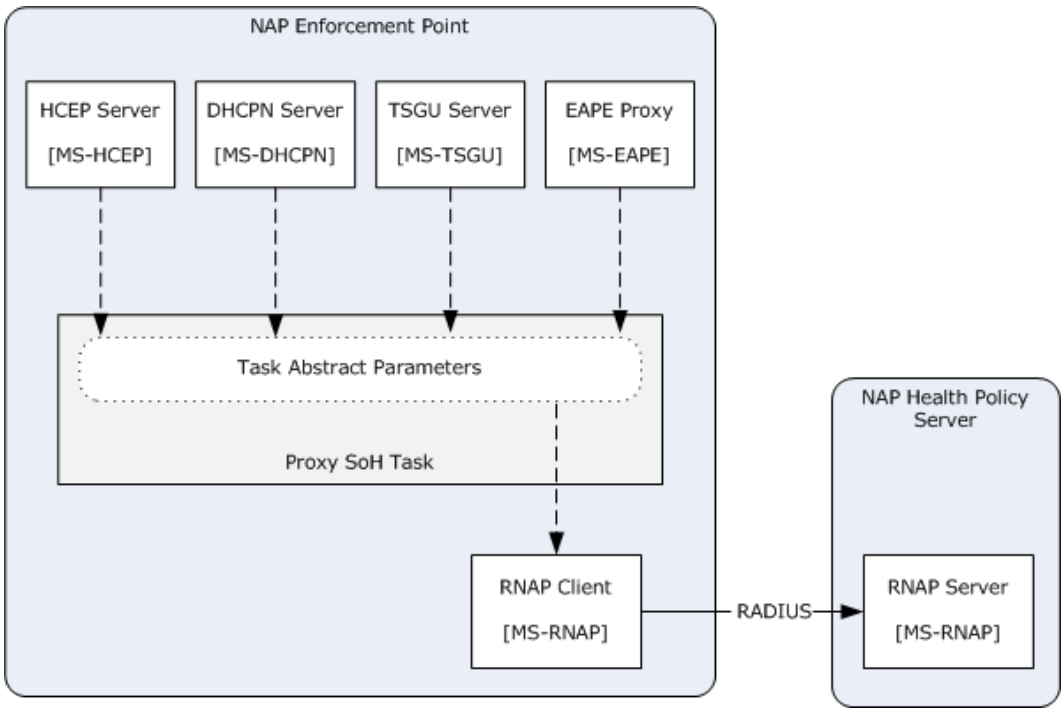

Figure 25: Proxy SoH Task white-box relationships

The white-box relationships of the Proxy SoH Task are shown in the previous figure.

From the Create and Send SoH Task perspective or the NAP health policy server perspective, the Proxy SoH Task provides SoH transportation services. These encapsulated SoH messages are handled by the Proxy SoH Task via various transport channels and are finally consumed by the Process SoH Task on a NAP health policy server.

7.3.6 Task Events

7.3.6.1 Task Timers

The Proxy SoH Task does not impose any additional timers to the outside entities other than the timers in the underlying transport system.

7.3.6.2 Task Non-Timer Events

This task does not use or respond to any additional non-timer events other than those in the underlying transport system.

7.3.7 Task Architecture and Communication

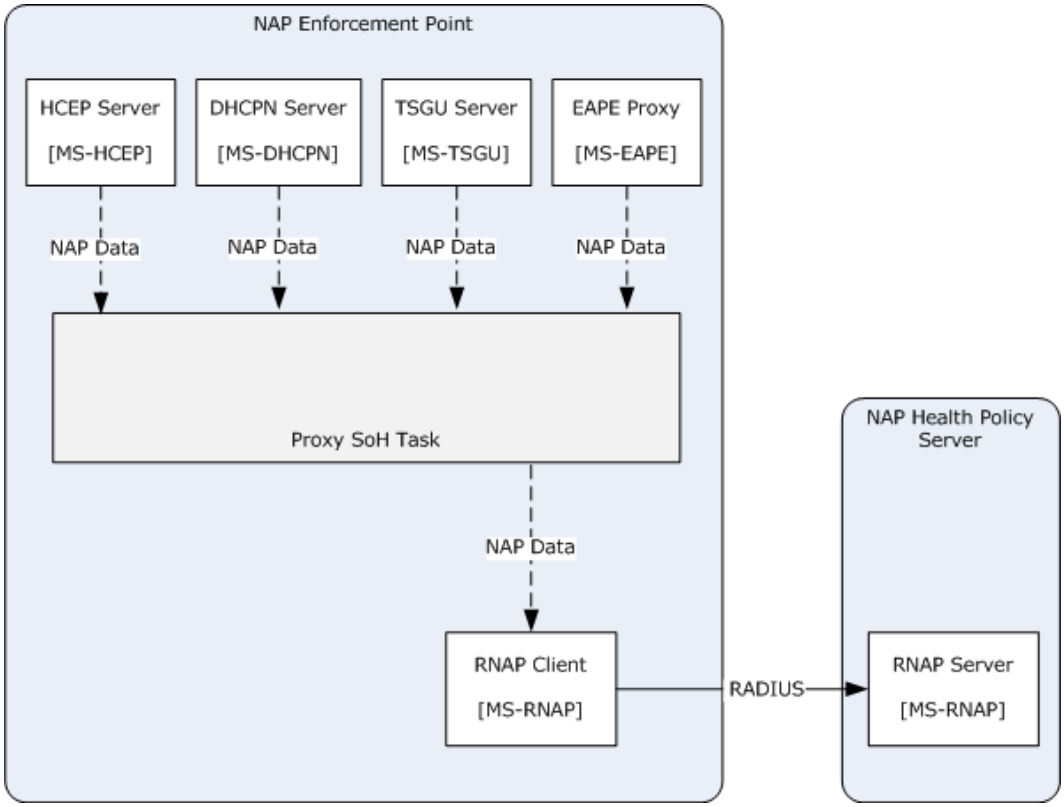

Figure 26: Proxy SoH Task architecture and communication

7.3.8 Task Processing Rules

The following describes the operational flow of the Proxy SoH Task:

1. Either an SoH message is passed into this task by the HCEP HRA [MS-HCEP], DHCP server [MS-DHCPN], or TSGU server [MS-TSGU], or an SoH message encapsulated in a PEAP message [MS-PEAP] is passed into this task by the PEAP Pass-through server. The task sets the **PEP Channel Used** ADM element (section 7.3.2) based on the transport protocol used.

2. Depending on the transport protocol, the NAP Enforcement Proxy does one of the following:

   - If the value of the **PEP Channel Used** ADM element is PEAP, the NAP Enforcement Proxy performs the actions described in [RFC3748] section 2.3. It builds a RADIUS message and injects the EAP message in the SoH parameter into an EAP-Message attribute as specified in [RFC3579] section 3.1. The RADIUS message is sent to the NPS via the EAP-supporting RADIUS client.

- Otherwise, the NAP Enforcement Proxy sends the SoH message in the SoH parameter and the Authentication data via the RNAP client using the **SendRadiusAccessRequest** abstract interface as defined in [MS-RNAP] section 3.3.4.1.

If an error is raised at any stage of the Proxy SoH Task, the task fails.

7.3.9 Task Failure Scenarios

7.3.9.1 NAP Health Policy Server and NAP Enforcement Proxy Communication

These failures can be caused by:

- Misconfigurations on the NAP health policy server and/or NAP Enforcement Proxy.
- Network connectivity issues wherein the NAP health policy server cannot communicate with the NAP Enforcement Proxy.

If the NAP health policy server cannot communicate with the NAP Enforcement Proxy, the server may not receive any encapsulated SoH messages from the NAP Enforcement Proxy. The system cannot recover from this failure. This failure cannot be detected by the NAP health policy server.

7.4 Task Details

This section contains the details that complete the descriptions in earlier sections of the document. These are needed to understand and implement this task.

7.4.1 Task Precondition Details

For the complete list of task preconditions and assumptions, see section 7.2.3. Details for some of the preconditions are as follows:

- Depending on the specific configuration, any of the required NAP Enforcement Proxy channels (the HTTP/S channel, the PEAP channel, or the DHCP channel) are functioning correctly.

7.4.2 Task Initialization of External Entities

None.

7.4.3 Task Event Details

7.4.3.1 Task Timer Details

This task does not impose any additional timers. Timers are related to the underlying transports and they are described in [MS-DHCPN], [MS-PEAP], and [MS-RNAP].

7.4.3.2 Task Non-Timer Event Details

This task does not impose any additional non-timer events. Non-timer events are related to the underlying transports and they are described in [MS-DHCPN], [MS-PEAP], and [MS-RNAP].

7.4.4 Task Architectural Details

This section gives an example of a NAP Enforcement Proxy proxying an SoH.

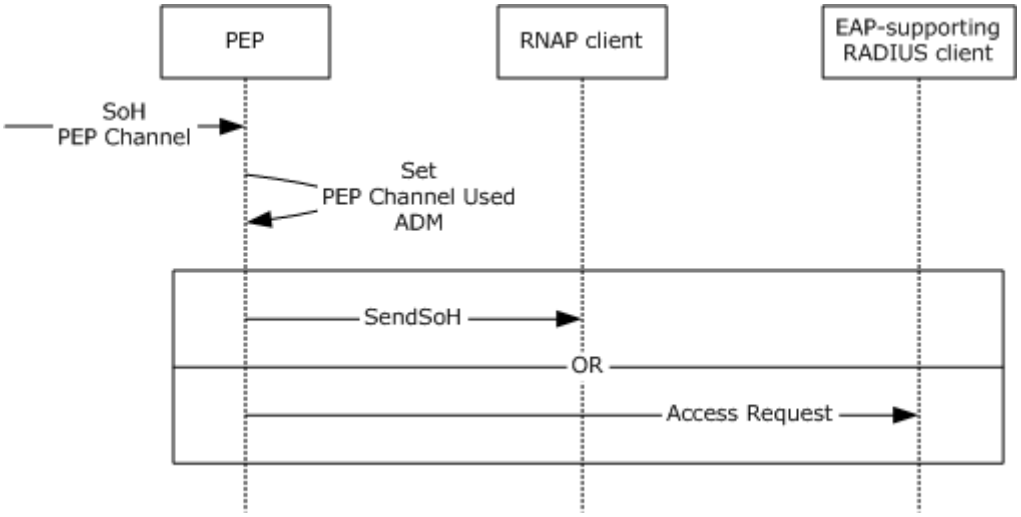

Figure 27: Sequence diagram for the main success scenario of the Proxy SoH Task

1. The NAP Enforcement Proxy receives the SoH message and a NAP Enforcement Proxy Channel enumerator.

2. The NAP Enforcement Proxy sets the **PEP Channel Used** ADM element (section 7.3.2) according to the transport protocol that was used to send the SoH message.

3. Based on the value of **PEP Channel Used**, the NAP Enforcement Proxy sends the SoH message to either the RNAP client or an EAP-supporting RADIUS client.

7.4.5 Task Processing Rule Details

The following describes the operational flow details of the Proxy SoH Task:

1. This task is initiated when the NAP Enforcement Proxy receives an encapsulated SoH as follows:

   - HCEP HRA triggers this task as described in [MS-HCEP] section 3.2.5.2 step 4. It passes an SoH message in the *SoH* and *Authentication data* parameters.
   - The DHCP server triggers this task as described in [MS-DHCPN] section 3.2.7.1. It passes an SoH message in the *SoH* and *Authentication data* parameters.
   - The TSGU server triggers this task as described in [MS-TSGU] section 3.2.6.1.2 step 6. It passes an SoH message in the *SoH* and *Authentication data* parameters.
   - The PEAP Pass-through server triggers this task when an EAP message is received. It passes an SoH message encapsulated in an EAP message in the *SoH* parameter.

2. The NAP Enforcement Proxy sets the value of the **PEP Channel Used** ADM element (section 7.3.2) as follows:

   - **PEP Channel Used** = DHCP, if the transport protocol that was used to send the SoH message is [MS-DHCPN].
   - **PEP Channel Used** = HCEP, if the transport protocol that was used to send the SoH message is [MS-HCEP].

- **PEP Channel Used** = **TSG**, if the transport protocol that was used to send the SoH message is [MS-TSGU].
- **PEP Channel Used** = PEAP, if the transport protocol that was used to send the SoH message is [MS-PEAP].

3. If the value of the **PEP Channel Used** ADM element is PEAP, the NAP Enforcement Proxy performs the actions described in [RFC3748] section 2.3. It builds a RADIUS message and injects the EAP message in the *SoH* parameter into an EAP-Message attribute as specified in [RFC3579] section 3.1. The RADIUS message is sent to the NPS via the EAP-supporting RADIUS client.
4. If the value of the **PEP Channel Used** ADM element is DHCP, HCEP, or TSG, the NAP Enforcement Proxy creates a unique **RAS Correlation Id** ADM element and sends the SoH message and the **Authentication data** abstract parameter via the RNAP client using the **SendRadiusAccessRequest** abstract interface as defined in [MS-RNAP] section 3.3.4.1. The sent interface parameters are as follows:
   - **message** – The SoH message passed in the **SoH** abstract parameter.
   - **clientName** – If the client name was received as part of the **Authentication data**, the **clientName** value is sent; otherwise, a value of NULL is sent.
   - **clientVersion** – The value of the **Version** ADM element (section 7.3.2).
   - **securityIdentity** – If the identity SID was received as part of the **Authentication data**, the **securityIdentity** value is sent; otherwise, a value of 0 is sent.
   - **identityType** – If the identity type was received as part of the **Authentication data**, the **identityType** value is sent; otherwise, a value of false is sent.
   - **serviceClass** – If the service class was received as part of the **Authentication data**, the **serviceClass** value is sent; otherwise, a value of NULL is sent.
   - **networkAccessServerType** – The value of the **PEP Channel Used** ADM element (section 7.3.2).
   - **machineName** – If the machine name was received as part of the **Authentication data**, the **machineName** value is sent; otherwise, a value of NULL is sent.
   - **rasCorrelationId** – The value of the **RAS Correlation Id** ADM element (section 7.3.2).
   - **userIpv4Address** – The value of the **User Ipv4 Address** ADM element (section 7.3.2).
   - **userIpv6Address** – The value of the **User Ipv6 Address** ADM element (section 7.3.2).

7.5 Task Security

The NAP Enforcement Proxy and the NAP health policy server must maintain a trust relationship. For additional information about security considerations, see section 16, as well as the Security sections of the referenced protocol Technical Documents.

8 Receive SoH Task

This section describes the task of receiving SoH messages on the NAP health policy server. The format of the SoH message is specified in [TNC-IF-TNCCSPBSoH]. The protocols that can be used to accomplish this task are specified in [TNC-IF-TNCCSPBSoH], [MS-RNAP], RADIUS [RFC2865], RADIUS Support for Extensible Authentication Protocol (EAP) [RFC3579], and [MS-PEAP].

Note The common information defined in section 4 is not applicable to this task.

8.1 Task Overview

8.1.1 Task Purpose

The purpose of this task is to ensure the reception of SoH messages forwarded by a PEP computer using the Proxy SoH Task (section 7) from a PEAP Server [MS-PEAP] or an RNAP Server [MS-RNAP].

8.1.2 Task Applicability

This task is used when a PEP computer forwards an SoH message as described in the Proxy SoH Task (section 7) to the NAP health policy server using RADIUS Support for EAP [RFC3579] or RNAP [MS-RNAP]. This task is not applicable if the NAP System is not deployed.

8.1.3 Task Use Cases

8.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Receive SoH Task are as follows:

Policy Engine: Responsible for receiving an SoH message sent from a client computer and forwarded by a PEP computer using RNAP [MS-RNAP] or RADIUS support for EAP [RFC3579], and passing the SoH message to the SoH Server. The main interest of the Policy Engine is to pass the SoH message to the SoH Server.

SoH Server: Responsible for verifying the validity and processing of SoH messages. The interest of the SoH Server in this task is that received SoH messages are passed to the SoH Server.

RNAP Server: This protocol server receives RNAP messages [MS-RNAP] sent from a PEP computer and extracts the SoH. Its interest in this task is that extracted SoH messages are processed.

EAP Supporting RADIUS server: This protocol server receives RADIUS/EAP messages [RFC3579] sent from a PEP computer, extracts EAP messages, and uses the PEAP Server to extract SoH messages from the EAP messages. Its interest in this task is that extracted SoH messages are processed.

8.1.3.2 Supporting Actors and Task Interests Summary

PEAP Server: The use case employs this actor to extract SoH messages from EAP messages as specified in [MS-PEAP]. The EAP messages are sent by a NAP Client computer and forwarded by a PEP computer using RADIUS support for EAP [RFC3579].

8.1.3.3 Use Case Diagrams

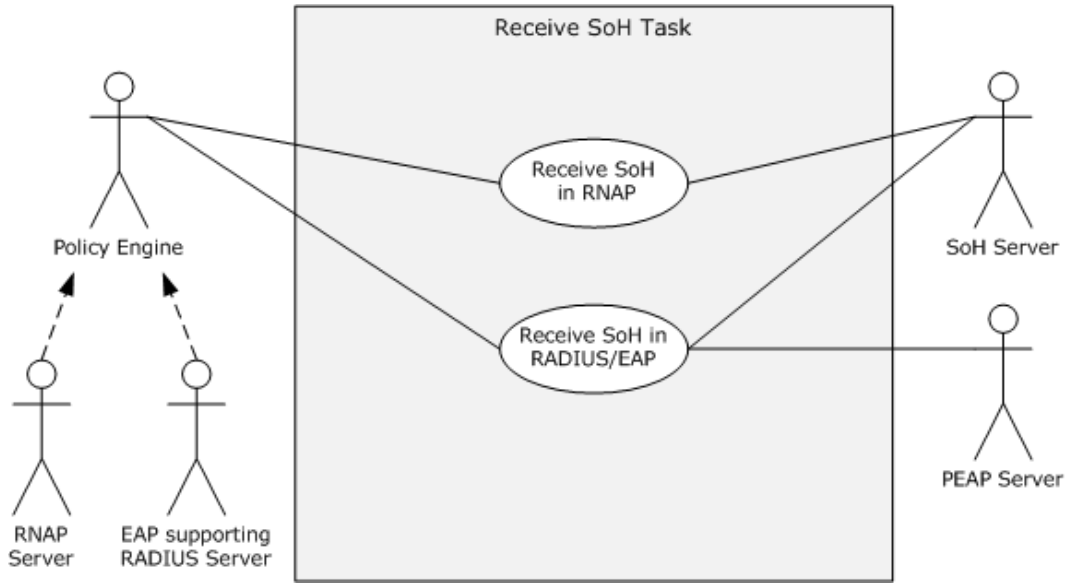

Figure 28: Receive SoH use case diagram

8.1.3.4 Use Case: Receive SoH -- Policy Engine (RNAP)

This use case is associated with the use case diagram in section 8.1.3.3.

Goal: To receive the SoH [TNC-IF-TNCCSPBSoH] forwarded by a PEP computer using the RNAP protocol [MS-RNAP] and to pass the SoH to the SoH Server.

Context of Use: This use case is initiated when the NAP health policy server receives an SoH message sent using the RNAP protocol [MS-RNAP].

Direct Actor: The direct actor in this use case is the Policy Engine.

Primary Actor: The primary actor in this use case is the RNAP Server.

Supporting Actors: There are no supporting actors in this use case.

Stakeholders and Interests: The stakeholders are defined as follows:

- **SoH Server**: Responsible for verifying the validity and processing of SoH messages. The interest of the SoH Server in this task is that successfully received and extracted SoH messages are passed to it.

Preconditions: The NAP health policy server components on the server are deployed and configured correctly by the server administrator.

Minimal Guarantees:

- The NAP software components will continue to execute regardless of the task outcome.
- Successfully extracted SoH messages are passed to the SoH Server.

Success Guarantee: All of the Minimal Guarantees are satisfied. Additionally, an SoH message forwarded by a PEP computer using the RNAP protocol [MS-RNAP] is successfully received and passed to the SoH Server.

Trigger: The arrival of an RNAP message [MS-RNAP] to the NPS computer.

Main Success Scenario:

1. The RNAP Server successfully receives an RNAP message [MS-RNAP].
2. The RNAP Server extracts the SoH message and passes it as an abstract parameter to this task.
3. The Policy Engine passes the SoH to the SoH Server.

Extensions: None.

8.1.3.5 Use Case: Receive SoH – Policy Engine (RADIUS/EAP)

This use case is associated with the use case diagram in section 8.1.3.3.

Goal: To receive the SoH [TNC-IF-TNCCSPBSoH] encapsulated in an EAP message forwarded by a PEP computer using the RADIUS support for EAP [RFC3579] protocol and to pass the SoH to the SoH Server.

Context of Use: This use case is initiated when the NAP health policy server receives an SoH message encapsulated in an EAP message and sent using the RADIUS support for EAP [RFC3579] protocol.

Direct Actor: The direct actor in this use case is the Policy Engine.

Primary Actor: The primary actor in this use case is the EAP supporting RADIUS Server.

Supporting Actors: The supporting actor in this use case is the PEAP Server.

Stakeholders and Interests: The stakeholders are defined as follows:

- **SoH Server:** Responsible for verifying the validity and processing of SoH messages. The interest of the SoH Server in this task is that successfully received and extracted SoH messages are passed to the SoH Server.
- **PEAP Server:** The use case employs this actor to extract SoH messages from EAP messages as specified in [MS-PEAP]. The EAP messages are sent by a NAP Client computer and forwarded by a PEP computer using RADIUS support for EAP [RFC3579].

Preconditions: The NAP health policy server components on the server are deployed and configured correctly by the server administrator.

Minimal Guarantees:

- The NAP software components will continue to execute regardless of the task outcome.
- Successfully extracted SoH messages are passed to the SoH Server.

Success Guarantee: All of the Minimal Guarantees are satisfied. Additionally, an SoH encapsulated in an EAP message that is forwarded by a PEP computer using the RADIUS support for EAP [RFC3579] protocol is successfully received, extracted, and passed to the SoH Server.

Trigger: The arrival of a RADIUS/EAP message [RFC3579] to the NPS computer.

Main Success Scenario:

1. The EAP supporting RADIUS Server successfully receives a RADIUS/EAP message [RFC3579].
2. The EAP supporting RADIUS Server extracts the EAP message and passes it to the PEAP Server.
3. The PEAP Server extracts the SoH message and returns it to the EAP supporting RADIUS server.
4. The EAP supporting RADIUS Server passes the SoH message as an abstract parameter to this task.
5. The Policy Engine passes the SoH to the SoH Server.

Extensions: None.

8.2 Task Context

This section describes the relationship between this task and its environment.

8.2.1 Task Environment

This task is accomplished by the NAP health policy server in an environment where the SoH messages forwarded by a PEP computer using RNAP [MS-RNAP] or RADIUS support for EAP [RFC3579] protocols have arrived at the server.

To accomplish this task, the NAP health policy server requires the following from its environment:

- **Requirement**: The Policy Engine, RNAP Server, EAP supporting RADIUS Server, PEAP Server, and SoH Server are correctly configured.
  - **Reason for requirement**: Correct configuration is required for the following reasons:
    1. The RNAP Server to receive and extract SoH messages and pass them to the Policy Engine.
    2. The EAP supporting RADIUS Server to receive and extract EAP messages and pass them to the PEAP Server.
    3. The PEAP Server to extract SoH messages from EAP messages and pass them to the Policy Engine.
    4. The Policy Engine to pass SoH messages to the SoH Server.
    5. The SoH server to receive SoH messages.
  - **Satisfying the requirement**: The NPS is configured by the system administrator.
  - **Verifying requirement is satisfied**: No errors related to configuration are logged by the RNAP Server, PEAP Server, or SoH Server.
  - **Consequences of not satisfying requirement**: The task is unable to receive the SoH.
- **Requirement**: There is network connectivity between the PEP computer and the NPS computer.
  - **Reason for requirement**: The PEP computer communicates with the NPS computer.
  - **Satisfying the requirement**:
    1. The network interface of the NPS computer is configured to operate on the local subnet.

2. The physical network path (network devices, Ethernet cables, and so on) between the local subnet and the PEP computer is connected.

3. All network devices between the local subnet and the PEP computer are configured to allow packet flow between the two entities.

4. The network infrastructure that provides name and address resolution and routing services is functional.

- **Verifying requirement is satisfied**: The NPS computer can successfully ping the PEP computer over the network.
- **Consequences of not satisfying requirement**: SoH messages cannot be received.

- **Requirement**: The RNAP Server is running and has the ability to communicate with the RNAP Client on the PEP computer using the RNAP [MS-RNAP] protocol.
  - **Reason for requirement**: The RNAP Server is used to receive SoH messages forwarded using the RNAP [MS-RNAP] protocol.
  - **Satisfying the requirement**: The RNAP Server component is enabled.
  - **Verifying requirement is satisfied**: No errors are logged by the RNAP Server.
  - **Consequences of not satisfying requirement**: The SoH cannot be received using the RNAP [MS-RNAP] protocol.
- **Requirement**: The EAP supporting RADIUS Server is running and has the ability to communicate with the EAP supporting RADIUS Client on the PEP computer using the RADIUS support for EAP [RFC3579] protocol.
  - **Reason for requirement**: The EAP supporting RADIUS Server is used to receive EAP messages sent using the RADIUS support for EAP [RFC3579] protocol.
  - **Satisfying the requirement**: The supporting RADIUS Server component is enabled.
  - **Verifying requirement is satisfied**: No errors are logged by the supporting RADIUS Server.
  - **Consequences of not satisfying requirement**: The SoH cannot be received using the RADIUS support for EAP [RFC3579] protocol.
- **Requirement**: The PEAP Server is running and has the ability to handle EAP messages sent by the PEAP Pass-Through Server on the PEP computer.
  - **Reason for requirement**: The PEAP Server is used to extract the SoH encapsulated within [MS-PEAP] packets.
  - **Satisfying the requirement**: The PEAP Server component is enabled.
  - **Verifying requirement is satisfied**: No errors are logged by the PEAP Server.
  - **Consequences of not satisfying requirement**: The SoH cannot be received using the RADIUS support for EAP [RFC3579] protocol.

8.2.2 Task Relationships

8.2.2.1 Black-Box Relationship Diagrams

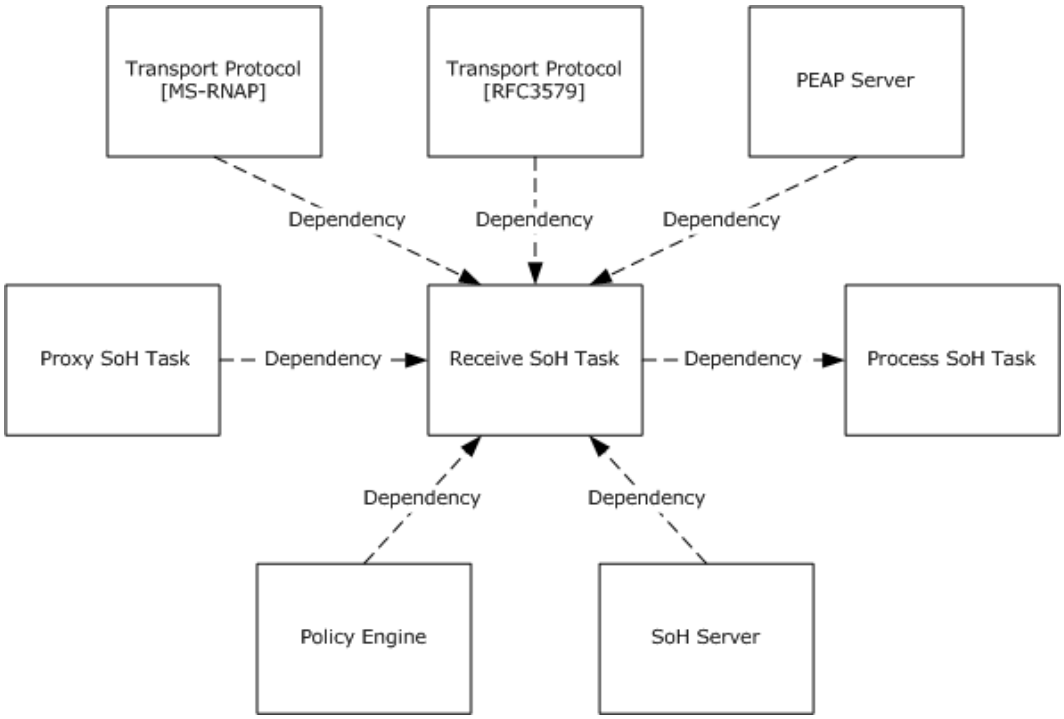

Figure 29: Receive SoH Task black-box relationships

In this task, the NAP health policy server receives SoH messages forwarded by a PEP computer using the RNAP [MS-RNAP] or RADIUS support for EAP [RFC3579] protocols.

8.2.2.2 Task Dependencies

The Receive SoH Task has a dependency on the Proxy SoH Task. Without the SoH messages forwarded from a PEP computer using the Proxy SoH Task, there is no use of the Receive SoH Task.

The Process SoH Task has a dependency on the Receive SoH Task. The NAP health policy server relies on the Receive SoH Task to get the SoH message first and pass it to the Process SoH Task.

The Receive SoH Task depends on the Policy Engine to receive the SoH message from the RNAP Server or PEAP Server and pass it to the SoH Server.

The Receive SoH Task depends on the RNAP Server to receive and extract SoH messages forwarded using the RNAP [MS-RNAP] protocol.

The Receive SoH Task depends on the EAP supporting RADIUS Server to receive and extract EAP messages forwarded using the RADIUS support for EAP [RFC3579] protocol.

The Receive SoH Task depends on the PEAP Server to extract SoH messages from EAP messages as described in [MS-PEAP].

The Receive SoH Task depends on the SoH Server to receive the SoH message from the task and process it.

8.2.2.3 Task Influences

None.

8.2.3 Task Assumptions and Preconditions

To accomplish this task, the NAP health policy server has the following preconditions and assumptions:

- The operating system on the server is trustworthy.
- The server administrators are trustworthy. The server administrators are responsible for deploying and configuring the NAP health policy server correctly. They are also responsible for the integrity of executables that provide NAP health policy server services.
- The underlying network infrastructures, such as the RADIUS channels, name and address resolution, and routing services, are configured correctly.
- The NAP health policy server is correctly configured by the server administrator.

8.2.4 Task Versioning and Capability Negotiation

The Receive SoH Task does not define any versioning and capability negotiation beyond those described in the specifications of the protocols supported or used by the task, as listed in section 2.3.

8.3 Task Architecture

This section describes the structure of the Receive SoH Task and the interrelationships among its parts.

8.3.1 Task Architectural Constraints

There can be more than one instance of the Receive SoH Task on each server. These task instances initialize themselves each time they start. These task instances run independently and concurrently. Different instances of this task on different servers also run independently. There are no constraints among these instances.

8.3.2 Task Abstract Data Model

This section describes state established, used, and maintained by processing rules of this task. State may be volatile or persisted. State may pertain to one, some, or all instances of the task. The task's state consists of the values of the named data elements (also called state variables) presented in this section. The overall organization of the data elements, with their names, is the Abstract Data Model. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Data Model, such association can be achieved in other ways. Implementations may depart from this model so long as their external behavior remains consistent with that described in this document.

ConnectionRequestPolicies: Sets of conditions and settings that specify which RADIUS servers perform the authentication, authorization, and accounting of connection requests received by the NPS server from RADIUS clients. Connection request policies can be configured to designate which RADIUS servers are used for RADIUS accounting. For more information, see [MSFT-ConnReqPolicies].

RequestorIdentity: Represents the identity of the entity requesting access and includes user name, machine name, user groups, machine groups, and so on.

8.3.2.1 Task Abstract Interfaces

SetSoH: An abstract interface used by the RNAP Server or EAP supporting RADIUS Server to pass the SoH to the Policy Engine. The interface is implemented by the Policy Engine and is defined as follows:

```
HRESULT SetSoH([in] SoH message);
```

SetRequestorIdentity: An abstract interface used by the underlying RADIUS implementation of the RNAP Server or the EAP supporting RADIUS Server to pass information about the requestor's identity. This information is a result of the RADIUS authentication process described in [RFC2865]. The interface is implemented by the Policy Engine and is defined as follows:

```
HRESULT SetRequestorIdentity([in] Identity identity);
```

The implementation of the interface stores the requestor's identity information in the **RequestorIdentity** ADM element (section 8.3.2). The Create and Send SoHR Task (section 10) uses this information for policy evaluation.

8.3.3 Task Abstract Parameters

This section describes data passed to an instance of this task at the time it is invoked or triggered. The parameters consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Parameter. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Parameters, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

SoH: SoH message received using the RNAP protocol [MS-RNAP] or RADIUS support for the EAP protocol [RFC3579]. The format of the SoH message is specified in [TNC-IF-TNCCSPBSoH].

8.3.4 Task Abstract Results

This section describes data returned by an instance of this task to its caller. The results consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Result. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Results, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

None.

8.3.5 White-Box Relationships

The following diagram shows the white-box relationships for the Receive SoH Task.

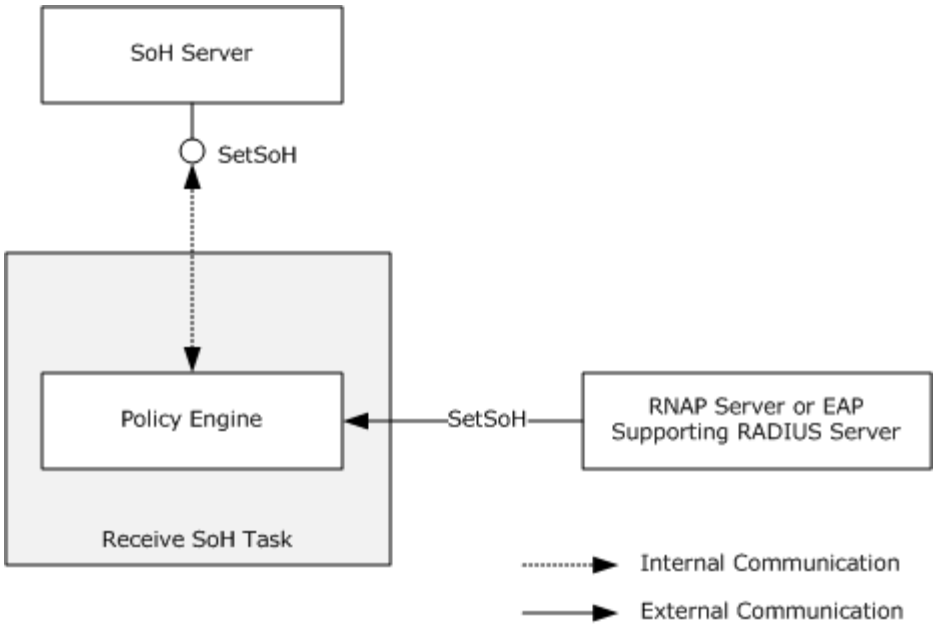

Figure 30: Receive SoH Task white-box relationships

The Receive SoH Task involves two major components: Policy Engine and SoH Server.

From the perspective of the Process SoH Task or the NAP health policy server, the Receive SoH Task receives the SoH messages so that they can be consumed later by the SoH Server. The Policy Engine within the NAP health policy server receives the SoH message from the RNAP Server or the EAP supporting RADIUS Server.

8.3.6 Task Events

8.3.6.1 Task Timers

The Receive SoH Task does not impose any additional timers to the outside entities other than the timers in the underlying transport system.

8.3.6.2 Task Non-Timer Events

This task does not use or respond to any additional non-timer events other than those in the underlying transport system.

8.3.7 Task Architecture and Communication

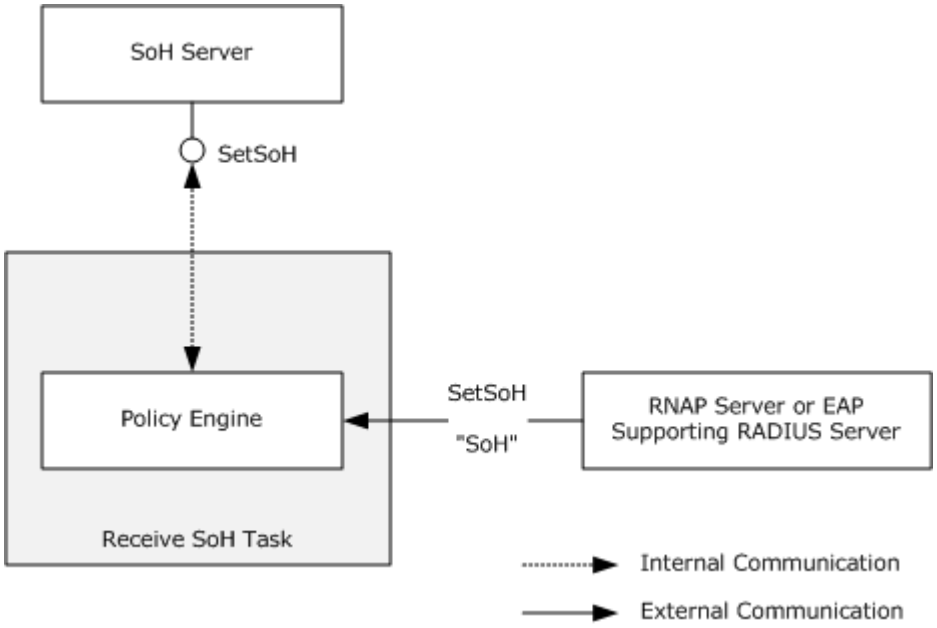

Figure 31: Receive SoH Task architecture and communication

As shown in the previous figure, the Policy Engine on the NAP health policy server receives the SoH message from the RNAP Server or the EAP supporting RADIUS Server. After receiving the SoH messages, the Policy Engine passes them to the SoH Server.

8.3.8 Task Processing Rules

The following describes the operational flow of the Receive SoH Task:

1. The RNAP Server receives a RADIUS Access-Request message encapsulating an SoH according to the RNAP protocol [MS-RNAP], or the EAP supporting RADIUS Server receives a RADIUS Access-Request message encapsulating an SoH according to the RADIUS support for EAP [RFC3579] and [MS-PEAP] protocols.

2. The RNAP Server or the EAP supporting RADIUS server processes the RADIUS request by first attempting to find a matching connection request policy and handle the request accordingly. For the Receive SoH Task, the implementation can assume that the chosen connection request policy specifies local processing of the RADIUS Access-Request message.

3. If RNAP [MS-RNAP] is used, the RNAP Server extracts the SoH message from RNAP.

4. If RADIUS support for EAP [RFC3579] is used, the EAP supporting RADIUS Server extracts the EAP message from RADIUS/EAP. The PEAP Server then extracts the SoH message from the EAP message.

5. The Policy Engine passes the SoH message to the SoH Server.

If an error is raised at any stage of the Receive SoH Task, the task fails.

8.3.9 Task Failure Scenarios

8.3.9.1 NAP Health Policy Server and PEP Communication

These failures can be caused by:

- Misconfigurations on the NAP health policy server and/or PEP.
- Network connectivity issues wherein the NAP health policy server cannot communicate with the PEP.

If the NAP health policy server cannot communicate with the PEP, the server may not receive any encapsulated SoH messages from the PEP. The system cannot recover from this failure. This failure cannot be detected by the NAP health policy server.

8.4 Task Details

This section contains the details that complete the descriptions in earlier sections of the document. These are needed to understand and implement this task.

8.4.1 Task Precondition Details

For the complete list of task preconditions and assumptions, see section 8.2.3. Details for some of the preconditions are as follows:

- The NAP Policy Engine is deployed, configured, and running correctly on the server.
- The RNAP Server and EAP supporting RADIUS Server are functioning correctly.
- The PEAP Server is running correctly.
- The SoH Server is running correctly.

8.4.2 Task Initialization of External Entities

None.

8.4.3 Task Event Details

8.4.3.1 Task Timer Details

This task does not impose any additional timers. Timers are related to the underlying transports and are defined in [MS-RNAP] and [MS-PEAP].

8.4.3.2 Task Non-Timer Event Details

This task does not impose any additional timers. Timers are related to the underlying transports and are defined in [MS-RNAP] and [MS-PEAP].

8.4.4 Task Architectural Details

This section illustrates an example of a NAP health policy server receiving an SoH. The NAP health policy server will utilize several Policy Engine and SoH functions to accomplish the request, as shown in the following diagram.

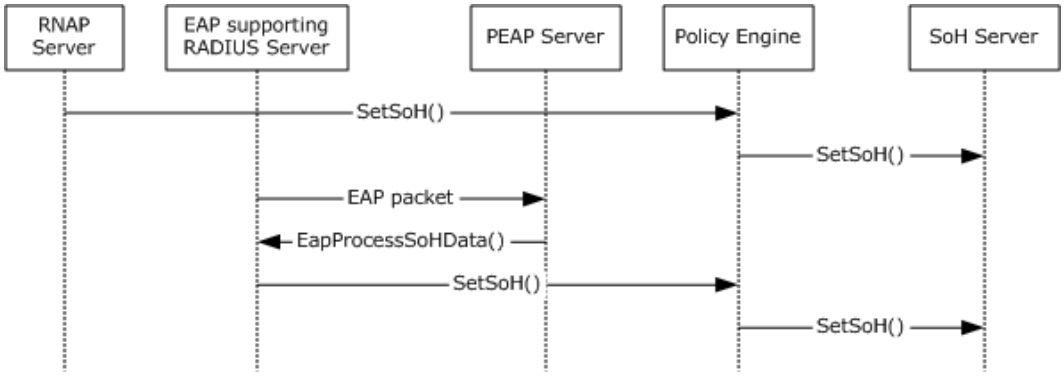

Figure 32: Sequence diagram for the main success scenario of the Receive SoH Task

1. The RNAP Server extracts an SoH and passes it to the Policy Engine.
2. The Policy Engine passes the SoH to the SoH Server.
3. The EAP supporting RADIUS Server extracts an EAP message and passes it to the PEAP Server.
4. The PEAP Server extracts an SoH and returns it to the EAP supporting RADIUS Server.
5. The EAP supporting RADIUS Server passes the SoH to the Policy Engine.
6. The Policy Engine passes the SoH to the SoH Server.

8.4.5 Task Processing Rule Details

The following describes the operational details of the Receive SoH Task:

1. The RNAP Server (as a RADIUS server), or the EAP supporting RADIUS server, processes the attributes of the RADIUS Access-Request message against the set of configured connection request policies represented by the ConnectionRequestPolicies ADM element specified in section 8.3.2. Connection request policies are sets of conditions and settings that allow network administrators to designate which RADIUS servers perform the authentication and authorization of incoming RADIUS Access-Request messages that the Policy Engine receives from RADIUS clients. Specifically, a RADIUS server can perform authentication and authorization itself or forward the request to other RADIUS servers, as described in [RFC2865]. For more information about connection request policies, see [MSFT-ConnReqPolicies].

   The RNAP Server or the EAP supporting RADIUS server applies the settings of the first connection request policy whose configured conditions match the RADIUS attributes or other aspects of the RADIUS Access-Request message, such as the day and time received. The applied settings determine whether the Access-Accept message is processed locally by the Policy Engine or forwarded to another RADIUS server. If the Access-Accept message does not match any connection request policies, the Policy Engine responds with a RADIUS Access-Reject message.

   For the Receive SoH Task, the implementation can assume that the chosen connection request policy specifies local processing of the Access-Request message: the Authentication setting on the Settings tab of the first matching connection request policy is set to either "Authenticate requests on this server" or "Accept users without validating credentials".

2. Depending on the transport protocol, the SoH is received as follows:

1. If the Vendor-Specific RADIUS Attributes for Network Access Protection (NAP) Data Structure [MS-RNAP] is used, the RNAP Server extracts the SoH message and passes it as an abstract parameter to this task, as described in section 8.3.2.1 and [MS-RNAP] section 3.2.5.2.3.

2. If RADIUS support for EAP [RFC3579] is used:

    1. The EAP supporting RADIUS Server extracts an EAP message as described in [RFC3579] and passes it to the PEAP Server.

    2. The PEAP Server extracts an SoH message, as described in [MS-PEAP] section 3.3.5.4.6 and returns it to the EAP supporting RADIUS Server using the EapProcessSoHData abstract interface described in [MS-PEAP] section 3.3.7.5.

    3. The EAP supporting RADIUS Server passes the SoH to this task, using the SetSoH abstract interface described in section 8.3.2.1.

3. If the SoH does not exist, the Receive SoH Task fails and the SoH is not evaluated. For information about possible failure scenarios, see section 8.3.9.

4. The Policy Engine passes the SoH to the SoH Server using the SetSoH abstract interface specified in [TNC-IF-TNCCSPBSoH] and triggers the Process SoH Task.

8.5 Task Security

The PEP and the NAP health policy server must maintain a trust relationship. For additional information about security considerations, see section 16, as well as the Security sections of the referenced protocol Technical Documents.

9 Process SoH Task

The Process SoH Task describes how the Health Policy Server (SoH Server component) evaluates health data in the SoH message. The protocols that can be used to accomplish this task are specified in [TNC-IF-TNCCSPBSoH] and [MS-WSH].

Note All common information defined in section 4 is not applicable to this task.

9.1 Task Overview

9.1.1 Task Purpose

The purpose of this task is to evaluate health data in the SoH message.

9.1.2 Task Applicability

This task is triggered when an SoH message is received by the NAP health policy server using the Receive SoH Task.

9.1.3 Task Use Cases

9.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Process SoH Task are as follows:

SoH Server: Responsible for evaluating the SoH message received from the Receive SoH Task and communicating with the SHVs to perform the evaluation process. The primary interest of the SoH Server is that the task always sets only valid health evaluation results.

Policy Engine: Responsible for receiving an SoH message and passing it to the SoH Server. The main interest of the Policy Engine in the Process SoH Task is to pass the SoH message to the SoH Server.

Create and Send SoHR Task: The purpose of the Create and Send SoHR Task is to ensure that the results of the health evaluation are collected into an SoHR and the SoHR is sent back to the PEP. As such, the Create and Send SoHR Task has to be assured that it will receive only valid health evaluation results.

Remediate Client Health Task: The purpose of the Remediate Client Health Task is to remediate the client computer based on the health evaluation results. As such, the Remediate Client Health Task has to be assured that only valid health evaluation results are received in the SoHR message.

9.1.3.2 Supporting Actors and Task Interests Summary

Policy Configuration Manager: This actor maintains the **Policy Configuration** ADM element (section 10.3.2) that contains the enforcement behavior instructions. This actor also includes a user interface which enables retrieving, creating, updating, and deleting policy configuration information from the **Policy Configuration** ADM element. The use case contacts the Policy Configuration Manager to obtain policy information for evaluation.

9.1.3.3 Use Case Diagrams

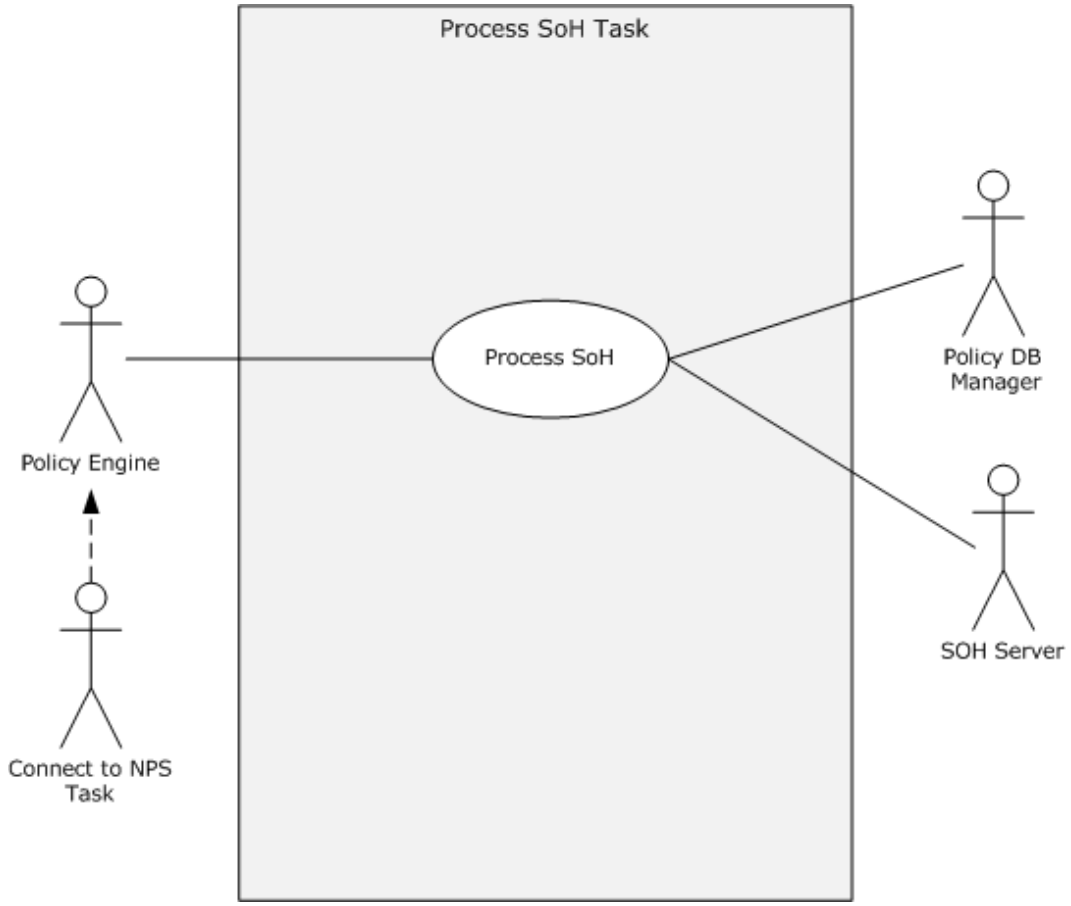

Figure 33: Process SoH Task use case diagram

9.1.3.4 Use Case: Process SoH -- SoH Server

This use case is associated with the use case diagram in section 9.1.3.3.

Goal: To set health evaluation results in the **Health evaluation result** ADM element (section 10.3.3).

Context of Use: The SoH Server receives an SoH message.

Direct Actor: The direct actor in this use case is the SoH Server. The purpose of this actor is to utilize the processing rules defined in [TNC-IF-TNCCSPBSoH] to communicate with SHVs, to evaluate the SoH message, and to set the health evaluation results in the **Health evaluation result** ADM element (section 10.3.3). The primary interest of the SoH Server is that the task always sets only valid health evaluation results.

Primary Actor: The primary actor in this use case is the Policy Engine. The Policy Engine is responsible for receiving an SoH message and passing it to the SoH Server. The main interest of the Policy Engine in the Process SoH Task is to pass the SoH message to the SoH Server.

Supporting Actors: The supporting actor is the Policy Configuration Manager as specified in section 9.1.3.2.

Stakeholders and Interests: The stakeholders are defined as follows:

- **Create and Send SoHR Task:** The purpose of the Create and Send SoHR Task is to ensure that the results of the health evaluation are collected into an SoHR and that the SoHR is sent back to the PEP. As such, the Create and Send SoHR Task has to be assured that it will receive only valid health evaluation results.
- **Remediate Client Health Task:** The purpose of the Remediate Client Health Task is to remediate the client computer based on the health evaluation results. As such, the Remediate Client Health Task has to be assured that only valid health evaluation results are received in the SoHR message.

Precondition: The NAP health policy server components on the server are deployed and configured correctly by the server administrator.

Minimal Guarantees:

1. The Policy Engine will always pass the SoH message to the SoH Server.
2. The task always creates only valid health evaluation results.

Success Guarantee: The SoH Client evaluates the information in the SoH message that was received correctly.

Trigger: The trigger is the arrival of a syntactically correct SoH message to the SoH Server.

Main Success Scenario:

1. The SoH Server successfully receives the SoH message from the Policy Engine.
2. The SoH Server extracts health information from the SoH message.
3. The SoH Server communicates with SHVs successfully and sends them the health information.
4. The SHVs communicate health evaluation results to the SoH Server.
5. The SoH Server sets the health evaluation results in the **Health evaluation result** ADM element (section 10.3.3).

Extensions: None.

9.2 Task Context

This section describes the relationship between this task and its environment.

9.2.1 Task Environment

This task is accomplished by the NAP health policy server in an environment where the NAP health policy server communicates with PEP using a RADIUS channel. This environment requests that the NAP health policy server be deployed and configured correctly on the server.

To accomplish this task, the NAP health policy server requires the following from its environment:

- **Requirement:** The SoH Server is correctly configured as specified in section 9.3.2.

- **Reason for requirement:** Correct configuration is required for the SoH Server to connect to SHVs, to pass health information to the SHVs, and to obtain health evaluation result from the SHVs.
- **Satisfying the requirement:** The PDP is configured by the system administrator.
- **Verifying requirement is satisfied:** No errors related to configuration are logged by the SoH Server or SHVs.
- **Consequences of not satisfying requirement:** The task is unable to obtain the health evaluation results.

- **Requirement:** The enabled SHVs are running and able to process the client health evaluation.
  - **Reason for requirement:** The SHVs provide health evaluation which is returned from this task.
  - **Satisfying the requirement:** The SHVs are enabled.
  - **Verifying requirement is satisfied:** No errors are logged by the SHVs.
  - **Consequences of not satisfying requirement:** The task is unable to process the SoH or the health evaluation results missing.

9.2.2 Task Relationships

9.2.2.1 Black-Box Relationship Diagrams

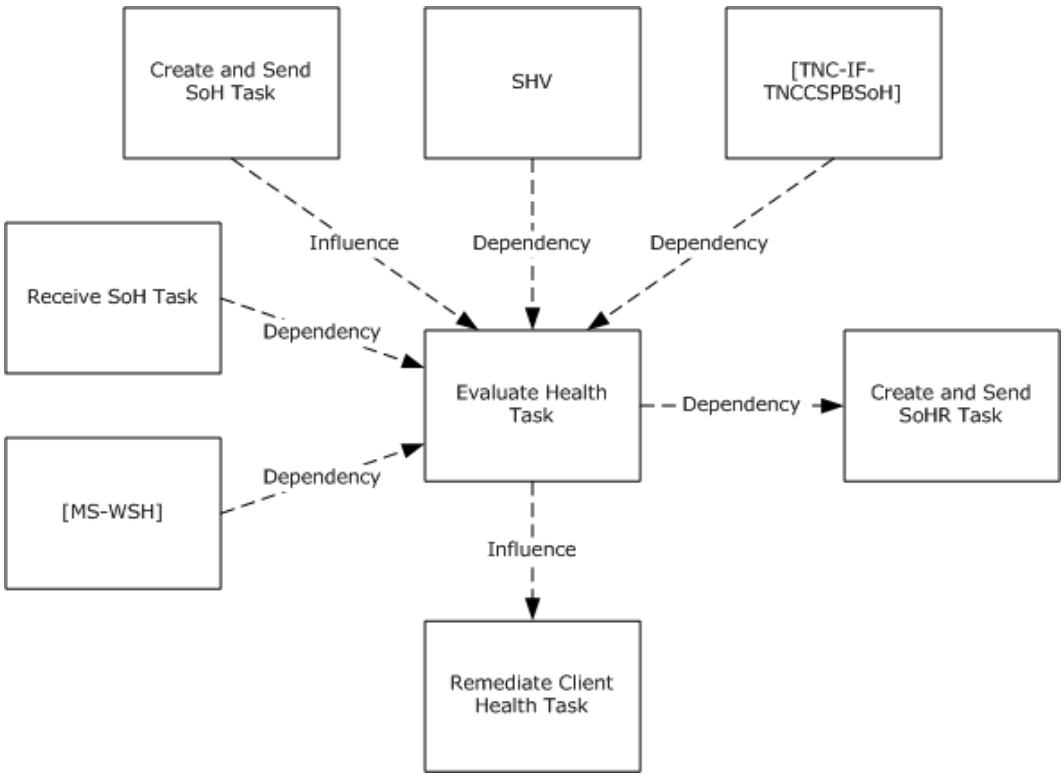

Figure 34: Process SoH Task black-box relationships

The NAP health policy server (PDP) is responsible to perform the health evaluation internally. This task has no relationship to external entities.

9.2.2.2 Task Dependencies

As shown in the previous figure, the Process SoH Task depends on the Receive SoH Task. Without the SoH messages received in the Receive SoH Task, there is no use of this Process SoH Task.

The Create and Send SoHR Task has a dependency on the Process SoH Task because the SoHR creation has to use the evaluation results from this task.

This task is also dependent on the Protocol Bindings for SoH because the NAP health policy server must follow the format defined in [TNC-IF-TNCCSPBSoH] to process the SoH messages.

This task is also dependent on the Security Health Validator (SHV) Protocol [MS-WSH] because the NAP health policy server must follow the format defined in [MS-WSH] to pass health information in SoH to SHV.

This task is also dependent on the SHV to evaluate the health state from the received SoH in order for it to be able to create an SoHR message.

9.2.2.3 Task Influences

This Process SoH Task may have influence on the Remediate Client Health Task because whether or not the Remediate Client Health Task is triggered partially depends on the health evaluation results from this task.

This Process SoH Task is influenced by the Create and Send SoH Task (section 6) as it is the consumer of the SoH generated by the Create and Send SoH Task.

9.2.3 Task Assumptions and Preconditions

To accomplish this task, the NAP health policy server has the following preconditions and assumptions:

- The operating system on the server is trustable to the NAP health policy server.
- The server administrators are trustable to the NAP health policy server. The server administrators are responsible for deploying and configuring the NAP health policy server correctly. They are also responsible for the integrity of executables that provide NAP health policy server services.
- The NAP health policy server is configured correctly by the server administrator.
- The SoH Server successfully initialized and maintains the SHV List.
- Authentication information was processed successfully by the underlying protocol.

9.2.4 Task Versioning and Capability Negotiation

The Process SoH Task does not define any versioning and capability negotiation beyond those described in the specifications of the protocols supported or used by the task, as listed in section 2.3.

9.3 Task Architecture

This section describes the structure of the Process SoH Task and the interrelationships among its parts.

9.3.1 Task Architectural Constraints

There can be more than one instance of the Process SoH Task on each server. These task instances initialize themselves each time they start and they run independently and concurrently. Different instances of this task on different servers also run independently. There are no constraints among these instances.

9.3.2 Task Abstract Data Model

This section describes state established, used, and maintained by processing rules of this task. State may be volatile or persisted. State may pertain to one, some, or all instances of the task. The task's state consists of the values of the named data elements (also called state variables) presented in this section. The overall organization of the data elements, with their names, is the Abstract Data Model. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Data Model, such association can be achieved in other ways. Implementations may depart from this model so long as their external behavior remains consistent with that described in this document.

Registered SHVs list: The SoH Server compiles a list of available SHVs in the system. The list is in the same form as the **System-Health-ID Mapping** ADM element specified in [TNC-IF-TNCCSPBSoH]. The mapping is a tuple: a 32-bit unsigned integer representing the SHV System-Health-ID and a reference to the SHV. The System-Health-ID consists of constants that never change for a given SHV/SHA pair and which represent the IANA SMI Code for the vendor of the SHV/SHA pair.

Health evaluation result: A Boolean used to indicate system health, where Healthy equals a value of True and NotHealthy equals a value of False. The SHV(s) evaluate the health information and return the evaluation results to the SoH Server. If one of the SHV evaluation results is not healthy, the state is changed to NotHealthy.

ShvTimeoutInMsec: A DWORD that specifies the timeout value for the call by the SoH Server to the SHV, in milliseconds. The default value for this ADM element is 2000 milliseconds.

HealthPolicies: One or more system health validators (SHVs) and other settings that can be configured to define client computer configuration requirements for the NAP-capable computers that attempt to connect to the network. For more information, see [MSFT-HealthPolicies].

9.3.3 Task Abstract Parameters

This section describes data passed to an instance of this task at the time it is invoked or triggered. The parameters consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Parameter. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Parameters, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

SoH: SoH message from the Receive SoH Task as specified in [TNC-IF-TNCCSPBSoH].

9.3.4 Task Abstract Results

This section describes data returned by an instance of this task to its caller. The results consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Result. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Results, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

The Process SoH Task is expected to return health evaluation results each time it is called. These results will be used by the Create and Send SoHR Task.

9.3.5 White-Box Relationships

The following diagram shows the white-box relationships between the Process SoH Task and other tasks.

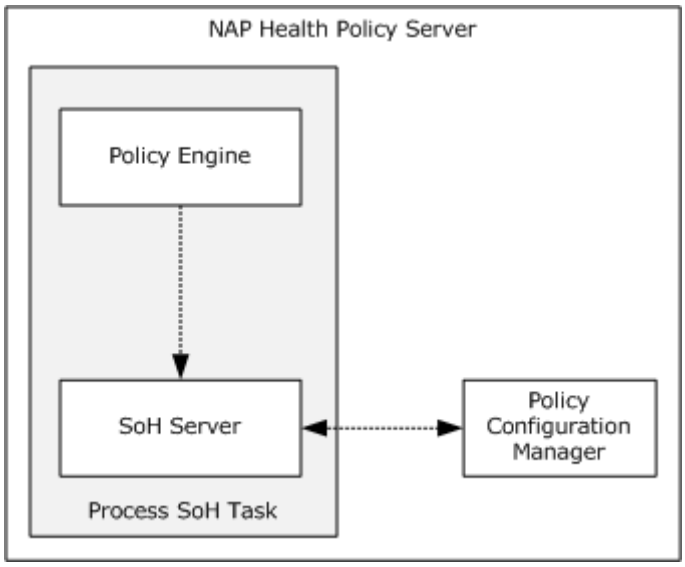

Figure 35: Process SoH Task white-box relationships

The Process SoH Task uses three major NAP health policy server components: SoH Server, Policy Engine, and Policy Configuration Manager.

From the Create and Send SoHR Task's perspective, the Process SoH Task provides health evaluation results and enforcement decisions. In this task, The SoH Server processes the SoH [TNC-IF-TNCCSPBSoH] and then sends the health information stored in the SoH to the installed SHV(s). The SHV(s) evaluates the health information and returns the evaluation results.

9.3.6 Task Events

9.3.6.1 Task Timers

In this task, there is a timer associated with all function calls that the NAP Validator makes into SHVs. This timer determines how soon these function calls must return. This timer can be configured via the Windows registry. Further details can be found in section 9.4.3.1.

9.3.6.2 Task Non-Timer Events

This task does not use or respond to any additional non-timer events.

9.3.7 Task Architecture and Communication

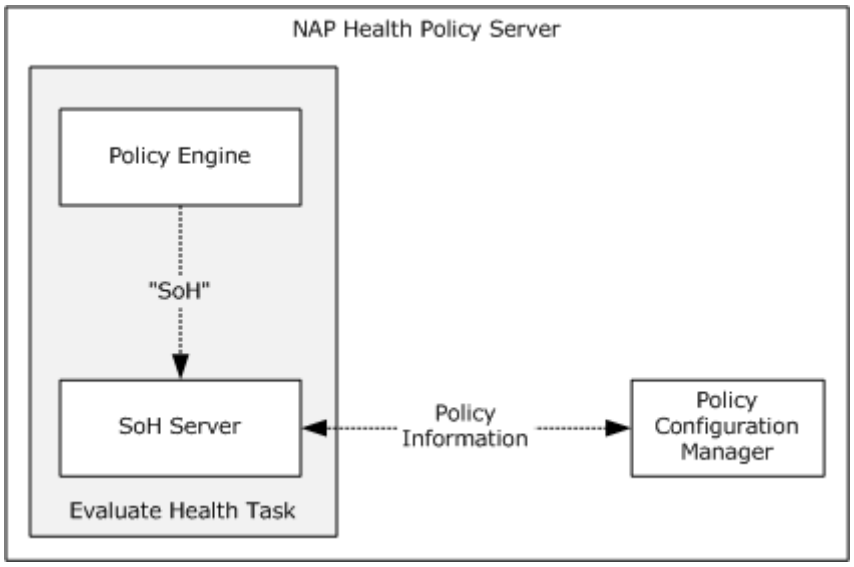

Figure 36: Process SoH Task architecture and communication overview

9.3.8 Task Processing Rules

The following describes the operational flow of the Process SoH Task:

1. The SoH Server successfully receives an SoH message from the Policy Engine.
2. The SoH Server reads the **Registered SHVs List** ADM element upon receiving the SoH.
3. The SoH Server provides health information gathered by the SHA's from the SoH to the corresponding SHV(s).
4. The SHV(s) evaluate health information and pass the results to the SoH Server.
5. The SoH Server processes the results returned from SHV(s) against the settings of health policies and sets the results in the **Health evaluation result** ADM element (section 10.3.3).

If at any stage Process SoH Task is failed, client may be considered as Not Healthy.

9.3.9 Task Failure Scenarios

9.3.9.1 Failures in SHV and SoH Server Communication with SHV

These failures are caused by an error with the initialization, registration, or binding of a SHV. The NAP System relies on its ability to communicate with the installed SHVs in order to evaluate the individual health statement that is designated to this SHV. In this failure scenario either the SHV fails or the SoH Server fails to send the corresponding health statement to the SHV, so the NAP health policy server will not be able to create an SoHR and send it back to the NAP client. The client experiencing this failure will not be able to see the expected SoHR message, which can result in the

client being categorized as unhealthy even if it is healthy. This failure can also cause missing enforcement actions on unhealthy clients. The failures are detected by a timer monitored by the SoH Server. The NAP System provides an error code enabling the administrator to configure fragility settings to detect and override the health policy decision on the PDP.

9.4 Task Details

This section contains the details that complete the descriptions in earlier sections of the document. These are needed to understand and implement this task.

9.4.1 Task Precondition Details

For the complete list of task preconditions and assumptions, see section 9.2.3. Details for some of the preconditions are as follows:

- The SoH Server is operational.
- SHVs are correctly configured, registered, and bound to the SoH Server so that the SoH Server has a complete SHV list.

9.4.2 Task Initialization of External Entities

None.

9.4.3 Task Event Details

9.4.3.1 Task Timer Details

Inside this task there is a timer associated with all function calls that the SoH Server makes into SHVs. When the SoH Server calls into an SHV to perform a health evaluation, a timeout is enforced. The SHV is expected to complete the call within the timeout; otherwise, the call is canceled and an error is reported by the SoH Server. The timeout value is the **ShvTimeoutInMsec** ADM described in section 9.3.2.

9.4.3.2 Task Non-Timer Event Details

None.

9.4.4 Task Architectural Details

This section illustrates an example of a NAP health policy server (PDP) evaluating health information. The SoH Server finds all available SHVs and passes the health information by calling the SHV INapSoHProcessor API. After the evaluation is completed, the SHV calls the INapServerCallback interface with the result. (A complete list of SHV APIs is specified in [MSDN-NAPAPI]).

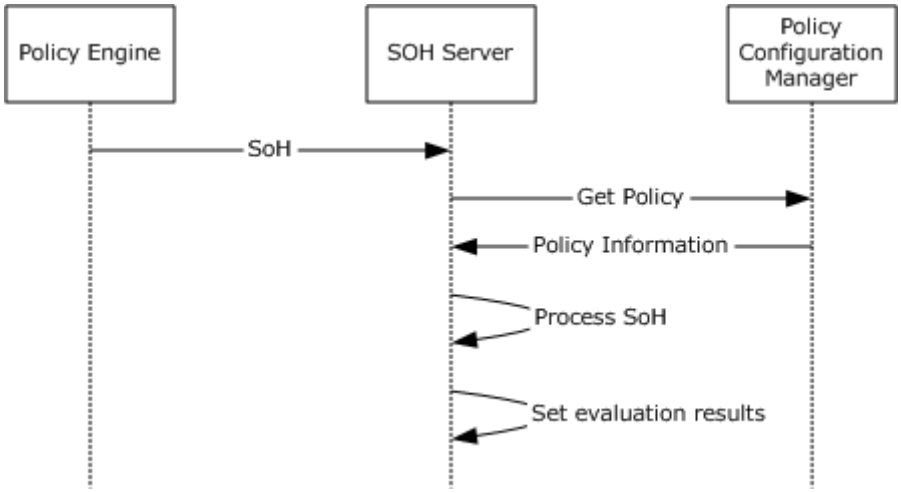

Figure 37: Sequence diagram for the main success scenario of the Process SoH Task

1. The SoH Server receives the SoH message and obtains the policy information from the Policy Configuration Manager.

2. The SoH Server processes the health information and sets the health evaluation results.

9.4.5 Task Processing Rule Details

For the complete list of task preconditions and assumptions, see section 9.2.3. Details for some of the preconditions are as follows:

1. The SoH Server successfully receives an SoH message from the Policy Engine.

2. The SoH Server compares the identity of the client stored in the **RequestorIdentity** ADM element (section 8.3.2) against the value in the health policy.

3. The SoH Server checks which SHV(s) from the **HealthPolicies** ADM element (section 9.3.2) are available on the server (SHVs are registered on the server at initialization and only SHV(s) specified in the **HealthPolicies** ADM element SHOULD be used for health evaluation) and compiles a **Registered SHVs List** (section 9.3.2) as follows:

    1. The SoH Server requests the value of the **NAPSystemHealthID** field from all SHV(s) listed in the **HealthPolicies** ADM element by using the SHV APIs specified in section 3.3.2.

    2. Based on the scan result, the SoH Server updates the **Registered SHVs List** ADM element which contains all of the **NAPSystemHealthID** values received in the previous step.

4. The SoH Server extracts all the **SoHReportEntry** message specified in [TNC-IF-TNCCSPBSoH] from the SoH message, as specified in [TNC-IF-TNCCSPBSoH], and passes each **SoHReportEntry** message to the SHV associated with it, as follows:

    1. For every **SoHReportEntry** message, get its System-Health-ID attribute.

    2. Match the value of the **System-Health-ID** field in the **SoHReportEntry** message to the corresponding **NAPSystemHealthID** in the **Registered SHVs List** to see if the **SoHReportEntry** message has a corresponding SHV API.

3. Pass the **SoHReportEntry** message to the SHV using the **INapSystemHealthValidator::Validate** method, which is part of SHV API, as described in [TNC-IF-TNCCSPBSoH].

5. The SHV(s) evaluates health information and passes the results to the SoH Server.

6. The SoH Server processes the results returned from SHVs against the settings of health policies, represented by the **HealthPolicies** ADM element (section 9.3.2). The SoH Server applies the settings of the first health policy with configured conditions that match the SoH message, such as the day and time received. The applied settings determine whether one or all of the SHVs SHOULD return a value of Healthy to cause the client computer to be treated as compliant. (Non-NAP-capable clients are treated as non-compliant.) The SoH Server sets the **Health evaluation result** ADM element (section 9.3.2) to Healthy or NotHealthy based on the values returned by all of the SHVs and the applied health policy.

9.5 Task Security

There are no task-specific security considerations. For additional information about security considerations, see section 16, as well as the Security sections of the referenced protocol Technical Documents.

10 Create and Send SoHR Task

This section describes the task of creating the statement of health response (SoHR) and of encapsulating and sending SoHR messages from the NAP health policy server (PDP) to PEP. This task takes place at the PDP.

PDP uses this task to communicate with PEP for the following:

- To send the SoHR back to the NAP client.
- To send the enforcement decisions to the PEP. The PEP will perform the enforcement accordingly. For more information, see the Enforce NAP Policy Task (section 11).

The enforcement decisions are sent to PEP directly using RADIUS response messages [RFC3580].

The SoHR messages are encapsulated into RADIUS response messages before they are sent out using the Vendor-Specific RADIUS Attributes for Network Access Protection (NAP) Data Structure [MS-RNAP], or the Protected Extensible Authentication Protocol [MS-PEAP] as specified in [RFC3579]. These protocols are extensions to the standard RADIUS protocol [RFC3580]. An encapsulated SoHR is transported to PEP using the RADIUS channel together with the enforcement decision in the same RADIUS response message.

The format of the SoHR message is specified in [TNC-IF-TNCCSPBSoH]. The format of the enforcement decision properties sent with the SoHR message is specified in [MS-RNAP] and in [MS-PEAP].

10.1 Task Overview

10.1.1 Task Purpose

The purpose of this task is to ensure that the results of the health evaluation are collected into an SoHR and the SoHR is sent back to the PEP. It is also to ensure that Enforcement decisions are correctly sent from the Health Policy Server to the PEP.

A NAP client creates an SoH for health validation. This SoH is transported to the SoH Server for evaluation. The SoH Server uses the installed SHVs and its policies to determine whether the NAP client is compliant with the configured policies. After the health evaluation, the SoH Server creates an SoHR, which indicates whether the NAP client is compliant or non-compliant and sends the SoHR to the sender of the SoH evaluation request. This task details the creation and sending of the SoHR.

10.1.2 Task Applicability

This task is used whenever a NAP health evaluation succeeds and health evaluation is set. After the SoH server processes the contents of the SoH against the configured system health requirement policies, it creates the SoHR indicating if the client is conformant, and the client's level of network restriction. In addition, it sends the SoHR using RNAP.

NAP health evaluations can occur at the initial connection to the network or network resources, periodically, when network state changes, when an element of system health that is being monitored by SHAs running on the NAP client changes or when it is manually triggered by user.

This task is not applicable if the NAP System is not deployed.

10.1.3 Task Use Cases

10.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Create and Send SoHR Task are as follows:

SoH Server: The purpose of this actor is to utilize the processing rules defined in [TNC-IF-TNCCSPBSoH] to create SoHR packets, as specified in [TNC-IF-TNCCSPBSoH]. It is responsible for creating the SoHR, encapsulating the health evaluation results and correlation ID, and calling the RNAP server or PEAP server to send the SoHR. Its main interest is ensuring that the SoHR message will be created and will always include the evaluation results as specified in [TNC-IF-TNCCSPBSoH].

Proxy SoHR Task: The purpose of the Proxy SoHR Task is to proxy the correlation ID, the SoHR packet, and the enforcement data from the RNAP server to the DHCPN, HCEP, TSGU, or PEAP servers. As such, the Proxy SoHR Task has to be assured that the Create and Send SoHR Task only sends protocol messages that contain all three components.

10.1.3.2 Supporting Actors and Task Interests Summary

The supporting actors for the Create and Send SoHR Task and their interests are as follows:

Policy Engine: The Policy Engine is the main software component on the NAP server computer. It is responsible for obtaining the SoHR message and policy configuration information and passing them to the RNAP Server or PEAP Server, depending on the transport protocol used. The use case uses the Policy Engine whenever it transfers SoHR messages and policy enforcement configuration to the RNAP Server or PEAP server.

RNAP Server: This protocol server is responsible for formatting and sending the SoHR to a PEP, using the RNAP protocol [MS-RNAP]. The use case employs this actor whenever an SoHR packet is sent to the PEP, using RNAP.

PEAP Server: This actor is responsible for formatting and sending the SoHR to a PEP, using RADIUS support for EAP [RFC3579]. The use case employs this actor whenever an SoHR packet is sent to the PEP, using PEAP.

Policy Configuration Manager: This actor maintains the <Policy Configuration> ADM element (section 10.3.2) that contains enforcement behavior instructions. This actor also includes a user interface which allows retrieving, creating, updating, and deleting policy configuration information from the <Policy Configuration> ADM element. The use case contacts the Policy Configuration Manager to obtain the policy information.

10.1.3.3 Use Case Diagrams

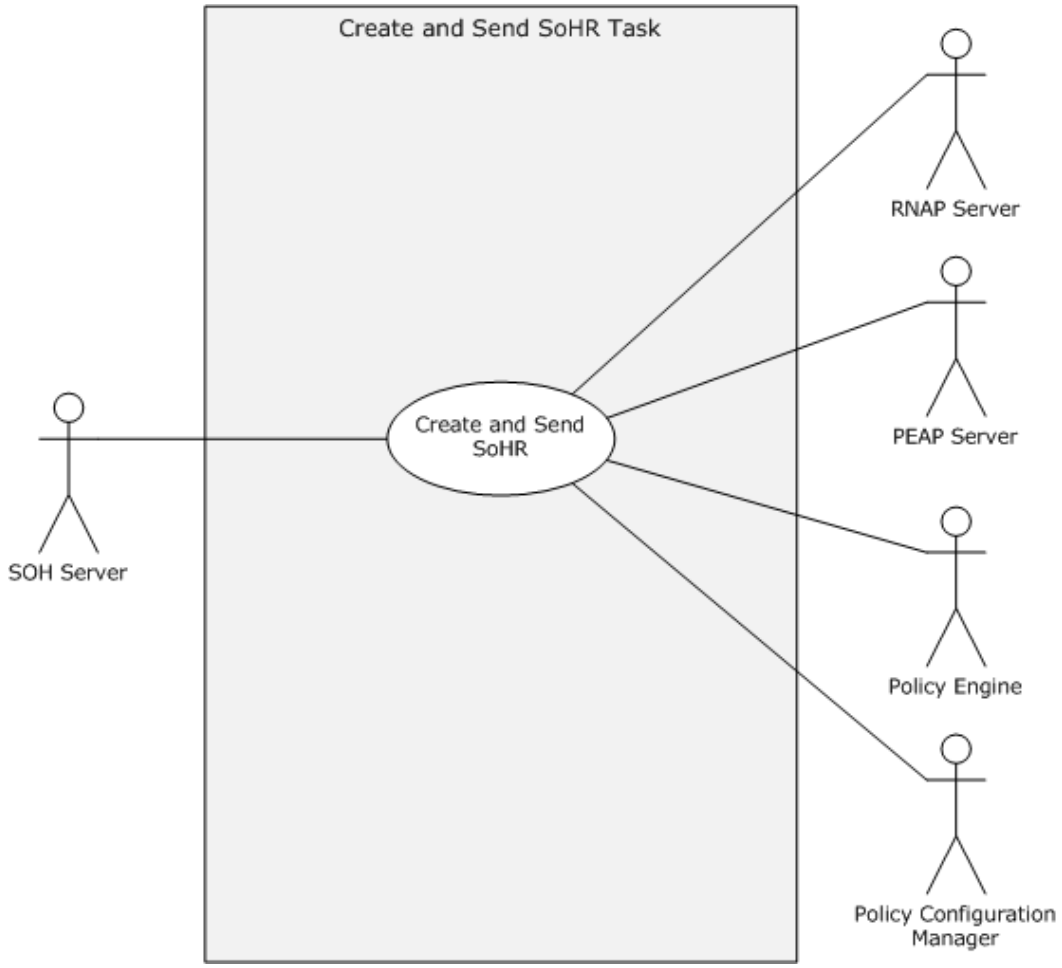

Figure 38: Create and Send SoHR Task use case diagram

10.1.3.4 Use Case: Create and Send SoHR – SoH Server

This use case is associated with the use case diagram in section 10.1.3.3.

Goal: To create an SoHR message [TNC-IF-TNCCSPBSoH] containing health evaluation results for the client computer and send it to the PEP.

Context of Use: This use case is initiated when the SoH evaluation is completed by the SoH Server.

Direct Actor: The direct actor in this use case is the SoH Server. The purpose of this actor is to utilize the processing rules defined in [TNC-IF-TNCCSPBSoH] to create SoHR packets, as specified in [TNC-IF-TNCCSPBSoH]. It is responsible for creating the SoHR, encapsulating the health evaluation results and correlation ID, and calling the RNAP Server or PEAP Server to send the SoHR. Its main interest is ensuring that the SoHR message will be created and will always include the evaluation results as specified in [TNC-IF-TNCCSPBSoH].

Primary Actor: The primary actor is the same as the direct actor.

Supporting Actors: The supporting actors are as listed in section 10.1.3.2.

Stakeholders and Interests: The stakeholders are defined as follows:

- **Proxy SoHR Task:** The purpose of the Proxy SoHR Task is to proxy the correlation ID, the SoHR packet, and the enforcement data from the RNAP Server to the DHCPN, HCEP, TSGU, or PEAP server. As such, the Proxy SoHR Task has to be assured that the Create and Send SoHR Task only sends protocol messages that contain all three components.

Preconditions:

- The NAP health policy server components on the server are deployed and configured correctly by the server administrator.
- PEP is configured so that it can communicate with the Health Policy Server when the RNAP Channel is used.

Minimal Guarantees:

- The task will always create an SoHR message.
- The task always sends only valid health evaluation information in the SoHR message.
- The task always sends only valid enforcement information in the RNAP and PEAP packets.
- No RNAP or RADIUS/EAP messages are sent by the task without a correlation ID, an SoHR packet, and the enforcement data.

Success Guarantee: SoHR is created and sent through the RNAP channel or PEAP channel.

Trigger: Health evaluation results set by the SoH Server.

Main Success Scenario:

1. The SoH Server sets the value of the health evaluation results and triggers this task.
2. The SoH Server requests and receives policy information from Policy Configuration Manager.
3. The SoH Server creates an SoHR, encapsulates the correlation ID and health evaluation results as described in [TNC-IF-TNCCSPBSoH], and then passes it, together with policy enforcement parameters, to the Policy Engine.
4. The Policy Engine passes the SoHR with the policy enforcement parameters to the RNAP Server or PEAP Server depending on the transport protocol that is used.
5. If the transport protocol is as specified in [MS-PEAP]:
    1. The PEAP Server encapsulates the SoHR message as specified in [MS-PEAP] section 2.2.8.1.3. The resulting EAP message is encapsulated using [RFC3579]. The PEAP Server then sends the enforcement decisions as vendor-specific quarantine attributes that instruct the PEP regarding which network access to enforce as a part of the RADIUS Access-Accept or Access-Reject message.
    2. The PEAP Server sends the SoHR to the PEP.
6. If the transport protocol is as described in [MS-HCEP], [MS-DHCPN], or [MS-TSGU]:

1. The RNAP Server encapsulates SoHR messages into a RADIUS Access-Accept or Access-Reject packet as specified in [MS-RNAP] section 2.2.1. In case of an SoHR message that is not compliant, the RNAP Server adds a vendor-specific quarantine attribute that instructs the PEP regarding which network access to enforce.

2. The RNAP Server sends the SoHR to the PEP.

Extensions: None.

10.2 Task Context

This section describes the relationship between this task and its environment.

10.2.1 Task Environment

This task is accomplished by a NAP health policy server acting as a PDP in an environment wherein a PEP is able to receive SoHR messages over an RNAP channel or PEAP channel. The environment must meet several requirements to support this task.

- **Requirement:** The SoH Client, RNAP Server, and PEAP Server are correctly configured as specified in section 10.3.2.
  - **Reason for requirement:** Correct configuration is required for the following:
    - The SoH Server, to compose the SoHR message and send it to the Policy Engine.
    - The Policy Engine, to send the SoHR to the RNAP Server or the PEAP Server.
    - The RNAP Server and the PEAP Server, to send the message to the PEP server.
  - **Satisfying the requirement:** The PDP is configured by the system administrator.
  - **Verifying requirement is satisfied:**
    1. A network capture performed during task execution shows the SoHR encapsulated within RADIUS/RNAP, as specified in the Main Success Scenario.
    2. No errors related to configuration are logged by the SoH Server or RNAP Server.
  - **Consequences of not satisfying requirement:** The task is unable to create and send the SoHR.
- **Requirement:** The task trigger described in section 10.1.3.4 is functioning correctly.
  - **Reason for requirement:** The trigger initiates the task.
  - **Satisfying the requirement:**
    1. The Process SoHR task executes successfully.
    2. SHVs are enabled.
  - **Verifying requirement is satisfied:** A network capture performed immediately after any triggering event shows the SoHR encapsulated within RADIUS/RNAP, as specified in the Main Success Scenario.
  - **Consequences of not satisfying requirement:** The task does not start and as a result, the SoHR is not created and not sent.

- **Requirement:** There is network connectivity between the NAP health policy server (PDP) and PEP servers.
  - **Reason for requirement:** The NPS communicates with the PEP.
  - **Satisfying the requirement:**
    1. The network interface of the PEP server is configured to operate on the local subnet.
    2. The physical network path (network devices, Ethernet cables, and so on) between the local subnet and the NPS is connected.
    3. All network devices between the local subnet and the NPS are configured to allow packet flow between the two entities.
    4. The network infrastructure that provides name and address resolution and routing services is functional.
  - **Verifying requirement is satisfied:** The NPS can successfully ping the PEP over the network.
  - **Consequences of not satisfying requirement:** The SoHR cannot be sent to the PEP.

10.2.2 Task Relationships

10.2.2.1 Black-Box Relationship Diagrams

This task consists of creation and sending of the SoHR after the health evaluation on the PDP. The following diagram illustrates the task to enable the creation of SoHR.

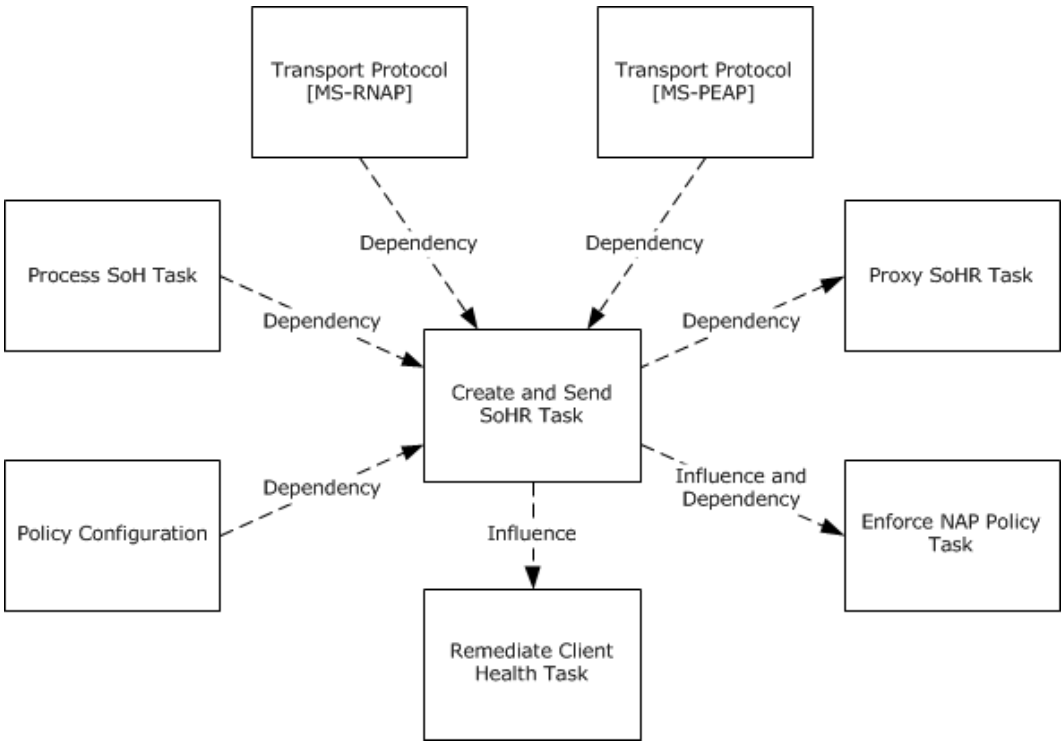

Figure 39: Create and Send SoHR Task black-box relationships

10.2.2.2 Task Dependencies

The Create and Send SoHR Task dependencies are as follows:

- The Process SoH Task (section 9) to provide health evaluation results that will be used to create the SoHR.
- The Proxy SoHR Task (section 12) depends on the Create and Send SoHR Task to provide the SoHR from the NAP health policy server.
- The Enforce NAP Policy Task (section 11) depends on the Create and Send SoHR Task to provide the enforcement parameters in the RNAP or PEAP response.
- The **Policy Configuration** ADM element to provide the information regarding which enforcement to perform in case of NotHealthy evaluation result.
- Vendor-Specific RADIUS Attributes for Network Access Protection (NAP) Data Structure [MS-RNAP] that govern the RNAP channel between PEP and the NAP health policy server (PDP).
- Protected Extensible Authentication Protocol [MS-PEAP] to govern the PEAP channel between PEP and the NAP health policy server (PDP).

10.2.2.3 Task Influences

The Create and Send SoHR Task influences the Enforce NAP Policy Task (section 11). The result of this task calculation on health state and policy configuration yields the enforcement parameters required by the PEP to be able to do the enforcement.

The Create and Send SoHR Task influences the Remediate Client Health Task because the SoHR created will contain information if remediation will be performed.

10.2.3 Task Assumptions and Preconditions

To accomplish this task, the PDP has the following preconditions and assumptions:

- The underlying task triggers, such as the Process SoH Task and the networking modules, are functioning correctly.
- The underlying network infrastructures, such as the RADIUS channel, name and address resolution, and routing services, are configured correctly.
- The NAP health policy server is configured correctly by the server administrator.
- The NAP client is enabled and configured correctly by the client administrator.
- The PDP is configured and can be reached by the PEPs.

10.2.4 Task Versioning and Capability Negotiation

The system does not define any versioning or capability negotiation beyond those described in the specifications of the protocols supported by the system.

10.3 Task Architecture

This section describes the structure of the Create and Send SoHR Task and the interrelationships among its parts.

10.3.1 Task Architectural Constraints

There can be more than one instance of the Create and Send SoHR Task on each server machine. These task instances initialize themselves each time they start. These task instances run independently and concurrently. Different instances of this task on different server machines also run independently.

10.3.2 Task Abstract Data Model

This section describes the state established, used, and maintained by processing rules of this task. State may be volatile or persisted and may pertain to one, some, or all instances of the task. The task's state consists of the values of the named data elements (also called state variables) presented in this section. The overall organization of the data elements, with their names, is the Abstract Data Model. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Data Model, such association can be achieved in other ways. Implementations may depart from this model so long as their external behavior remains consistent with that described in this document.

SoHR: Stores the SoHR which is used as means of communication with [MS-PEAP].

SoHR compliance state: A Boolean that specifies the compliance state of the SoHR, where Compliant equals a value of True.

Policy Configuration: Provides the configuration for the policy as follows:

- **ipv4Filter** – An object of an IPv4Filter abstract type (defined in [MS-RNAP] section 3.2.4.1) that specifies the network access scope of the endpoint.
- **ipv6Filter** - An object of an IPv6Filter abstract type (defined in [MS-RNAP] section 3.2.4.1) that specifies the network access scope of the endpoint.
- **quarantineSessionTimeout** – A DWORD value that specifies the time in seconds that a restricted connection can remain in a restricted state before being disconnected.
- **quarantineGraceTime** – A DWORD value that specifies the end of the time period within which the noncompliant endpoint can obtain full access before being moved to a restricted connection. The time is represented as the number of seconds since 1/1/1970 UTC (GMT).
- **ipv4RemediationServers** – An array of objects of IPv4Address abstract type (defined in [MS-RNAP] section 3.2.4.1) that specifies the addresses of available IPv4 remediation servers.
- **ipv6RemediationServers** – An array of objects of IPv6Address abstract type (defined in [MS-RNAP] section 3.2.4.1) that specifies the addresses of available IPv6 remediation servers.
- **Remediation required** - A Boolean specifying whether remediation is required, where a value of True indicates that remediation is required.
- **dhcpQuarantineUserClass** – A string that specifies the user class to be used when assigning an IP address to an endpoint that is granted restricted access. This field is only relevant when the client is a DHCP server.

- **rdgDeviceRedirection** – A DWORD value that specifies the device redirection options. This field is only relevant when the client is an RDG server; otherwise, it SHOULD be ignored.
- **afwZone** – A DWORD value that specifies the NAP zone, as described in [MS-HCEP]. This parameter is only relevant when the client is an HCEP server; otherwise, it SHOULD be ignored.
- **afwLevel** – A DWORD value that specifies the NAP protection level, as described in [MS-HCEP]. This parameter is only relevant when the client is an HCEP server.

NetworkPolicies: Sets of conditions, constraints, and settings that specify which clients are authorized to connect to the network and the circumstances under which they can connect. For more information, see [MSFT-NetworkPolicies].

10.3.2.1 Task Abstract Interfaces

SetSoHR: An abstract interface used by the SoH Server to send the SoHR to the Policy Engine. The interface is implemented by the Policy Engine and is defined as follows:

```
HRESULT SetSoHR(
    [in] SoHR message,
    [in] DWORD quarantineState,
    [in] DWORD extendedQuarantineState );
```

The Policy Engine does not process any of the received properties, but only forwards them to RNAP Server or PEAP Server depending on the transport protocol that is used.

10.3.3 Task Abstract Parameters

This section describes data passed to an instance of this task at the time it is invoked or triggered. The parameters consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Parameter. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Parameters, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

Health evaluation result: The results of the health evaluations from all SHV's.

10.3.4 Task Abstract Results

This section describes data returned by an instance of this task to its caller. The results consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Result. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Results, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

The Create and Send SoHR Task is expected to return an SoHR message at each time it is called and send the SoHR directly to PEP. The returned SoHR message follows the format defined in [TNC-IF-TNCCSPBSoH]. In addition, an enforcement decision is returned in the RNAP Microsoft VSAs. The transport of the SoH and enforcement decision must follow [RFC3580] and [MS-RNAP].

10.3.5 White-Box Relationships

The following diagram shows the white-box relationships for the Create and Send SoHR Task.

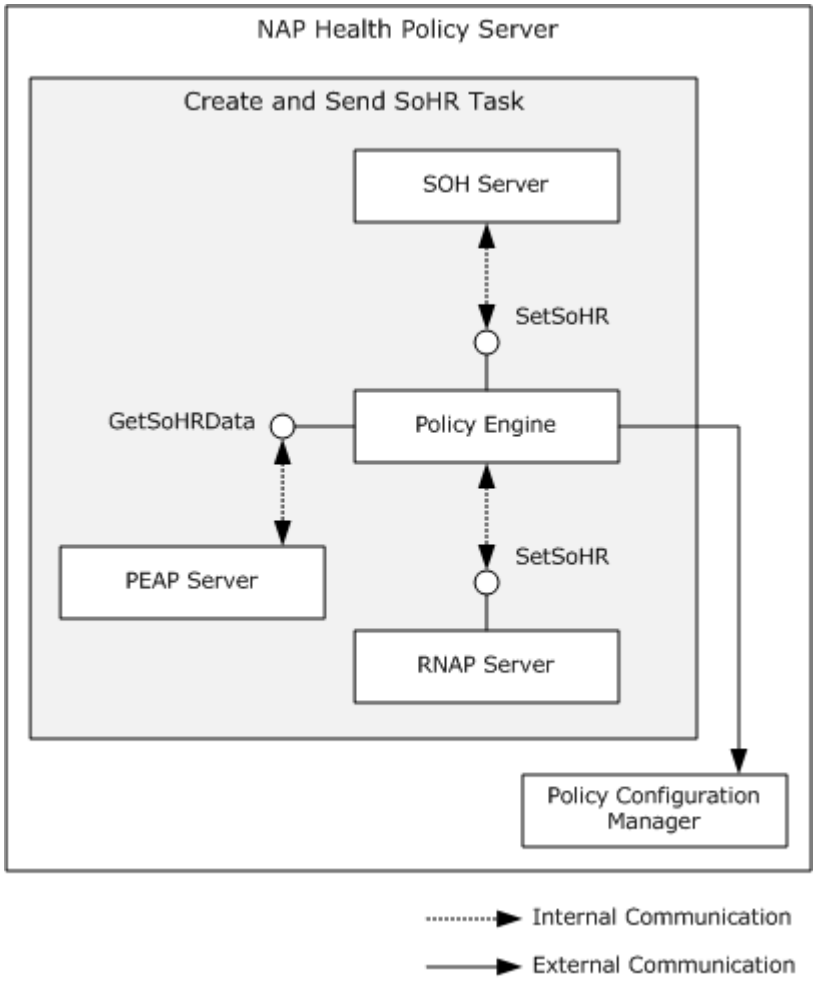

Figure 40: Create and Send SoHR Task white-box relationships

The Create and Send SoHR Task creates and sends the SoHR using the steps outlined below:

1. The SoH Server builds the SoHR header and appends the **Health evaluation results** as attributes as described in [TNC-IF-TNCCSPBSoH] and pass SoHR to Policy Engine.

2. If using RNAP to send the SoHR, the RNAP Server creates the RADIUS Access-Accept or RADIUS Access-Reject message containing the SoHR and the RNAP VSAs [MS-RNAP] and transmits the message to the PEP.

3. If using PEAP to send the SoHR, the PEAP Server creates the RADIUS Access-Accept or RADIUS Access-Reject message containing the SoHR, and transmits the message to the PEP.

10.3.6 Task Events

10.3.6.1 Task Timers

The system does not define any task timers beyond those protocols supported by the system and defined in [MS-RNAP].

10.3.6.2 Task Non-Timer Events

The system does not define any task non-timer events beyond those protocols supported by the system and defined in [MS-RNAP].

10.3.7 Task Architecture and Communication

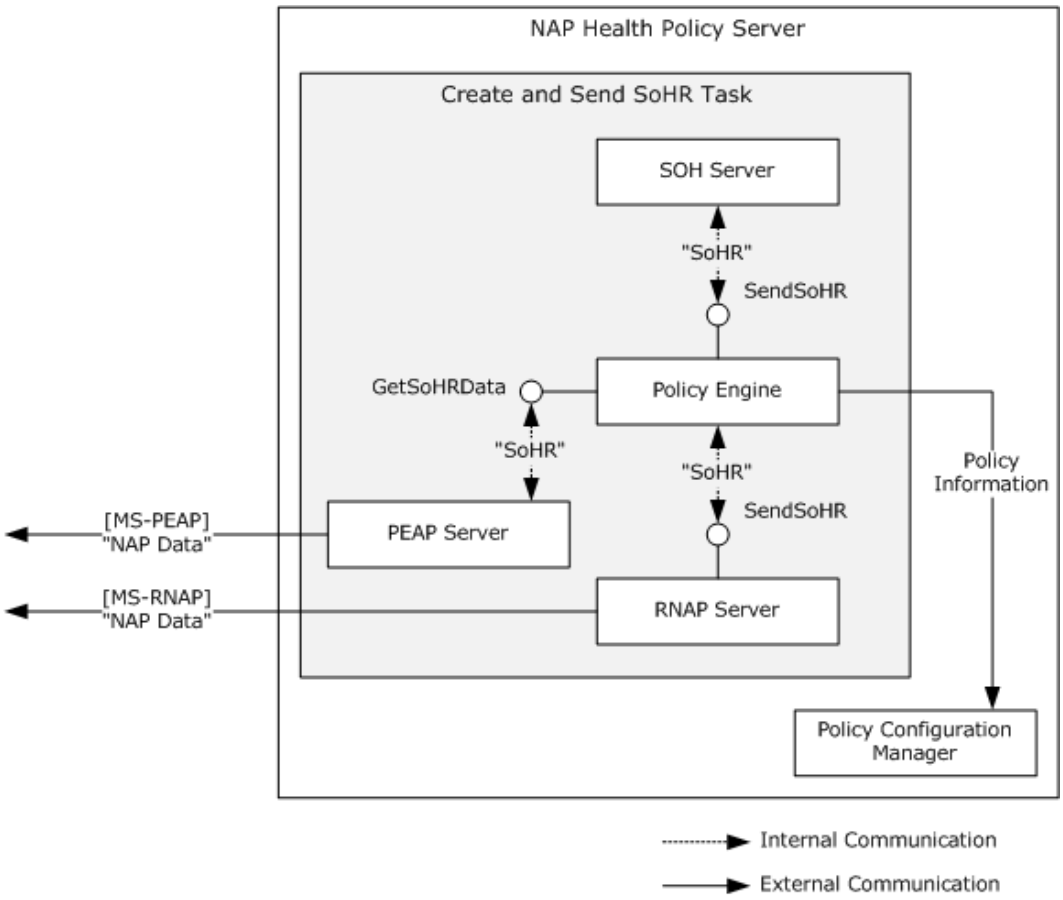

Figure 41: Create and Send SoHR Task architecture and communication

The diagram shows the architectural details and the interworking between the SoH Server, RNAP Server, and PEAP Server components to accomplish the Create and Send SoHR Task, and its supporting (dependent and influenced) tasks.

The Create and Send SoHR Task is triggered by the evaluation of health on the NAP health policy server (PDP). The PDP (SoH Server, Policy Engine, RNAP Server, and PEAP Server) evaluates the

health of the NAP client, creates the SoHR as described in [TNC-IF-TNCCSPBSoH], and sends it directly to PEP.

The Create and Send SoHR Task is triggered on the PDP after the receipt of an evaluated SoH results.

10.3.8 Task Processing Rules

The following describes the operational flow of the Create and Send SoHR Task:

1. The SoH Server sets the value of the health evaluation results and triggers this task.
2. Based on the health evaluation results, the SoH Server sets the **SoHR compliance state** ADM element.
3. The SoH Server retrieves the policy enforcement parameters from the **Policy Configuration** ADM element (section 10.3.2).
4. The SoH Server creates an SoHR, encapsulates the correlation ID and health evaluation results as specified in [TNC-IF-TNCCSPBSoH], and stores it in the **SoHR** ADM element (section 10.3.2).
5. The SoH Server passes the SoHR and policy configuration information to the Policy Engine using the **SetSoHR** abstract interface specified in section 10.3.2. The Policy Engine stores the SoHR in the **SoHR** ADM element (section 10.3.2).
6. The Policy Engine passes the SoHR and policy configuration information received to the RNAP Server using the **SetSoHR** abstract interface specified in [MS-RNAP] section 3.2.4.1, or to the PEAP Server using the **EapGetSoHRData** abstract interface specified in [MS-PEAP] section 3.3.7.5, depending on the transport protocol that is used.
7. If the transport is [MS-PEAP]:
    1. The PEAP Server retrieves the SoHR from the **SoHR** ADM element as specified in [MS-PEAP] section 3.3.5.4.6 and encapsulates the SoHR in the PEAP message using the EAP Extension Method "SoH Response TLV" as specified in [MS-PEAP] section 2.2.8.1.3.
    2. The resulting EAP message is encapsulated using the EAP-Message attribute as specified in [RFC3579] and sent to the PEP.
8. If the transport is as defined in [MS-HCEP], [MS-DHCPN], or [MS-TSGU]:
    1. The RNAP Server creates a RADIUS message containing the SoHR.
    2. The RNAP Server sends the SoHR directly to the PEP as defined in [MS-RNAP].

If an error is raised at any stage of the Create and Send SoHR Task, the task fails.

10.3.9 Task Failure Scenarios

10.3.9.1 SoH Server Communication with RNAP Server

These failures are caused by an internal error either in the RNAP Server or in the SoH Server. The NAP health policy server relies on the communication between the SoH Server and the RNAP Server to provide the transport of the SoH and the SoHR. A server experiencing this failure will not be able to provide health evaluation results and enforcement decisions to the PEP, which can make the affected clients healthy and be put into restricted state. These failures can be detected by the NAP

System using internal error codes. The NAP System cannot recover from such a failure except to restart the NAP health policy server.

10.3.9.2 NAP Health Policy Server and PEP communication

These failures can be caused by:

- Misconfigurations on the NAP health policy server and/or the PEP.
- Network connectivity issues wherein the NAP health policy server cannot communicate with the PEP.

If the NAP health policy server cannot communicate with the PEP, the NAP health policy server may not send any RADIUS messages to the PEP. The system cannot recover from this failure. This failure cannot be detected by the NAP Server because RADIUS uses UDP.

10.3.9.3 NAP Fragility Settings

The NAP System provides PDP fragility settings to change the evaluation that is returned by the PDP under specific error conditions. Fragility settings enable the system to recover from the following failures:

- SHV server is unreachable
- Remediation server unreachable
- SHA failure
- NAP health policy server failure
- All other errors

10.4 Task Details

This section contains the details that complete the descriptions in earlier sections of the document. These are needed to understand and implement this task.

10.4.1 Task Precondition Details

For the complete list of task preconditions and assumptions, see section 10.2.3. Details for some of the preconditions are as follows:

- The RNAP Server is deployed, configured, and running correctly on the server.
- The PEAP Server is deployed, configured, and running correctly on the server.
- The SoH Server is operational.
- The Policy Engine is operational.

10.4.2 Task Initialization of External Entities

None.

10.4.3 Task Event Details

10.4.3.1 Task Timer Details

There are no task timer events for this task.

10.4.3.2 Task Non-Timer Event Details

None.

10.4.4 Task Architectural Details

This section illustrates an example of a PDP (NAP health policy server) creating an SoHR and sending it directly to the PEP. The SoH Server creates the SoHR and passes it to the RNAP Server using the SetSoHR function specified in [MS-RNAP] section 3.2.4.1. The RNAP Server then sends the SoHR using the RNAP channel, or sends the SoHR to the PEAP Server using the **EapGetSoHRData** method, as specified in [MS-PEAP] section 3.3.7.5, which in turn sends it using the PEAP channel.

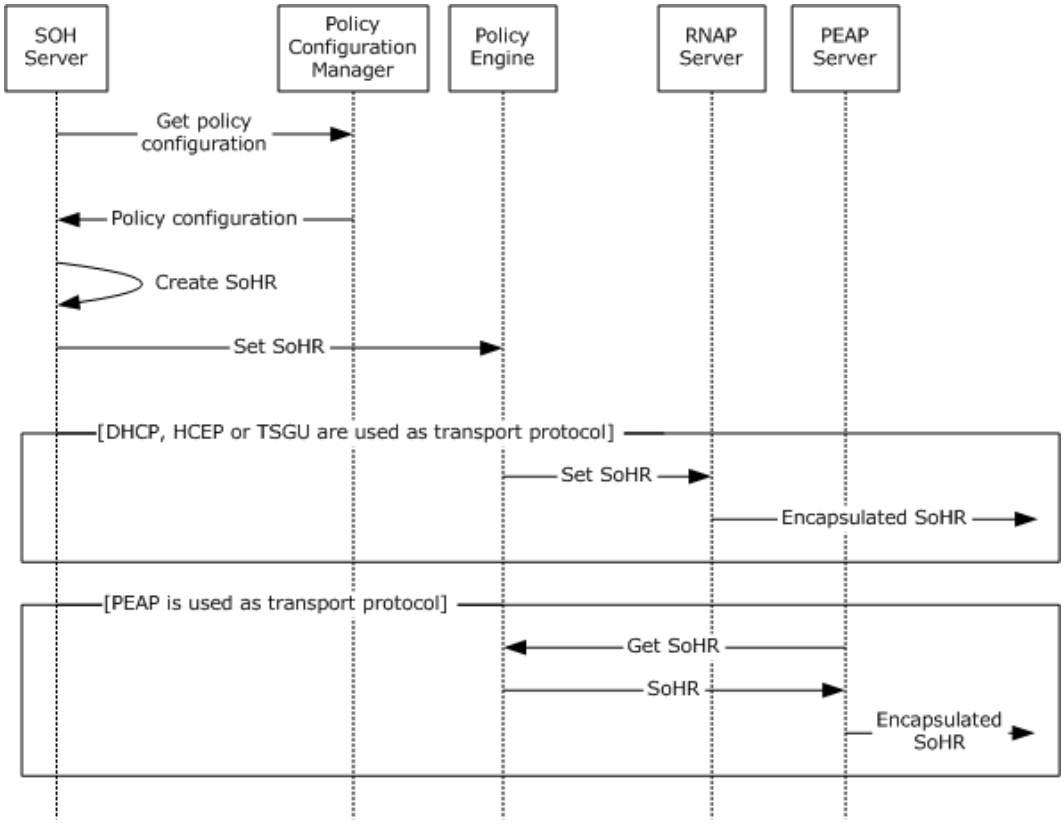

Figure 42: Sequence diagram for the main success scenario of the Create and Send SoHR Task

1. The SoH Server creates an SoHR.
2. The SoH Server forwards the SoHR to the Policy Engine.
3. The Policy Engine stores the SoHR in the **SoHR** ADM element.

4. The Policy Engine forwards the SoHR to the RNAP Server or PEAP Server.

5. If the transport is [MS-PEAP]:

    1. The PEAP Server retrieves the SoHR from the **SoHR** ADM element as specified in [MS-PEAP] section 3.3.5.4.6 and encapsulates the SoHR in the PEAP message using the EAP Extension Method "SoH Response TLV" as specified in [MS-PEAP] section 2.2.8.1.3.

    2. The resulting EAP message is encapsulated using the EAP-Message attribute as specified in [RFC3579] and sent to the PEP via the RNAP channel.

6. When the transport is as defined in [MS-HCEP], [MS-DHCPN], or [MS-TSGU], then:

    1. The RNAP Server creates a RADIUS message containing the SoHR.

    2. The RNAP Server sends the SoHR directly to the PEP via the RNAP Channel as defined in [MS-RNAP].

10.4.5 Task Processing Rule Details

The following describes the operational details of the Create and Send SoHR Task:

1. The SoH Server sets the value of the health evaluation results and triggers this task.

2. The SoH Server requests and receives policy information from the Policy Configuration Manager.

3. The SoH Server creates an SoHR as follows:

    1. Create the SoHR header ([TNC-IF-TNCCSPBSoH]) and sets the value of the correlation ID following the processing rules specified in [TNC-IF-TNCCSPBSoH].

    2. Add all of the **SoHRReportEntry** messages ([TNC-IF-TNCCSPBSoH]) returned by the SHVs as specified in [TNC-IF-TNCCSPBSoH].

    3. Set the **MS-Quarantine-State** VSA **Flags** field, as specified in [TNC-IF-TNCCSPBSoH], to reflect enforcement and remediation decision as follows:

        - The **ExtState** field indicates if the health evaluation results point that the machine was evaluated as an infected machine.

        - The **f** bit field indicates if remediation is required according to **Remediation required** field stored in the **Policy Configuration** ADM element.

        - The **qState** field indicates if the network connectivity is being restricted.

        The exact format and range of values for each field are specified in [TNC-IF-TNCCSPBSoH].

4. The SoH Server reviews the SoHR **MS-Quarantine-State** and sets the **SoHR compliance state** ADM element accordingly. If the value of the **qState** field of the MS-Quarantine-State VSA ([TNC-IF-TNCCSPBSoH]) is 3, then the client is compliant with the NAP health policies; otherwise, it is non-compliant.

5. The SoH Server processes the client authentication information stored in the **RequestorIdentity** ADM element (section 8.3.2) against the policy and determines whether the request should be accepted.

6. The SoH Server processes the attributes of the SoH message and the health evaluation results against the set of configured network policies, represented by the **NetworkPolicies** ADM

element (section 10.3.2). The SoH Server applies the settings of the first network policy with configured conditions that match aspects of the SoH message and health evaluation results. The applied settings of the policy determine whether the request will be accepted.

7. The SoH Server passes the SoHR and policy enforcement information to the Policy Engine using the **SetSoHR** abstract interface specified in section 10.3.2. Interface parameters are sent as follows:

   - **SoHR** – The created SoHR message.
   - **quarantineState** – The value of SoHR compliance state ADM element (section 10.3.2).
   - **extendedQuarantineState** – The value of the extended quarantine state, set in the ExtState field of the SoHR, as described in [TNC-IF-TNCCSPBSoH].

8. The Policy Engine stores the SoHR in the **SoHR** ADM element (section 10.3.2).

9. The Policy Engine passes the SoHR and the policy enforcement information to the RNAP Server or to the PEAP Server, depending on the transport protocol that is used:

   1. When PEAP (as defined in [MS-PEAP]) is the transport protocol, the PEAP Server retrieves the SoHR message using the **EapGetSoHRData** abstract interface specified in [MS-PEAP] section 3.3.7.5, and encapsulates the SoHR message in the EAP Extension Method "SoH Response TLV" as specified in [MS-PEAP] section 2.2.8.1.3. The resulting EAP message is encapsulated using the **EAP-Message** attribute as specified in [RFC3579]. The PEAP Server then sends the enforcement decisions as vendor-specific quarantine attributes that instruct the PEP regarding which network access to enforce as a part of the RADIUS Access-Accept or Access-Reject message.

   2. When the transport protocol is as defined in [MS-HCEP], [MS-DHCPN], or [MS-TSGU], the Policy Engine passes the SoHR received from SoH Server and policy configuration information to the RNAP Server using the **SendRadiusAccessAccept** abstract interface specified in [MS-RNAP] section 3.2.4.1. Interface parameters are sent as follows:

      - **SoHR** – The SoHR message received from the SoH Server.
      - **quarantineState** – The value of **quarantineState** received from the SoH Server.
      - **extendedQuarantineState** – The value of the **extendedQuarantineState** received from the SoH Server.
      - **ipv4Filter** – The value of **ipv4Filter** field of the **Policy Configuration** ADM element (section 10.3.2).
      - **ipv6Filter** – The value of **ipv6Filter** field of the Policy Configuration ADM element.
      - **quarantineSessionTimeout** – The value of **quarantineSessionTimeout** field of the **Policy Configuration** ADM element.
      - **quarantineGraceTime** – The value of **quarantineGraceTime** field of the **Policy Configuration** ADM element.
      - **ipv4RemediationServers** – If the value of the **Remediation required** field stored in the **Policy Configuration** ADM element is true, this parameter contains the value of **ipv4RemediationServers** field of the **Policy Configuration** ADM element. Otherwise, this parameter contains an empty array.

- **ipv6RemediationServers** – If the value of the **Remediation required** field stored in the **Policy Configuration** ADM element is true, this parameter contains the value of **ipv6RemediationServers** field of the **Policy Configuration** ADM element. Otherwise, this parameter contains an empty array.
- **dhcpQuarantineUserClass** – The value of **dhcpQuarantineUserClass** field of the **Policy Configuration** ADM element.
- **rdgDeviceRedirection** – The value of **rdgDeviceRedirection** field of the **Policy Configuration** ADM element.
- **afwZone** – The value of **afwZone** field of the **Policy Configuration** ADM element.
- **afwLevel** – The value of **afwLevel** field of the **Policy Configuration** ADM element.

10. The RNAP Server encapsulates SoHR messages into the MS-Quarantine-SOH VSA of the RADIUS Access-Accept or Access-Reject packet as specified in [MS-RNAP] section 2.2.1 and sends the response to the PEP using RADIUS [RFC2865] over the RNAP Channel.

10.5 Task Security

The security consideration for this task is that the PEP and the NAP health policy server must maintain a trust relationship.

For additional information about security considerations, see section 16, as well as the Security sections of the referenced protocol Technical Documents.

11 Enforce NAP Policy Task

This section describes the Enforce NAP Policy Task. This task is performed on the PEP. This task is expected to be used by the PEP to enforce network restrictions for a noncompliant NAP client. The PEP enforces network restrictions on different PEP channels by using a number of different protocols, including [MS-HCEP], [MS-DHCPN], [MS-TSGU], or [MS-PEAP].

Note This task uses the **PEP Channel Used** ADM element (section 4.1.1). All other common information defined in section 4 is not applicable to this task.

11.1 Task Overview

11.1.1 Task Purpose

The purpose of this task is to create a message response containing the network restrictions that have to be enforced on the NAP client based on the health evaluation. The exact method of enforcement depends on the specific PEP channel used (DHCP, HCEP, PEAP, TSG).

11.1.2 Task Applicability

This task is used whenever a NAP health evaluation occurs. After the PDP processes the contents of the SoH against the configured system health requirement policies, it creates and sends the SoHR back to the PEP indicating if the client is compliant and its level of network access or restriction.

NAP health evaluations can occur at the initial connection to the network or network resources, periodically, when network state changes, when an element of system health that is being monitored by SHAs running on the NAP client changes or when it is manually triggered by the user.

11.1.3 Task Use Cases

11.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Enforce NAP Policy Task are as follows:

NAP Enforcement Proxy: Refers to the generic component that represents a server endpoint capable of performing NAP enforcement. The interest of this actor in this task is that the task will process all received health evaluations for noncompliant clients.

Proxy SoHR Task: The main interest of the Proxy SoHR Task in this use case is that all received health evaluations will be processed according to the PEP channel used (DHCP, HCEP, PEAP, TSG).

11.1.3.2 Supporting Actors and Task Interests Summary

DHCPN Server: This protocol server is used to create a description of the network restrictions that have to be enforced on the NAP client when the DHCP Extensions for NAP Protocol [MS-DHCPN] is used as the PEP Channel. The use case employs this actor to produce a description of the network restrictions that have to be enforced on the client as a DHCPACK message.

HCEP HRA: This protocol server is used to create a description of the network restrictions that have to be enforced on the NAP client when the Health Certificate Enrollment Protocol [MS-HCEP] is used as the PEP Channel. The use case employs this actor to produce an HCEP response containing an SoHR and PKCS#7 message with an X.509 certificate to be sent to the client.

PEAP Server: This protocol server is used to create a description of the network restrictions that have to be enforced on the NAP client when the Protected Extensible Authentication Protocol [MS-

PEAP] is used as the PEP Channel. The use case employs this actor to produce a description of the network restrictions that have to be enforced on the client as a RADIUS Access-Accept message.

TSGU Server: This protocol server is used to create a description of the network restrictions that have to be enforced on the NAP client when the Terminal Services Gateway Server Protocol [MS-TSGU] is used as the PEP Channel. The use case employs this actor to generate an SoHR that will be sent back to the client by the Proxy SoHR Task (section 12) and to determine whether to grant access to the remote desktop.

11.1.3.3 Use Case Diagrams

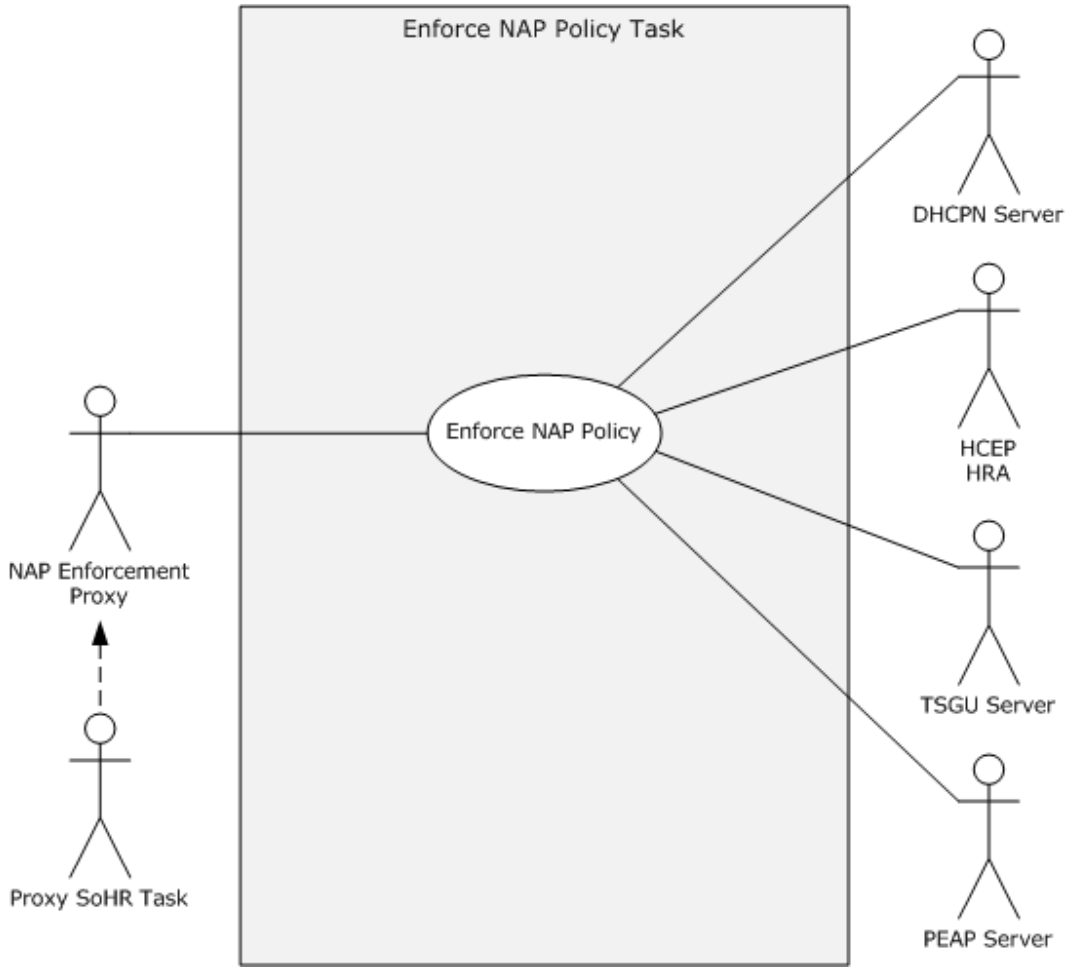

Figure 43: Enforce NAP Task use case diagram

11.1.3.4 Use Case: Enforce NAP Policy -- PEP Channel

Goal: To create a description of the network restrictions that have to be enforced on the NAP client based on the health evaluation of the client. The method of enforcement depends on the specific PEP channel being used.

Context of Use: This use case is employed when the client is noncompliant and network restrictions MUST be enforced on the NAP client.

Direct Actor: This role is performed by the NAP Enforcement Proxy. The NAP Enforcement Proxy refers to the generic component representing a server endpoint capable of performing NAP enforcement. The interest of this actor in this task is that the use case will process all received health evaluations for noncompliant clients.

Primary Actor: The primary actor is the Proxy SoHR Task. The main interest of this actor in this use case is that all received health evaluations will be processed according to the specific PEP channel used.

Supporting Actors: The supporting actors are as defined in section 11.1.3.2.

Stakeholders and Interests: The stakeholders are as defined in section 11.1.3.1.

Preconditions: The NAP Enforcement Proxy is configured to restrict the noncompliant clients.

Minimal Guarantees:

- The use case will process all received health evaluations for noncompliant clients.
- The received health evaluations will be processed according to the PEP channel used.

Success Guarantee: The NAP Enforcement Proxy generates a description of the enforcement restrictions that have to be applied on the client.

Trigger: This use case is triggered when the Proxy SoHR Task (section 12) identifies that an access restriction is required on the client.

Main Success Scenario:

1. The task is triggered when the Proxy SoHR Task (section 12) identifies that the SoHR message and enforcement decision received from the RNAP channel contain the decision to restrict client access and the task sets the SoHR abstract parameter to correspond to the noncompliant client and enforcement decision.
2. The NAP Enforcement Proxy uses the value of the **PEP Channel Used** ADM element to request the appropriate protocol server to create a response containing the network restrictions to be applied on the NAP client. Depending on the value of **PEP Channel Used**, there are some possible scenarios:
   - If TSG enforcement is being used, the TSGU Server rejects the remote desktop access request using [MS-TSGU].
   - If IPsec enforcement is being used, HCEP HRA obtains a health certificate with restricted access for the certificate request, as specified in [MS-HCEP], and creates an HCEP response.
   - If PEAP enforcement is being used, the PEAP Server builds a RADIUS Access-Accept message that contains RADIUS attributes to restrict the traffic of the NAP client. The PEAP Server creates a response, as specified in [MS-PEAP], with an encapsulated SoHR.
   - If DHCP enforcement is being used, the DHCPN Server builds a DHCPACK message containing a default gateway and subnet mask that restricts access to the network.
3. The NAP Enforcement Proxy returns the response message created in the previous step to the calling task, Proxy SoHR Task (section 12).

Extensions: None.

11.2 Task Context

This section describes the relationship between this task and its environment.

11.2.1 Task Environment

This task is accomplished by a PEP wherein an SoHR is received, containing the decision to restrict the NAP client access by using different PEP channels. The environment should meet the following requirement to support this task.

- **Requirement**: The NAP Enforcement Proxy together with the corresponding protocol server (DHCPN Server, HCEP HRA, PEAP Server, TSGU Server) are correctly configured.
  - **Reason for requirement**: Correct configuration of the NAP Enforcement Proxy and the corresponding protocol server is required to ensure the enforcement decision carries correct information to the client.
  - **Satisfying the requirement**: The NAP capabilities for an implementation of the underlying protocol server (DHCPN Server, HCEP HRA, PEAP Server, TSGU Server) are installed and configured.
  - **Verifying requirement is satisfied**:
    1. A network capture performed during the execution of the Proxy SoHR Task (section 12) shows that the NAP Enforcement Proxy sends the appropriate enforcement decisions to the NAP client.
    2. No errors related to configuration are logged by the NAP Enforcement Proxy.
  - **Consequences of not satisfying requirement**: The task is unable to provide an adequate enforcement decision.

Unless explicitly specified otherwise, the task implementation may assume its environment is properly configured and is not expected to verify that every requirement is satisfied. On the other hand, the task implementation should be able to gracefully handle errors which may be caused by environment misconfiguration or temporary dysfunction. The implementation should log these errors along with relevant information to allow troubleshooting.

11.2.2 Task Relationships

11.2.2.1 Black-Box Relationship Diagram

This task consists of providing network access (either restricted or full access) or resource access (either denied or granted) to the client based on the RADIUS Access-Accept message from the PDP. The following diagram illustrates the task to enable the enforcement function of the NAP System.

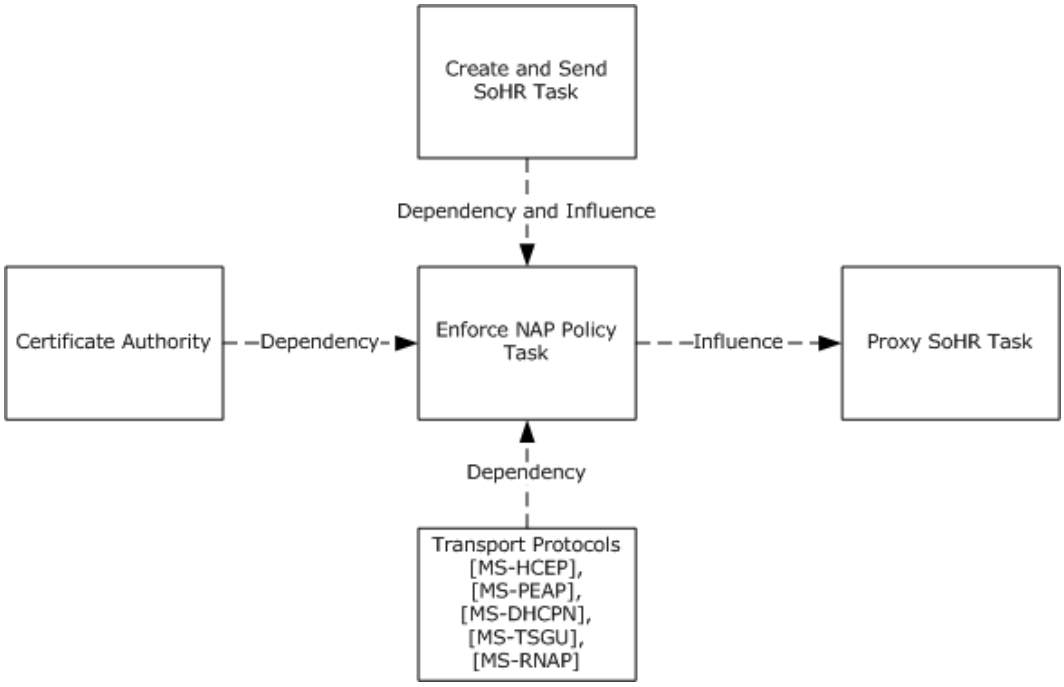

Figure 44: Enforce NAP Policy Task black-box relationships

11.2.2.1.1 Enforcement with the HTTP/S Channel

The IPsec and TSG enforcement methods use the HTTP/S channel.

IPsec enforcement in NAP consists of adding a Health registration authority (HRA) and NAP clients to an IPsec deployment. The HRA obtains X.509-based health certificates from a Windows-based certification authority (CA) on behalf of NAP clients when the PDP (NAP health policy server) has determined that the clients are compliant. NAP clients use health certificates for IPsec authentication when they initiate IPsec-protected communications with other compliant NAP clients on a network. If a NAP client does not have a health certificate, the IPsec peer authentication fails and the NAP client cannot communicate with a compliant IPsec peer.

TSG enforcement in NAP consists of a NAP-capable Terminal Services client initiating a connection to a NAP-enabled TSG. If the NAP client is authenticated, authorized, and is compliant, the TSG grants the connection. If the NAP client is not authenticated, not authorized, or is noncompliant, the TSG denies the connection.

11.2.2.1.2 Enforcement with the PEAP Channel

The 802.1X and VPN enforcement methods use the PEAP channel.

802.1X enforcement uses an access control list (ACL) or a virtual local area network (VLAN) identifier (ID) to restrict the access of the noncompliant NAP client. An ACL is a set of Internet Protocol version 4 (IPv4) or Internet Protocol version 6 (IPv6) message filters configured on the PEAP-based server or device. Multiple switch ports grouped to create a separate network are referred to as a VLAN.

Enforcement using an ACL

There are two approaches to enforce ACLs on the network access device:

1. The PEP is configured by creating the ACLs required for compliant and noncompliant machines. These ACLs restrict protocols, ports, and access to certain IP addresses only. The ACLs configured on the network access device are specified on the NPS server using the Filter-ID attribute for both compliant and noncompliant machines.

   When NPS receives a PEAP message, it checks the health state of the client. Based on the evaluation, the NPS fills the Filter-ID attribute and sends it to the PEAP, which in turn applies the appropriate ACL.

2. The ACLs are configured on the NPS and sent to the PEP using MS-Filter, as specified in [RFC2548] section 2.7.3, and MS-IPv6-Filter, as specified in [MS-RNAP] section 2.2.1.15. The PEP applies the ACL for the restricted network to the connection and traffic from noncompliant NAP clients MUST match the ACL filters.

Enforcement using a VLAN

Each VLAN is identified with a VLAN ID. When VLANs are used, the PEP applies the VLAN ID for the restricted network to the connection and traffic from noncompliant NAP clients does not leave the restricted network.

For VLAN enforcement, the NPS server is configured with a policy having a set of attributes with values that match the VLAN created on the PEP. The Tunnel-Type, Tunnel-Medium-Type, and Tunnel-Private-Group-ID RADIUS attributes are used for specifying dynamic VLAN ID assignment, as specified in [RFC3580]. The values for these attributes usually match the VLAN name or number created on the PEP. When a machine joins the network and meets the criteria of one of the policies, the NPS sends back this tunnel information to the switch to instruct the switch to add that machine to the proper VLAN. For more information about the network infrastructure requirement, see [MSFT-802.1XEnforceConfig].

11.2.2.1.3 Enforcement with the DHCP Channel

DHCP enforcement uses a limited access IPv4 address configuration and a set of host routes to restrict the access of a noncompliant NAP client. The noncompliant NAP client obtains an IPv4 address, a subnet mask of 256.256.256.255, and no default gateway. With this configuration, the noncompliant NAP client cannot send messages to other computers on its subnet or other subnets. The set of host routes correspond to the remediation server group that is configured on the PDP. With the host routes in its IPv4 routing table, the noncompliant NAP client can send messages to the remediation servers on the intranet (as described in [MS-DHCPN]).

11.2.2.2 Task Dependencies

The Enforce NAP Policy Task dependencies are the following:

- The Create and Send Task (section 10) to provide enforcement instruction for PEP to follow.
- Underlying communication protocols [MS-HCEP], [MS-DHCPN], [MS-TSGU], or [MS-PEAP] to perform the access restriction as specified in section 11.4.5.
- A Certificate Authority, in case of IPsec enforcement, is used to sign the health certificates created for healthy client computers.

11.2.2.3 Task Influences

The Enforce NAP Policy Task influences the Proxy SoHR Task. The result of this task is enforcement methods performed on the PEP Channels. The PEP channels then send responses to the NAP client using the Proxy SoHR Task.

11.2.3 Task Assumptions and Preconditions

To accomplish this task, the PEP has the following preconditions and assumptions:

- The underlying network infrastructures, such as the RADIUS, DHCP, PEAP, HTTP/S channel name and address resolution, and routing services, are configured correctly.
- The NAP health policy server is configured correctly by the server administrator.
- The PEP is trustable and functioning correctly.
- The NAP client is enabled and correctly configured by the client administrator.
- The PEP is configured and is reachable for the clients in the restricted network and intranet.

11.2.4 Task Versioning and Capability Negotiation

The system does not define any versioning or capability negotiation beyond those described in the specifications of the protocols supported by the system.

11.3 Task Architecture

11.3.1 Task Architectural Constraints

There can be more than one instance of the Enforce NAP Policy Task on each PEP. These task instances initialize themselves each time they start and run independently. Different instances of this task on different PEPs also run independently.

11.3.2 Task Abstract Data Model

This section describes the state established, used, and maintained by processing rules of this task. State may be volatile or persisted and may pertain to one, some, or all instances of the task. The task's state consists of the values of the named data elements (also called state variables) presented in this section. The overall organization of the data elements, with their names, is the Abstract Data Model. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Data Model, such association can be achieved in other ways. Implementations may depart from this model so long as their external behavior remains consistent with that described in this document.

network access state for known clients: An enumeration {NO_ACCESS, RESTRICTED_ACCESS, UNRESTRICTED_ACCESS} that specifies the network access enforcement state per known client per protocol, where a known client is a client that sent an SoH and the system generated an SoHR in return.

PEP Channel Used: A common ADM element specified in section 4.1.1.

11.3.2.1 Task Abstract Interface

None.

11.3.3 Task Abstract Parameters

This section describes data passed to an instance of this task at the time it is invoked or triggered. The parameters consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Parameter. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Parameters, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

SoHR: SOHR message and enforcement decision.

11.3.4 Task Abstract Results

This section describes data returned by an instance of this task to its caller. The results consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Result. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Results, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

This task returns a message containing the network restrictions that have to be enforced on the client. Proxy SoHR Task (section 12) sends this message to the NAP client. The exact form of the generated message depends on the underlying protocol (DHCPN Server generates a DHCPACK message, PEAP Server produces an SoHResponse_TLV message, HCEP HRA creates an HCEP response, TSGU Server builds a TSG_PACKET).

11.3.5 White-Box Relationships

The PDP creates a RADIUS Access-Accept message that contains the SoHR and RADIUS attributes, such as IP filters, or a VLAN ID. PEAP and DHCP servers use RADIUS attributes in the RADIUS Access-Accept message to enforce the restricted access of the noncompliant NAP client.

The following diagram shows the interactions between the NAP Enforcement Proxy and the protocol servers.

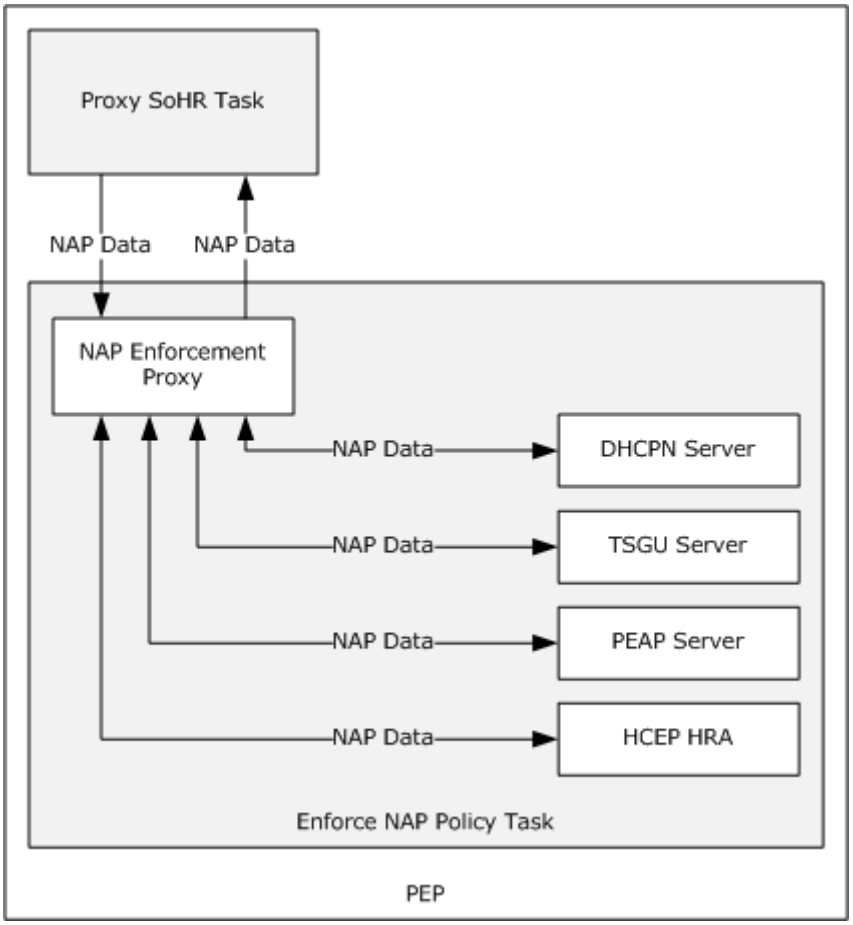

Figure 45: Enforce NAP Policy Task white-box relationship diagram

11.3.5.1 HTTP/S Channel

HTTP/S channel consists of two specific enforcements IPsec enforcement and TSG enforcement.

11.3.5.1.1 TSG Enforcement

To access a remote desktop, a **NAP client** using TSG enforcement starts up on the network and uses the following process (see [MS-TSGU] for additional details):

1. The NAP client obtains network access and an IP address configuration.
2. The TSGU client on the NAP client sends its credentials and its SoH to the PEP (the TSG server).
3. The TSGU server passes the SoH to the PDP in a RADIUS Access-Request message.
4. The policy engine on the PDP receives the RADIUS Access-Request message, extracts the SoH, and passes it to the NAP Validator component on the PDP.
5. The NAP Validator component passes the individual health statements within the SoH to the appropriate system health validators (SHVs).

6. The SHVs analyze the contents of their health statements and return health responses to the NAP Validator.

7. The NAP Validator processes the health responses and other RADIUS attributes in the Access-Request message against the health requirement policies and creates the SoHR, containing the individual health responses and a compliance indicator.

8. The policy engine sends a RADIUS Access-Accept message with the SoHR to the TSGU server.

9. The TSGU server sends the SoHR back to the TSGU client on the NAP client. If the NAP client is compliant, the TSGU server also grants access to the remote desktop.

10. If the NAP client is not compliant, the TSGU server rejects the remote desktop access request.

11.3.5.1.2 IPsec Enforcement

To obtain a health certificate and to become a member of the secure network, a NAP client using IPsec enforcement starts up on the network and sends an SoH message to the PEP.

The HCEP HRA is based on the NAP health policy server (PDP) that utilizes two components: an HRA and a Policy Engine. The Policy Engine has the capability to proxy SoH messages received in an HCEP request to a second PDP using a RADIUS Access-Request message (see section 3.3.3). As shown in the figure below, the RADIUS access response is received by the Policy Engine component in the PDP.

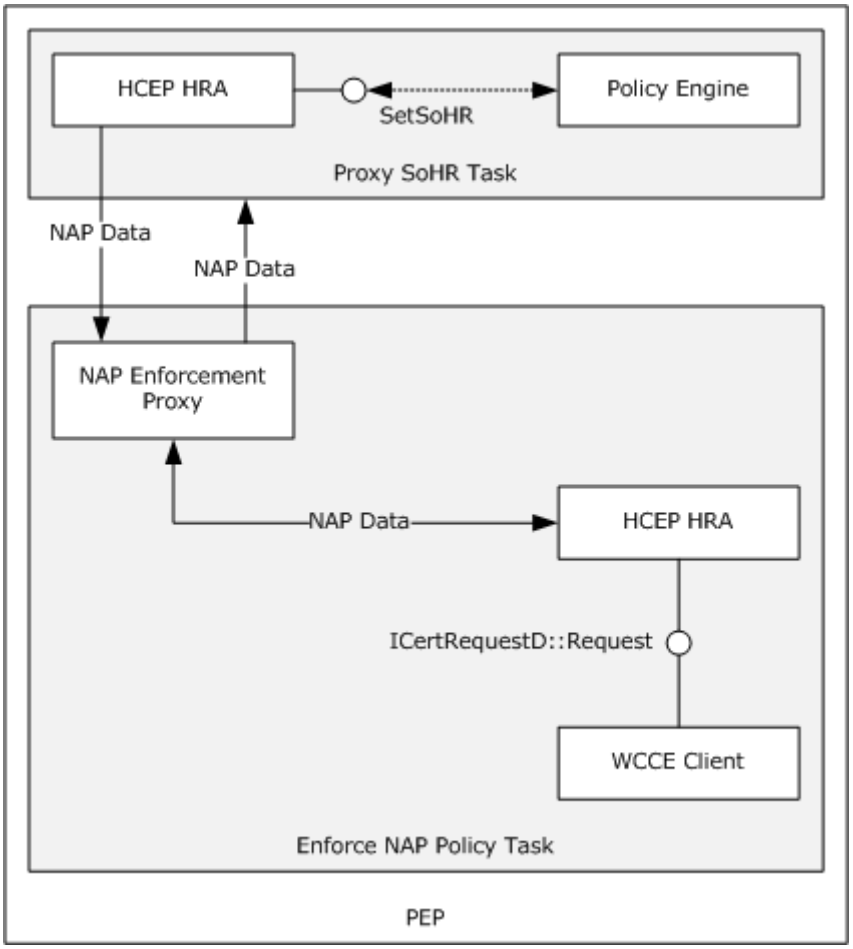

Figure 46: IPsec enforcement white-box relationship diagram

1. The PEP receives an SoHR message and enforcement decision.

2. Based on the MS-Quarantine-State attribute in the SoHR message as specified in [TNC-IF-TNCCSPBSoH], the Policy Engine sets the **network access state for known clients** ADM element and passes the SoHR to the HRA as a parameter to the HRA **SetSoHR** abstract interface.

3. If the value of **network access state for known clients** equals UNRESTRICTED_ACCESS, the HRA obtains a health certificate for the certificate request in the corresponding HCEP request. This may be accomplished using the Windows Client Certificate Enrollment Protocol [MS-WCCE] as follows:

    1. Create a certificate request as specified in [MS-HCEP] section 3.2.5.4.

    2. Send the request to the CA using the interface ICertRequestD::Request as specified in [MS-WCCE] section 3.1.1.4.3.1.1. For information about error handling, see [MS-HCEP] section 3.2.8.

4. The HRA creates an HCEP response as specified in [MS-HCEP] section 2.2.2. In this case, the payload of the response contains an SoHR as specified in [TNC-IF-TNCCSPBSoH] and a PKCS #7 message as specified in [RFC2315] with an X.509 certificate as specified in [RFC3280]. For more

information about the certificate, see [X509]. The SoHR MUST then be encoded using base64 as specified in [RFC3548], and used to set the HCEP-SoHR in the HCEP response.

11.3.5.2 PEAP Channel

The following process triggers the enforcement for the PEAP channel:

If the PEAP connection is authenticated and authorized, the PDP sends a RADIUS Access-Accept message to the PEP as specified in [MS-PEAP] section 3.3.5.4.6.

- If the NAP client is noncompliant, the RADIUS Access-Accept message also contains RADIUS attributes to restrict the traffic of the NAP client. For 802.1X enforcement, the MS-Filter ([RFC2548] section 2.7.3) and MS-IPv6-Filter ([MS-RNAP] section 2.2.1.15) VSAs are used for specifying an ACL. Alternatively, the RADIUS attributes Tunnel-Type, Tunnel-Medium-Type, and Tunnel-Private-Group-ID are used for specifying dynamic VLAN ID assignment, as specified in [RFC3580]. For VPN enforcement, a set of IPv4 or IPv6 packet filters is provided by the MS-Filter and MS-IPv6-Filter VSAs. ACL and VLAN ID traffic restrictions are enforced by the PEP, that is, the PEAP Server (see section 11.2.2.1.2).
- If the NAP client is compliant, the RADIUS Access-Accept message contains the appropriate RADIUS attributes to allow full access to the intranet.

  PEP creates a response as specified in [MS-PEAP] section 3.3.5.4 with the SoHR encapsulated as specified in [MS-PEAP] section 3.3.5.4.6.

11.3.5.3 DHCP Channel

The following process triggers the enforcement for the DHCP channel:

The Policy Engine component sends an Access-Accept message containing the SoHR to the DHCP server.

- If the NAP client is noncompliant, the policy engine sends a RADIUS Access-Accept message with the MS-IPv4-Remediation-Servers VSA, as specified in [MS-RNAP] section 11.3.5.3, containing the IPv4 addresses of the remediation server group to restrict the traffic of the DHCP client. After receiving the RADIUS Access-Accept message, the PEP (that is, the DHCP server) enforces client access to remediation servers as described in section 11.2.2.1.3 and [MS-DHCPN] section 3.2.5.2.1. After the DHCP configuration completes, the NAP client will have restricted network access. As an example when the client is not compliant with the health policies, the DHCP server includes the following in the DHCPAck message:
  - A default gateway (DHCP option 3, the router option, as specified in [RFC2132] section 3.5) of 0.0.0.0.
  - A subnet mask (DHCP option 1, as specified in [RFC2132] section 3.3) of 255.255.255.255.
  - A Classless Static Route option (as specified in [MS-DHCPN]) that contains the static routes to the remediation servers.
- If the NAP client is compliant, the RADIUS Access-Accept message does not contain the additional packet filters for the remediation server group. After the DHCP configuration completes, the NAP client will have full access to the intranet.

11.3.6 Task Events

11.3.6.1 Task Timers

The system does not define any task timers beyond those described in the specifications of the protocols supported by the system.

11.3.6.2 Task Non-Timer Events

The system does not define any task non-timer events beyond those described in the specifications of the protocols supported by the system.

11.3.7 Task Architecture and Communication

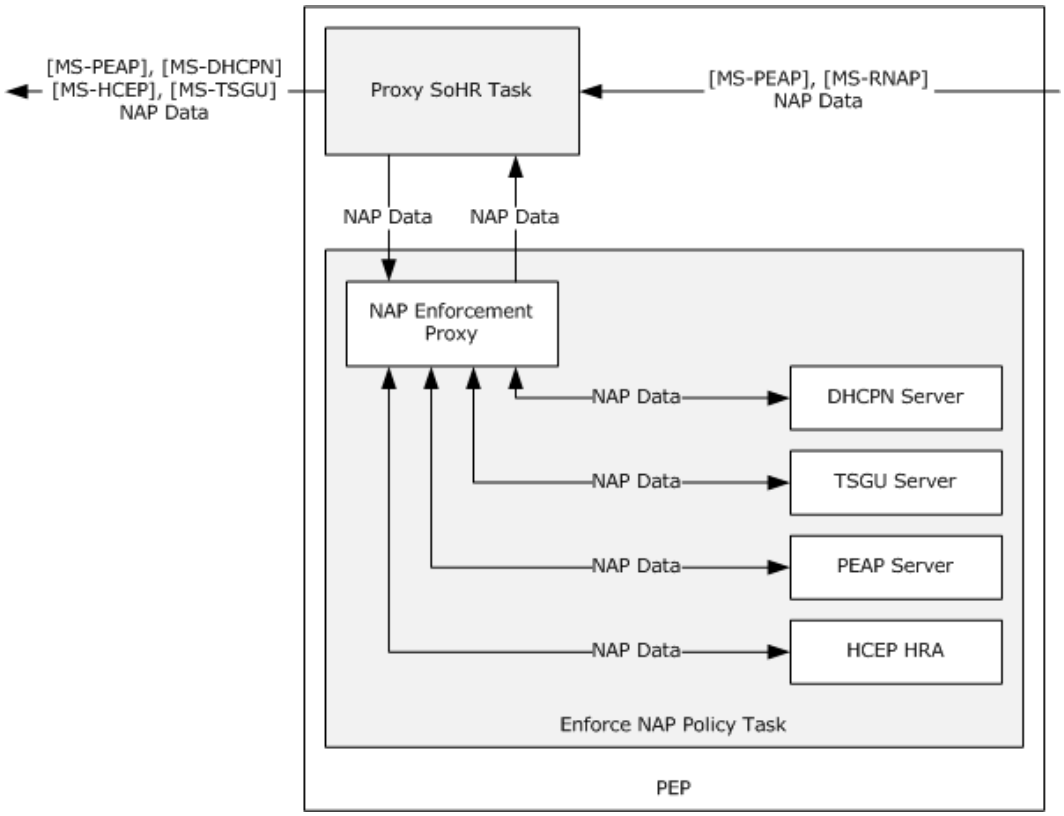

Figure 47: Enforce NAP Policy Task architecture and communication

The preceding diagram shows the architectural details and the interaction between the Enforce NAP Policy Task and the invoking Proxy SoHR Task (section 12).

The Enforce NAP Policy Task is triggered by the Proxy SoHR Task when it receives the evaluation of health on the PDP. The PDP (SHV and the policy engine) evaluates the health of the NAP client and creates the RADIUS Access-Accept message with the SoHR and RADIUS attributes as described in [MS-RNAP]. The PDP then transmits the RADIUS Access-Accept message to the PEP. The PEP interprets the RADIUS attributes in the Access-Accept message for enforcement of the NAP client and builds a response that is passed to the Proxy SoHR Task. The Proxy SoHR Task then sends the message to the NAP client using one of the underlying protocols.

11.3.8 Task Processing Rules

The following describes the operational flow of the Enforce NAP Policy Task:

1. This task is initiated when the Proxy SoHR Task (section 12) sets the SoHR abstract parameter to correspond to a noncompliant client.

2. The NAP Enforcement Proxy uses the value of the **PEP Channel Used** ADM element to request the appropriate protocol server to create a response containing the network restrictions to be applied on the NAP client. The following steps describe the processing rules based on the value of **PEP Channel Used**:

    1. If **PEP Channel Used** is DHCP:

        1. The DHCPN server generates a DHCPACK Message containing the NAP-SoH attribute as specified in [MS-DHCPN] section 2.2.1.1.

        2. As described in [MS-DHCPN] section 3.2.5.2.1, the DHCPN Server fixes the default network configuration options corresponding to the NAP user and overrides three option values:

            - The Router option (DHCP option 3, as specified in [RFC2132] section 3.3) is set to the value 0.0.0.0.

            - The Subnet Mask option (DHCP option 1, as specified in [RFC2132] section 3.3) is set to the value 255.255.255.255.

            - The Microsoft Classless Static Route option (as specified in [MS-DHCPE] section 2.2.8) is configured with static routes to the IPv4 addresses of the NAP remediation servers.

            - If the DHCP client is being quarantined, the DHCPN server includes the NAP-Mask option (as specified in [MS-DHCPN] section 2.2.1.2), and the IPv6 addresses of the NAP remediation servers in the NAP-IPv6 option if configured to do so on the server (as specified in [MS-DHCPN] section 2.2.1.4).

        3. As stated in [MS-DHCPN] section 3.2.5.2.4, there are no differences in the way which the DHCPN server determines enforcement restrictions during New Lease Acquisition or Lease Renewal.

    2. If **PEP Channel Used** is PEAP:

        1. The PEAP server prepares an SoH Response TLV message ([MS-PEAP] section 2.2.8.1.3) as specified in [MS-PEAP] section 3.3.5.4.6. The SoH Response TLV is embedded into a RADIUS Access-Accept message containing attributes to restrict the traffic of the NAP client.

        2. Depending on the type of enforcement used, the PEAP server sets the following attributes:

            - For 802.1X enforcement, the PEAP server sends the MS-Filter ([RFC2548] section 2.7.3) and MS-IPv6-Filter ([MS-RNAP] section 2.2.1.5) VSAs to specify an ACL. Alternatively, the Tunnel-Type, Tunnel-Medium-Type, and Tunnel-Private-Group-ID RADIUS attributes are used to specify dynamic VLAN ID assignment as specified in [RFC3580].

            - For VPN enforcement, a set of IPv4 or IPv6 packet filters are provided by the MS-Filter and MS-IPv6-Filter VSAs.

    3. If **PEP Channel Used** is HCEP:

        1. The HCEP HRA creates an HCEP response as described in [MS-HCEP] section 3.2.5.3.

2. The HCEP HRA creates the HCEP response header using the HCEP-Correlation-Id of the received request.

3. The HCEP HRA encodes the **SoHR** abstract parameter using Base64 (as specified in [RFC3548]) and stores this value in the HCEP-SoHR field of the HCEP response header.

4. The HCEP HRA sets the HCEP-AFW-Zone and HCEP-AFW-Protection-Level fields of the response with the values received from the Policy Server.

Note The HCEP HRA can request a certificate from the CA for a noncompliant client as described in [MS-HCEP] section 3.2.5.3.

4. If **PEP Channel Used** is TSG:

1. The TSGU server builds a TSG_PACKET ([MS-TSGU] section 2.2.9.2) as described in [MS-TSGU] section 3.2.6.1.2.

2. The TSGU server signs the SoHR using an SHA-1 hash, encodes the SoHR with the TSGU server certificate, and appends the signed and encoded SoHR to the responseData field of the TSD_PACKET_RESPONSE field.

Note For a noncompliant client, the TSGU server returns the error code E_PROXY_QUARANTINE_ACCESSDENIED to indicate that the connection will be rejected.

3. The NAP Enforcement Proxy returns the response created in the previous step to the calling task which will communicate the decision to the NAP Client.

If an error is raised at any stage of the Enforce NAP Policy Task, the task fails.

11.3.9 Task Failure Scenarios

11.3.9.1 NAP Client and PEP Communication

These failures can be caused by:

- Improper configuration of the NAP client or PEP.
- Network connectivity issues in which the NAP client cannot communicate with the PEP.

If the NAP client cannot communicate with the PEP, the client may not have access to network resources. For example, the NAP client will not have any network access if it cannot communicate with an 802.1X switch but will have network access if the NAP client receives a response from a non-NAP enabled DHCP server. The system may recover from certain types of failures (for example, the DHCP client can attempt to connect to secondary DHCP server if there is no response from the primary server) and cannot recover from various other failures (for example, if the NAP client cannot communicate with an 802.1X switch or VPN server then the NAP System cannot recover from this failure). The failures can be detected by the timers on the NAP client.

11.3.9.2 PEP and PDP communication

PEP and PDP communication failure can be caused when the IIS worker process on the HRA requires a reset (IPsec enforcement). This failure can be detected by the events on the PDP. The NAP System does not recover from this error.

11.4 Task Details

This section contains the details that complete the descriptions in earlier sections of the document. These are needed to understand and implement this task.

11.4.1 Task Precondition Details

For the complete list of task preconditions and assumptions, see section 11.2.3. Details for some of the preconditions are as follows:

- All deployed PEPs must be configured correctly.
- IPFilters are properly configured on NAP health policy servers to quarantine noncompliant client users.
- Remediation servers are properly configured on NAP health policy servers for noncompliant client users to access remediation services.
- VLAN ID are properly configured on NAP health policy servers to quarantine noncompliant client users.
- If the PEP is an HRA server, this HRA server and its related CA have to be functioning so that the HRA server can obtain health certificates depending on the enforcement decisions.
- If the PEP is a DHCP server, it must recognize the RNAP protocol so that it can assign IP addresses correctly according to the enforcement decisions.
- If the PEP is a TSGU server, it must recognize the RADIUS VSAs [MS-RNAP] so that it can correctly accept or reject the **client user's** Terminal Service requests according to the enforcement decisions.
- If the PEP is a VPN server, it must recognize the RNAP protocol so that it can correctly accept or reject the client user's VPN connection requests according to the enforcement decisions.
- If the PEP is an 802.1X server, it must recognize the RNAP protocol so that it can accept or reject the client user's connection request correctly following the enforcement decisions.

11.4.2 Task Initialization of External Entities

None.

11.4.3 Task Event Details

11.4.3.1 Task Timer Details

This task does not impose any additional timers. Timers are related to the underlying transports and they are described in [MS-DHCPN], [MS-PEAP], and [MS-TSGU].

11.4.3.2 Task Non-Timer Event Details

This task does not impose any additional non-timer events. Non-timer events are related to the underlying transports and they are described in [MS-DHCPN], [MS-PEAP], and [MS-TSGU].

11.4.4 Task Architectural Details

This section illustrates an example of a PEP enforcing network restrictions on a client.

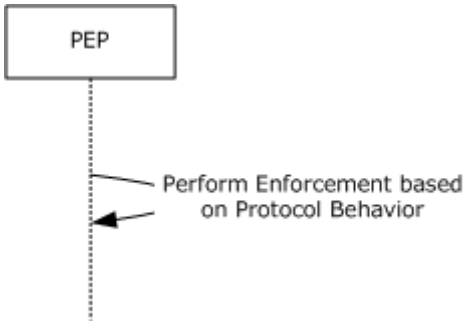

Figure 48: Sequence diagram for the main success scenario of the Enforce NAP Policy Task

The PEP performs network enforcement based on the protocol behavior as specified in sections 11.3.5.1, 11.3.5.2, and 11.3.5.3.

11.4.5 Task Processing Rule Details

The following describes the operational details of the Enforce NAP Policy Task:

1. The Proxy SoHR Task invokes the task and sets the abstract parameter SoHR corresponding to a noncompliant client.
2. The PEP identifies that the enforcement decision that was received with an SoHR message, contains the decision to restrict client access.
3. The PEP updates the **network access state for known clients** ADM element according to the value of MS-Quarantine-State of the received enforcement decision.
4. The NAP Enforcement Proxy uses the value of the **PEP Channel Used** ADM element to request the appropriate protocol server to create a response containing the network restrictions to be applied on the NAP client. The following steps describe the processing rules based on the value of **PEP Channel Used**:
    1. If **PEP Channel Used** is DHCP:
        1. The DHCPN server generates a DHCPACK Message containing the NAP-SoH attribute as specified in [MS-DHCPN] section 2.2.1.1.
        2. As described in [MS-DHCPN] section 3.2.5.2.1, the DHCPN Server fixes the default network configuration options corresponding to the NAP user and overrides three option values:
            - The Router option (DHCP option 3, as specified in [RFC2132] section 3.3) is set to the value 0.0.0.0.
            - The Subnet Mask option (DHCP option 1, as specified in [RFC2132] section 3.3) is set to the value 255.255.255.255.
            - The Microsoft Classless Static Route option (as specified in [MS-DHCPE] section 2.2.8) is configured with static routes to the IPv4 addresses of the NAP remediation servers.
            - If the DHCP client is being quarantined, the DHCPN server includes the NAP-Mask option (as specified in [MS-DHCPN] section 2.2.1.2), and the IPv6 addresses of the NAP remediation servers in the NAP-IPv6 option if configured to do so on the server (as specified in [MS-DHCPN] section 2.2.1.4).

3. The DPACK message is embedded into a Radius Access-Accept message containing attributes to restrict the traffic of the NAP client. For example:

   - The MS-Quarantine-IPFilter ([MS-RNAP] section 3.3.5.2.1) and MS-IPv6-Filter ([MS-RNAP] section 3.3.5.2.8) attributes can be set to indicate the accessible servers.
   - The MS-IPv4-Remediation-Servers ([MS-RNAP] section 3.3.5.2.9) and MS-IPv6-Remediation-Servers ([MS-RNAP] section 3.3.5.2.10) attributes can be set to indicate the accessible servers for the client when remediation requires data from the network. For example, Windows Server Update Services.

   The values assigned to these VSAs are part of the enforcement decision that this task receives as an abstract parameter. These values are fixed by the Create and Send SoHR Task (section 10.4.5). For example, usage of the **Policy Configuration** ADM element settings **ipv4RemediationServers** and **ipv6RemediationServers** as specified in section 10.3.2.

2. If **PEP Channel Used** is PEAP:

   1. The PEAP server prepares an SoH Response TLV message ([MS-PEAP] section 2.2.8.1.3) as specified in [MS-PEAP] section 3.3.5.4.6. The SoH Response TLV is embedded into a RADIUS Access-Accept message containing attributes to restrict the traffic of the NAP client.
   2. Depending on the type of enforcement used, the PEAP server sets the following attributes:

      - For 802.1X enforcement, the PEAP server sends the MS-Filter ([RFC2548] section 2.7.3) and MS-IPv6-Filter ([MS-RNAP] section 2.2.1.5) VSAs to specify an ACL. Alternatively, the Tunnel-Type, Tunnel-Medium-Type, and Tunnel-Private-Group-ID RADIUS attributes are used to specify dynamic VLAN ID assignment as specified in [RFC3580]. The following tunnel attributes are used:
        - Tunnel-Type=VLAN (13)
        - Tunnel-Medium-Type=802
        - Tunnel-Private-Group-ID=VLANID, where VLANID is 12 bits and contains a value between 1 and 4094 inclusive. The VLANID value is obtained from the enforcement decision received as an abstract parameter to this task.

        Note This value is assigned by the NPS in the Create and Send SoHR Task (section 10) when composing the corresponding RADIUS Access-Accept message as described in section 10.4.5.

      - For VPN enforcement, a set of IPv4 or IPv6 packet filters are provided by the MS-Filter and MS-IPv6-Filter VSAs.

3. If **PEP Channel Used** is HCEP:

   1. The HCEP HRA creates an HCEP response as described in [MS-HCEP] section 3.2.5.3.
   2. The HCEP HRA creates the HCEP response header using the HCEP-Correlation-Id of the received request.
   3. The HCEP HRA encodes the **SoHR** abstract parameter using Base64 (as specified in [RFC3548]) and stores this value in the **HCEP-SoHR** field of the HCEP response header.

4. The HCEP HRA sets the **HCEP-AFW-Zone** and **HCEP-AFW-Protection-Level** fields of the response with the values received from the Policy Server.

   Note The HCEP HRA can request a certificate from the CA for a noncompliant client as described in [MS-HCEP] section 3.2.5.3.

4. If **PEP Channel Used** is TSG:

   1. The TSGU server builds a TSG_PACKET ([MS-TSGU] section 2.2.9.2) as described in [MS-TSGU] section 3.2.6.1.2.

   2. The TSGU server signs the SoHR using an SHA-1 hash, encodes the SoHR with the TSGU server certificate, and appends the signed and encoded SoHR to the responseData field of the TSD_PACKET_RESPONSE field.

   Note For a noncompliant client, the TSGU server returns the error code E_PROXY_QUARANTINE_ACCESSDENIED to indicate that the connection will be rejected.

5. The NAP Enforcement Proxy returns the response created in the previous step to the calling task which communicates the decision to the NAP client.

11.5 Task Security

The security consideration for this task is that when HCEP is transported over TLS, an X.509 certificate is required in order to use SSL as specified in [MS-TLSP]. For additional information about security considerations, see section 16, as well as the Security sections of the referenced protocol Technical Documents.

12 Proxy SoHR Task

This section describes the task of receiving encapsulated SoHR messages from the NAP health policy server by the NAP Enforcement Proxy (PEP) and proxying them to the NAP EC. The encapsulated SoHR message is received by the PEP from the RNAP client [MS-RNAP] or from the EAP supporting RADIUS client [RFC3579].

Note This task uses the **PEP Channel Used** ADM element (section 4.1.1). All other common information defined in section 4 is not applicable to this task.

12.1 Task Overview

12.1.1 Task Purpose

The purpose of this task is to ensure the following:

- SoHR messages are correctly retrieved by the NAP Enforcement Proxy after the SoHR messages have been constructed and sent by the Create and Send SoHR Task (section 10).
- SoHR messages are correctly encapsulated and sent from the NAP Enforcement Proxy to the NAP EC.

12.1.2 Task Applicability

This task is used when an enforcement decision has been made and an SoHR message has been created on the NAP health policy server and has been sent to the NAP Enforcement Proxy. This task is not applicable if the NAP System is not deployed.

12.1.3 Task Use Cases

12.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Proxy SoHR Task are as follows:

RNAP client: A client that uses Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] to transport the SoH/SoHR messages between the NAP ES and the NAP health policy server. The interest of this actor in this task is that the use case will always process the received RNAP messages.

EAP supporting RADIUS client: A client that uses RADIUS [RFC2865] and RADIUS support for EAP [RFC3579] to transport PEAP messages that contain the SoH/SoHR messages between the NAP ES and the NAP health policy server. The interest of this actor in this task is that the use case will always process the received RADIUS messages.

NAP Enforcement Proxy: Used to proxy SoHR messages to the NAP EC. The interest of this actor in this task is that it will attempt to send any SoHR that it receives via the Health Certificate Enrollment Protocol [MS-HCEP], the DHCP Extensions for NAP [MS-DHCPN], the Terminal Services Gateway Server Protocol [MS-TSGU], or the Protected Extensible Authentication Protocol [MS-PEAP].

12.1.3.2 Supporting Actors and Task Interests Summary

HCEP HRA: This protocol server is used to receive Health Certificate Enrollment Protocol [MS-HCEP] messages from an HCEP client on the NAP EC computer. It acts in the role of an enforcement server (NAP ES) in this use case when HCEP is used. When HCEP enforcement is used, the Proxy SoHR Task uses the HCEP HRA to send the SoHR from the NAP Enforcement Proxy to the NAP EC.

DHCPN Server: This protocol server is used to receive DHCP Extensions for NAP [MS-DHCPN] messages from a DHCPN client on the NAP EC computer. It acts in the role of an enforcement server (NAP ES) in this use case when DHCPN is used. When DHCP enforcement is used, the Proxy SoHR Task uses the DHCPN Server to send the SoHR from the NAP Enforcement Proxy to the NAP EC.

TSGU Server: This protocol server is used to receive Terminal Services Gateway Server Protocol [MS-TSGU] messages from a TSGU client on the NAP EC computer. It acts in the role of an enforcement server (NAP ES) in this use case when TSGU is used. When TSGU enforcement is used, the Proxy SoHR Task uses the TSGU Server to send the SoHR from the NAP Enforcement Proxy to the NAP EC.

PEAP Pass-Through Server: This protocol server is used to receive Protected Extensible Authentication Protocol [MS-PEAP] messages from a PEAP peer on the NAP EC computer. It acts in the role of an enforcement server (NAP ES) in this use case when PEAP is used. When PEAP enforcement is used, the Proxy SoHR Task uses the PEAP Pass-through Server to send the SoHR encapsulated in PEAP message from the NAP Enforcement Proxy to the NAP EC.

Enforce NAP Policy Task: The purpose of this task is to create a message response containing the network restrictions that have to be enforced on the NAP client based on the health evaluation. The Proxy SoHR Task uses the task to update the message response per the enforcement decisions.

12.1.3.3 Use Case Diagrams

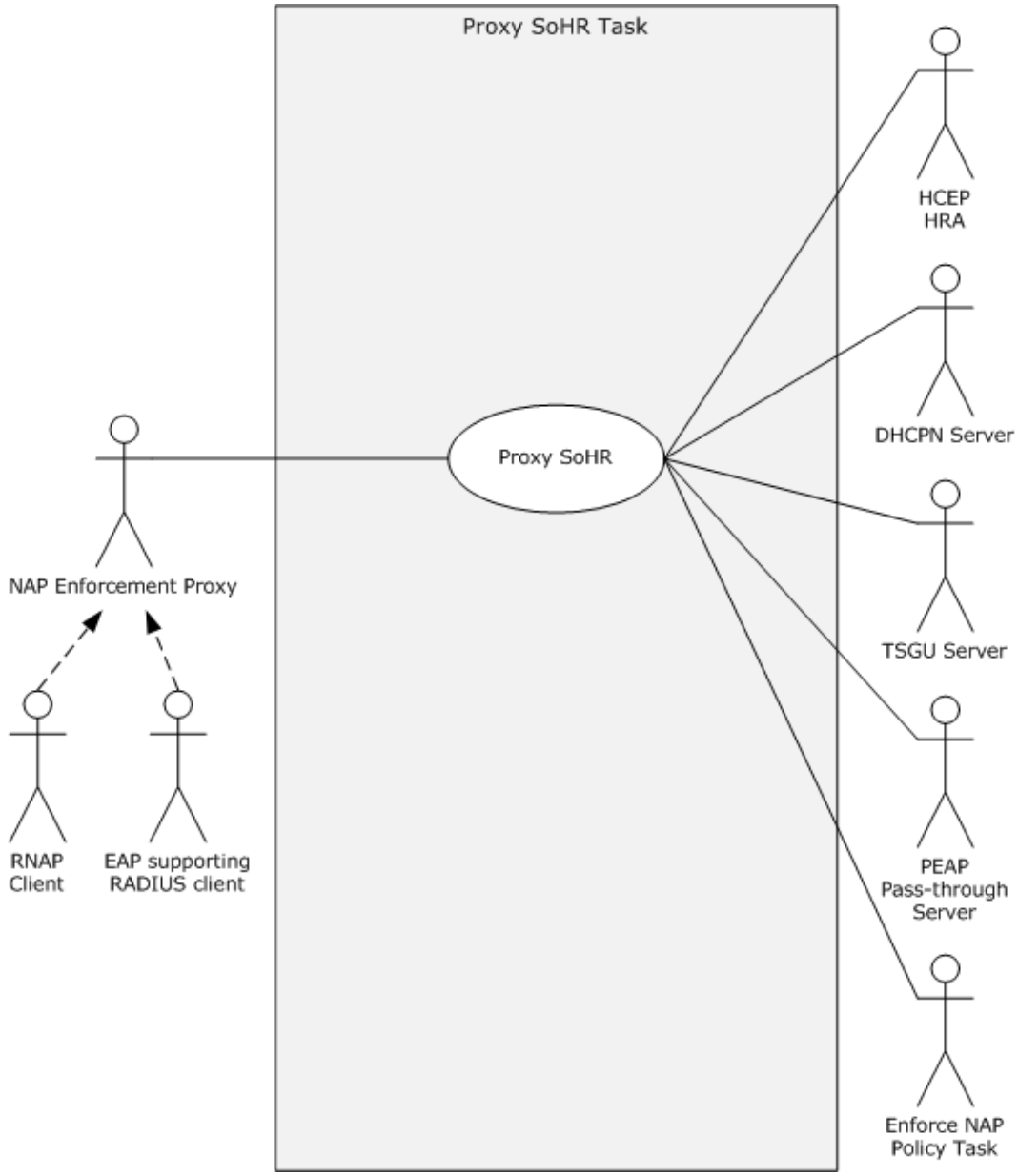

Figure 49: Proxy SoHR Task use case diagram

12.1.3.4 Use Case: Proxy SoHR -- NAP Enforcement Proxy

Goal: To proxy the SoHR message [TNC-IF-TNCCSPBSoH] from the NAP Enforcement Proxy to the NAP EC.

Context of Use: This use case is used when the SoHR message is passed by the RNAP client or an SoHR encapsulated in a PEAP message is passed by a RADIUS message which encapsulates SoHR messages.

Direct Actor: This role is performed by the NAP Enforcement Proxy. The interest of this actor in this task is the ability to integrally proxy SoHR messages via the HCEP HRA, DHCPN Server, TSGU Server, or PEAP Pass-through Server.

Primary Actor: This role is performed by the RNAP client and the EAP supporting RADIUS client. The interest of these actors in this task is that the use case will always process their received messages.

Supporting Actors: The supporting actors are the HCEP HRA, DHCPN Server, TSGU Server, PEAP Pass-through Server, and the Enforce NAP Policy Task (section 11).

Stakeholders and Interests: The stakeholders are defined as follows:

- Receive SoHR Task: The purpose of this task is to receive an SoH message. The main interest of the task in this use case is to receive SoHR messages only via the Health Certificate Enrollment Protocol [MS-HCEP], the DHCP Extensions for NAP [MS-DHCPN], the Terminal Services Gateway Server Protocol [MS-TSGU], or the Protected Extensible Authentication Protocol [MS-PEAP].

Preconditions: The Create and Send SoHR Task (section 10) was completed successfully, and the RNAP client or the EAP supporting RADIUS client received a message that encapsulates SoHR from the NAP health policy server.

- The NAP health policy server components on the server are deployed and configured correctly by the server administrator.
- The RNAP Channel and PEP channel are functional.

Minimal Guarantees:

- The use case will always process the received messages.
- The use case will always attempt to send any SoHR that it receives via the Health Certificate Enrollment Protocol [MS-HCEP], the DHCP Extensions for NAP [MS-DHCPN], the Terminal Services Gateway Server Protocol [MS-TSGU], or the Protected Extensible Authentication Protocol [MS-PEAP].

Success Guarantee: All the Minimal Guarantees are satisfied. Additionally, the payload that contains an SoHR message is sent to the NAP EC successfully.

Trigger: The RNAP client or the EAP supporting RADIUS client triggers this event when a RADIUS response arrives with a payload that contains an SoHR message.

Main Success Scenario:

1. In the case of PEAP enforcement, the SoHR encapsulated in the EAP message and the enforcement decision are passed from the PEAP Pass-through Server to the NAP Enforcement Proxy.
2. In the case of HCEP, DHCPN, or TSGU enforcement, the SoHR and the enforcement decision are passed by the RNAP client.
3. Depending on the value of the **PEP Channel Used** ADM element (section 12.3.2), the NAP Enforcement Proxy passes the SoHR buffer to the appropriate NAP ES, which is either an HCEP HRA, DHCPN Server, TSGU Server, or PEAP Pass-through Server.
4. If quarantine is required, the HCEP HRA, DHCPN Server, TSGU Server, or PEAP Pass-through Server successfully performs the Enforce NAP Policy Task (section 11).

5. The HCEP HRA, DHCPN Server, TSGU Server, or PEAP Pass-through Server successfully send a Health Certificate Enrollment Protocol [MS-HCEP], the DHCP Extensions for NAP [MS-DHCPN], the Terminal Services Gateway Server Protocol [MS-TSGU], or the Protected Extensible Authentication Protocol [MS-PEAP] message containing the SoHR.

Extensions: None.

12.2 Task Context

This section describes the relationship between this task and its environment.

12.2.1 Task Environment

This task is accomplished by a NAP Enforcement Proxy when an encapsulated SoHR originating in the NPS is received by an RNAP client or by an EAP supporting RADIUS client and passed to the NAP Enforcement Proxy to be forwarded to a NAP EC over a PEP channel. The environment SHOULD meet the following requirement to support this task.

- **Requirement**: All the NAP ESs (HCEP HRA, DHCP Server, TSGU Server, and PEAP Pass-through Server) are running and have the ability to send messages in the supported protocol ([MS-HCEP], [MS-DHCPN], [MS-TSGU], and [MS-PEAP], respectively) to the NAP EC of their client counterpart.
  - **Reason for requirement**: The NAP ESs are used to send SoHR messages encapsulated within a message of the protocol they support to the NAP EC.
  - **Satisfying the requirement**:
    1. The NAP ESs have network access to the client counterpart at the NAP EC:
       - The network interface of the server computer is configured to operate on the local subnet.
       - The physical network path (network devices, Ethernet cables, and so on) between the local subnet and the NAP EC is connected.
       - All network devices between the local subnet and the NAP EC are configured to allow packet flow between the two entities.
       - The routing tables in the client computer are configured to enable correct packet routing between the client computer and the NAP Enforcement Proxy.
    2. The NAP ES services have been started.
  - **Verifying requirement is satisfied**:
    1. The client computer can successfully ping the NAP ES computer over the network.
    2. A sniffer trace performed during the SoH validation event shows packets of one of the supported protocols traveling between the NAP EC computer and the NAP ES computer.
    3. The NAP ES is shown as running within the list of services.
    4. No errors are logged by the NAP ES.
  - **Consequences of not satisfying requirement**: The task is unable to forward the encapsulated SoHR messages to the NAP EC.

Unless explicitly specified otherwise, the task implementation may assume its environment is properly configured and is not expected to verify that every requirement is satisfied. On the other hand, the task implementation should be able to gracefully handle errors which may be caused by environment misconfiguration or temporary dysfunction. The implementation should log these errors along with relevant information to allow troubleshooting.

12.2.2 Task Relationships

12.2.2.1 Black-Box Relationship Diagrams

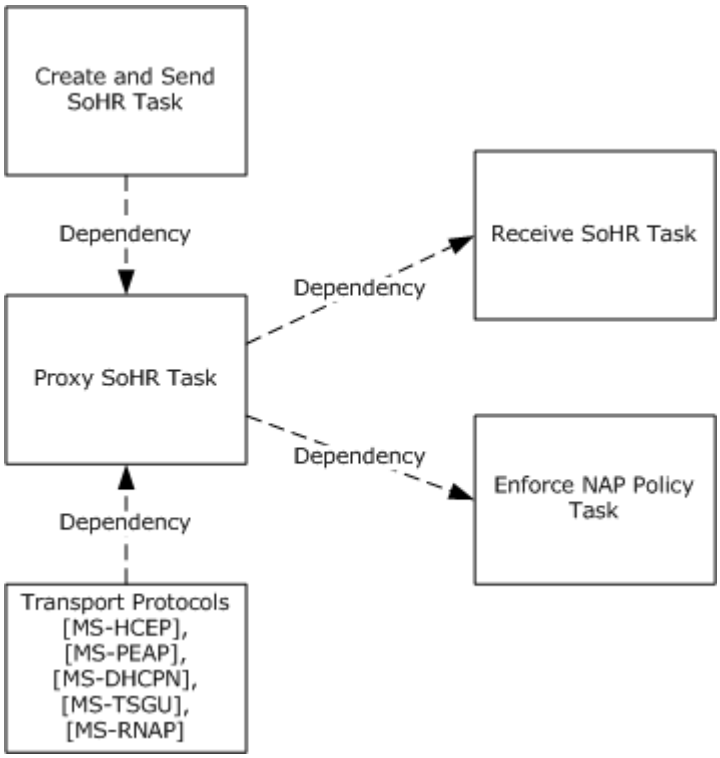

Figure 50: Proxy SoHR Task black-box relationships

In this task, the NAP health policy server sends encapsulated SoHR messages and enforcement decisions to the NAP Enforcement Proxy via the RADIUS channel.

12.2.2.2 Task Dependencies

This task depends on the Create and Send SoHR Task. This is because the Proxy SoHR Task must rely on the Create and Send SoHR Task to generate an SoHR message so that it can transfer this SoHR message to the NAP client.

The Receive SoHR Task has a dependency on the Proxy SoHR Task. Without the SoHR message sent from the Proxy SoHR Task, there is no use for the Receive SoHR Task.

The Enforce NAP Policy Task (section 11) has a dependency on the Proxy SoHR Task. Without the SoHR message and the enforcement decision, the NAP ESs cannot determine whether enforcement is required.

This task is also dependent on the protocols that govern the transport channel between the NAP health policy server and the NAP Enforcement Proxy. These include RADIUS Usage Guidelines [RFC3580], the Vendor-Specific RADIUS Attributes for Network Access Protection (NAP) Data Structure [MS-RNAP], and RADIUS Support for Extensible Authentication Protocol (EAP) [RFC3579], as well as the various underlying transport protocols that govern the PEP channels, including [MS-HCEP], [MS-DHCPN], [MS-TSGU], [MS-PEAP], and [MS-RNAP].

12.2.2.3 Task Influences

None.

12.2.3 Task Assumptions and Preconditions

To accomplish this task, the NAP Enforcement Proxy has the following preconditions and assumptions:

- The operating system on the server is trustable to the PDP.
- The underlying network infrastructures, such as the RADIUS channel, name and address resolution, and routing services, are configured correctly.
- The NAP health policy server is configured correctly by the server administrator.
- The PDP is trustable and functioning correctly.

12.2.4 Task Versioning and Capability Negotiation

The Proxy SoHR Task does not define any versioning and capability negotiation beyond those described in the specifications of the protocols supported or used by the task, as listed in section 2.3.

12.3 Task Architecture

This section describes the structure of the Proxy SoHR Task and the interrelationships among its parts.

12.3.1 Task Architectural Constraints

There can be more than one instance of the Proxy SoHR Task on each server. These task instances initialize themselves each time they start and run independently and concurrently. Different instances of this task on different servers also run independently. There are no constraints among these instances.

12.3.2 Task Abstract Data Model

This section describes state established, used, and maintained by processing rules of this task. State may be volatile or persisted. State may pertain to one, some, or all instances of the task. The task's state consists of the values of the named data elements (also called state variables) presented in this section. The overall organization of the data elements, with their names, is the Abstract Data Model. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Data Model, such association can be achieved in other ways. Implementations may depart from this model so long as their external behavior remains consistent with that described in this document.

PEP Channel Used: A common ADM element specified in section 4.1.1.

12.3.3 Task Abstract Parameters

This section describes data passed to an instance of this task at the time it is invoked or triggered. The parameters consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Parameter. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Parameters, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

SoHR: A buffer that contains either an SoHR message as specified in [TNC-IF-TNCCSPBSoH], or a PEAP message encapsulating an SoHR message as described in [MS-PEAP] section 2.2.8.2.2.

Authentication data: A collection of RADIUS attributes where each entry contains an attribute type number and a value. The attributes and their type numbers are defined in [RFC2865] section 5.44. The list can be empty.

The following parameters can be referred to collectively as enforcement decision parameters. Depending on the enforcement protocol used, as specified in the **PEP Channel Used** ADM element (section 12.3.2), some of these parameters can be irrelevant and might either be set to NULL, zero, or an arbitrary value, or not set at all. When any of these parameters is not relevant to the PEP Channel, the parameter is not expected to be set and it SHOULD be ignored during the processing rule.

- MS-Quarantine-State: For a list of possible values, see [MS-RNAP] section 2.2.1.9.
- MS-AFW-Zone: For a list of possible values, see [MS-RNAP] section 2.2.1.12.
- MS-AFW-Protection-Level: For a list of possible values, see [MS-RNAP] section 2.2.1.13.

12.3.4 Task Abstract Results

This section describes data returned by an instance of this task to its caller. The results consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Result. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Results, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

None.

12.3.5 White-Box Relationships

The following diagram shows the white-box relationships for the Proxy SoHR Task.

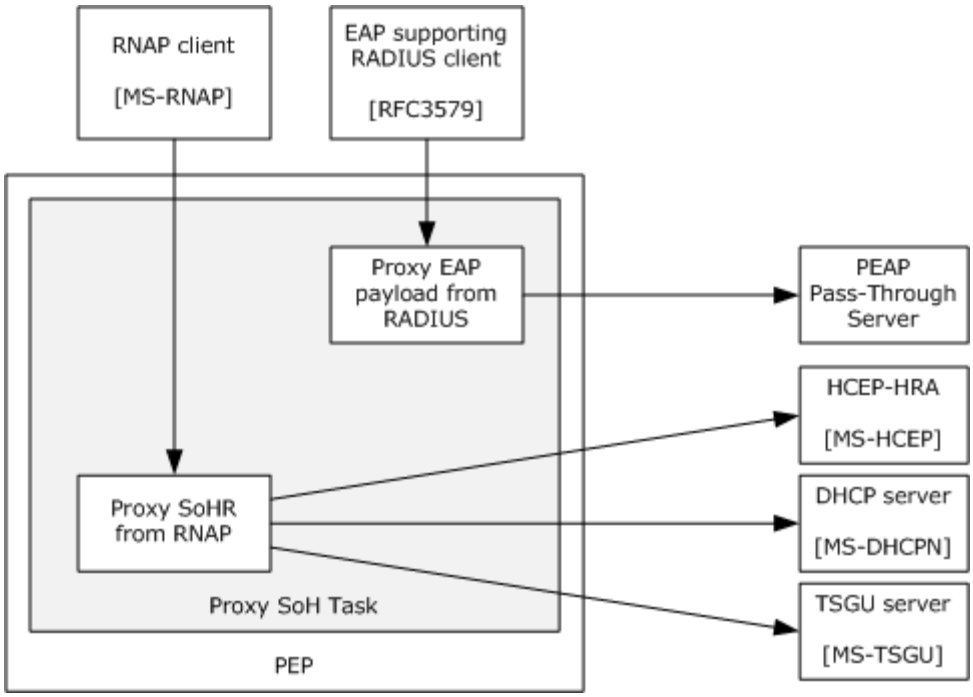

Figure 51: Proxy SoHR Task white-box relationships

When the value of the **PEP Channel Used** ADM element (section 12.3.2) is PEAP, the Proxy SoHR Task passes the SoHR encapsulated in PEAP to the PEAP Pass-through Server. Otherwise, the Proxy SoHR Task passes the SoHR and the other parameters to the appropriate NAP ES (HCEP HRA, DHCPN Server, TSGU server). From the perspective of the Create and Send SoHR Task (section 10) or the PEP, the Proxy SoHR Task provides SoHR encapsulation and transportation services.

12.3.6 Task Events

12.3.6.1 Task Timers

The Proxy SoHR Task does not impose any additional timers to the outside entities other than the timers in the underlying transport system.

12.3.6.2 Task Non-Timer Events

This task does not use or respond to any additional non-timer events other than those in the underlying transport system.

12.3.7 Task Architecture and Communication

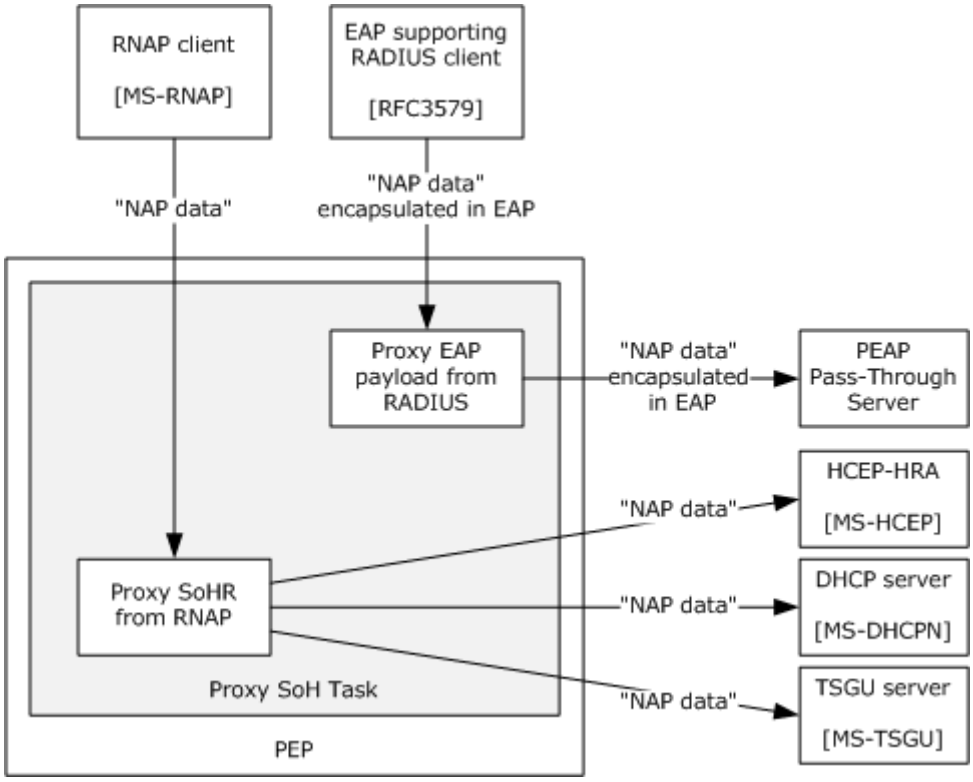

Figure 52: Proxy SoHR Task architecture and communication

12.3.8 Task Processing Rules

The following describes the operational flow of the Proxy SoHR Task:

1. A buffer containing the SoHR message, authentication data, and other enforcement decision parameters is passed into this task by the RNAP client, or an SoHR encapsulated in a PEAP message [MS-PEAP] is passed into this task by the EAP supporting RADIUS client.

2. Depending on the value of the **PEP Channel Used** ADM element (section 12.3.2), the NAP Enforcement Proxy performs the following:

    1. If the value of **PEP Channel Used** is PEAP, it passes the SoHR buffer to the PEAP Pass-through Server.

    2. Otherwise, it passes the SoHR message in the SoHR parameter and the other enforcement decision parameter to the appropriate NAP ES according to the value of **PEP Channel Used** (HCEP HRA, DHCPN Server, TSGU server).

3. If the NAP ES identifies that the enforcement decision requires access restriction to the client, the NAP ES performs the Enforce NAP Policy Task (section 11), passing the SoHR message and the enforcement decision as abstract parameters, and receives a response message in the appropriate protocol format.

4. The NAP ES sends the response message containing the SoHR message in its protocol to the NAP EC.

If an error is raised at any stage of the Proxy SoHR Task, the task fails.

12.3.9 Task Failure Scenarios

12.3.9.1 NAP Health Policy Server and PEP communication

These failures can be caused by:

- Misconfigurations on the NAP health policy server and/or NAP ES.
- Network connectivity issues wherein the NAP health policy server cannot communicate with the PEP.

If the NAP health policy server cannot communicate with the PEP, the NAP health policy server may not send any RADIUS messages to the PEP. The system cannot recover from this failure. This failure cannot be detected by the NAP server because RADIUS uses UDP.

12.3.9.2 NAP Client and PEP communication

These failures can be caused by:

- Mis-configurations on the NAP client and/or PEP.
- Network connectivity issues wherein the NAP client cannot communicate with the PEP.

If the NAP client cannot communicate with the PEP, The client may not have access to the network resources. The system may recover from certain types of failures (for example, the DHCP EC can attempt to connect to secondary DHCP server if there is no response from the primary server) and cannot recover from various other failures (for example, if the NAP client cannot communicate with an 802.1X switch or VPN server then the NAP System cannot recover from this failure). The failures can be detected by the timers on the enforcement clients.

12.4 Task Details

This section contains the details that complete the descriptions in earlier sections of the document. These are needed to understand and implement this task.

12.4.1 Task Precondition Details

For the complete list of task preconditions and assumptions, see section 12.2.3. Details for some of the preconditions are as follows:

- Depending on the specific configuration, any of the required PEP channels are functioning correctly, including the HTTP/S channel, the PEAP channel, or the DHCP channel.
- The RADIUS channel between the NAP health policy server and the PEP is functioning correctly, and the PEP recognizes the RADIUS protocol.

12.4.2 Task Initialization of External Entities

None.

12.4.3 Task Event Details

12.4.3.1 Task Timer Details

This task does not impose any additional timers. Timers are related to the underlying transports and they are described in [MS-TSGU], [MS-DHCPN], [MS-PEAP], and [MS-HCEP].

12.4.3.2 Task Non-Timer Event Details

This task does not impose any additional non-timer events. Non-timer events are related to the underlying transports and they are described in [MS-DHCPN], [MS-PEAP], and [MS-HCEP].

12.4.4 Task Architectural Details

This section illustrates an example of a NAP health policy server sending an SoHR.

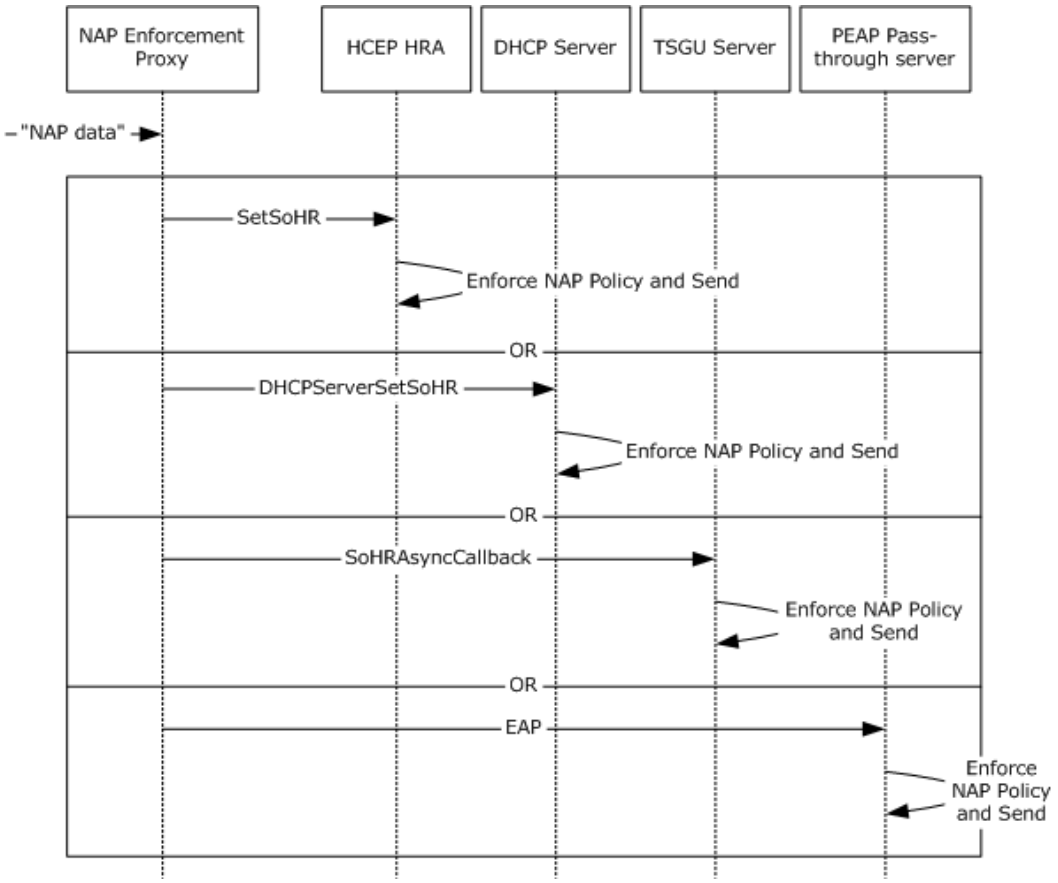

Figure 53: Sequence diagram for the main success scenario of the Proxy SoHR Task

1. An SoHR buffer is passed to the NAP Enforcement Proxy.

2. The NAP Enforcement Proxy passes the SoHR buffer to one of the NAP ESs (HCEP HRA, DHCPN Server, TSGU Server, PEAP Pass-through Server).

3. The NAP ES performs the Enforce NAP Policy Task (section 11) and sends the SoHR message to the NAP client.

12.4.5 Task Processing Rule Details

The following describes the operational details of the Proxy SoHR Task:

1. The RNAP client passes to this task an SoHR message, authentication data, and the enforcement decision parameters, or an EAP supporting RADIUS client passes the SoHR encapsulated in a Protected Extensible Authentication Protocol [MS-PEAP] message from the EAP-Message attribute as specified in [RFC3579] section 3.1.

2. Depending on the value of the **PEP Channel Used** ADM element (section 12.3.2), the NAP Enforcement Proxy employs the NAP ES to inject and send the SoHR in the response of its protocol and then sends it to the NAP EC as follows:

    1. If **PEP Channel Used** is PEAP, the NAP Enforcement Proxy passes the SoHR buffer which contains the SoHR encapsulated in a PEAP message to the PEAP Pass-through Server and performs the behavior described in [RFC3748] section 2.3.

    2. If **PEP Channel Used** is HCEP, the NAP Enforcement Proxy calls SetSoHR as described in [MS-HCEP] section 3.2.4, and passes to it the SoHR and the authentication data. The following enforcement parameters are also passed:

        - If the MS-Quarantine-State enforcement decision parameter is set to 0x00000000 (full access) or 0x00000002 (on probation), the compliance parameter is set to TRUE.
        - MS-AFW-Zone
        - MS-AFW-Protection-Level

    3. If **PEP Channel Used** is DHCP, the NAP Enforcement Proxy passes the SoHR and the authentication data to the DhcpServerSetSoHR abstract interface defined in [MS-DHCPN] section 3.2.7.3.

    4. If **PEP Channel Used** is TSG, the NAP Enforcement Proxy passes the SoHR and the authentication data to the SoHRAsyncCallback abstract interface defined in [MS-TSGU] section 3.1.3.

3. If the MS-Quarantine-State enforcement decision parameter is 1 (quarantined), the NAP ES performs the Enforce NAP Policy Task (section 11), passing the SoHR message and the enforcement decision as abstract parameters, and receives a response message in the appropriate protocol format.

4. The NAP ES sends the response message containing the SoHR message in its protocol to the NAP EC.

12.5 Task Security

The PEP and the NAP EC must maintain a trust relationship. For additional information about security considerations, see section 16, as well as the Security sections of the referenced protocol Technical Documents.

13 Receive SoHR Task

This section describes the task of receiving SoHR messages by the NAP agent. The SoHR messages are transported on PEP channels. The SoHR messages can arrive on a number of different transport protocols. The format of the SoHR message is specified in [TNC-IF-TNCCSPBSoH]. The protocols that can be used to transport the SoHR are specified in [MS-HCEP], [MS-TSGU], [MS-PEAP], and [MS-DHCPN]. This task also stores the certificate conveyed in the HCEP response.

Note All common information defined in section 4 is not applicable to this task.

13.1 Task Overview

13.1.1 Task Purpose

The purpose of this task is to ensure the reception of SoHR messages from PEP channel(s) or HCEP Channel after they were sent by the NAP health policy server using the Proxy SoHR Task (section 12).

13.1.2 Task Applicability

This task is used when an SoHR message has been sent from the PEP through a specific PEP channel to the client computer. This task is not applicable if the NAP System is not deployed or enabled on the client computer.

13.1.3 Task Use Cases

13.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Receive SoHR Task (section 13) are as follows:

NAP ECs: The NAP ECs include the TSGU client defined in [MS-TSGU] section 3.5, the DHCPN client defined in [MS-DHCPN] section 3.1, the PEAP peer defined in [MS-PEAP] section 3.2, and the HCEP **HCEA** defined in [MS-HCEP] section 3.1. NAP ECs trigger this task when the message is received over the corresponding PEP channel. The main interest of the NAP ECs is for this task to process all message-reception events.

NAP agent: The main software component on the NAP client that mediates between the NAP ECs, SoH client, and the Certificate Storage Manager. It delivers received SoHR messages to the SoH client and passes computer certificates received from the HCEP HCEA to the Certificate Store Manager for storage. The NAP agent has two main interests in the Receive SoHR Task (section 13):

- Deliver received SoHR messages to the SoH client.
- Pass computer certificates received from the HCEP HCEA to the Certificate Store Manager for storage.

13.1.3.2 Supporting Actors and Task Interests Summary

Certificate Store Manager: This is the operating system component that implements the functionality of the Certificate Store, including storage and retrieval of certificates. Specifically, it implements the **Persisted.ComputerCertificates** ADM element specified in [MS-CAESO] section 4.3.2.4. The main interest of the Certificate Store Manager in the Receive SoHR Task (section 13) is to store certificates accompanying an SoHR message originating from the HCEP HCEA. The client certificate is not consumed by any task described in this document. The client certificate is intended for use by other applications, such as IPSec as described in [MS-WSO] section 3.1.1.6.

SoH client: As defined in [TNC-IF-TNCCSPBSoH], the SoH client is responsible for syntactical validation and processing of SoHR messages. The main interest of the SoH client in the Receive SoHR Task is to process all SoHR messages received from the NAP ECs.

13.1.3.3 Use Case Diagrams

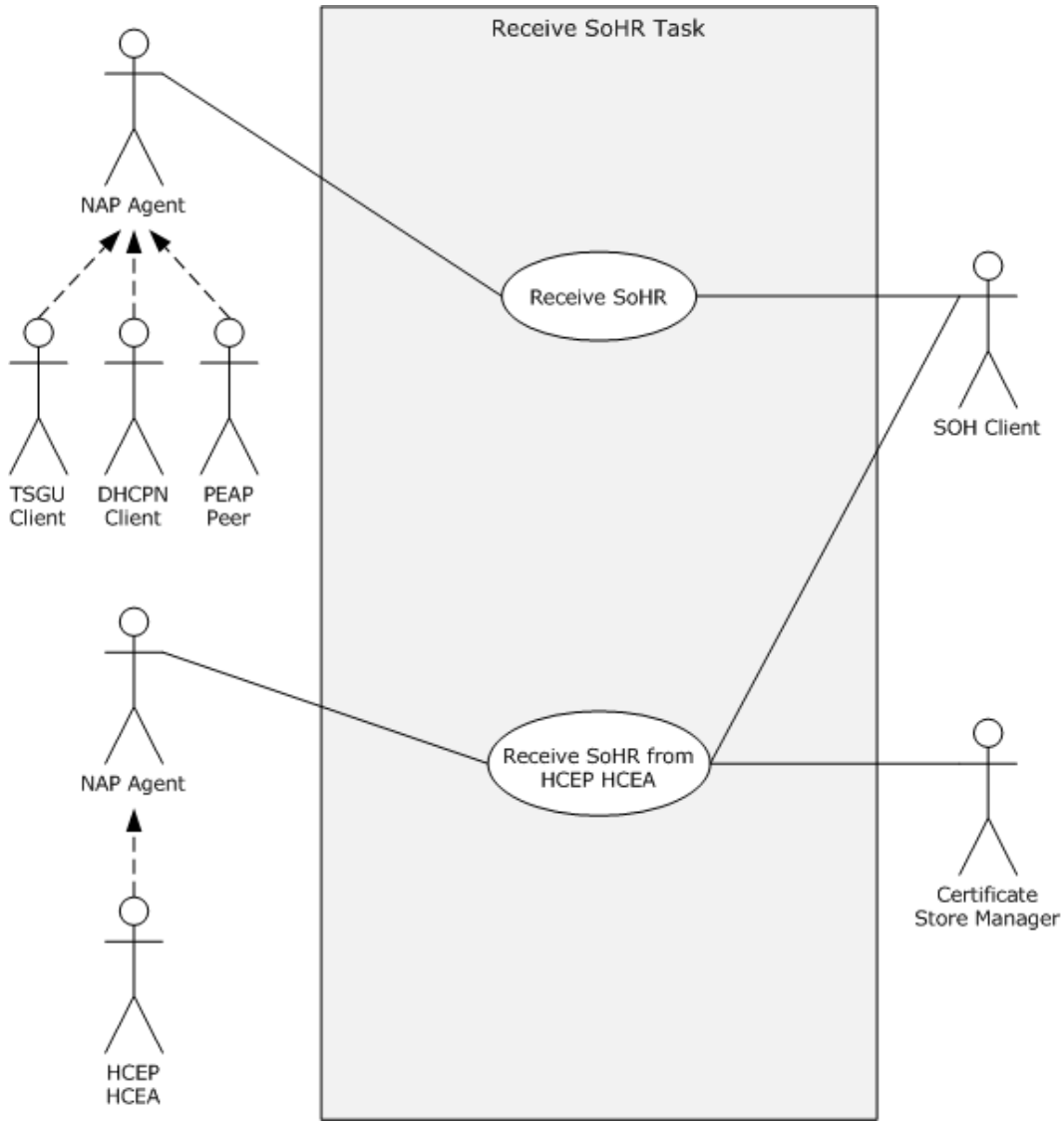

Figure 54: Receive SoHR Task use case diagram

13.1.3.4 Use Case: Receive SoHR from PEP -- NAP Agent

This use case is associated with the use case diagram in section 13.1.3.3.

Goal: To extract SoHR messages from a specific PEP channel and deliver the messages to the SoH client.

Context of Use: This use case is initiated when an SoHR message has been sent through one of the following PEP channels: TSGU, DHCPN, or PEAP, and the message has arrived at the client computer.

Direct Actor: The direct actor in this use case is the NAP agent, which is the main software component on the NAP client that mediates between the NAP ECs, the SoH client, and the Certificate Storage Manager. It delivers received SoHR messages to the SoH client and passes Computer Certificates received from an HCEP HCEA to the Certificate Store Manager for storage. Its interest in this use case is to deliver received SoHR messages to the SoH client.

Primary Actor: The primary actor role in this use case is performed by the NAP ECs, which include the TSGU client defined in [MS-TSGU] section 3.5, the DHCPN client defined in [MS-DHCPN] section 3.1, and the PEAP peer defined in [MS-PEAP] section 3.2. NAP ECs trigger this task when the message is received over the corresponding PEP channel. The main interest of the NAP ECs is for this task to process all message reception events.

Supporting Actor: The SoH client. As defined in [TNC-IF-TNCCSPBSoH], the SoH client is responsible for syntactical validation and processing of SoHR messages. In the Receive SoHR Task, the SoH client processes all SoHR messages received from the NAP ECs.

Stakeholders and Interests: None.

Preconditions:

- The NAP client components on the client computer are deployed and configured correctly by the client administrator.
- One or more of the NAP ECs are functional.

Minimal Guarantees:

- The NAP agent processes all message-reception events triggered by NAP ECs.
- The NAP Agent sends all received SoHR messages to SoH client.

Success Guarantee: The SoHR message is recovered from the protocol message and passed to the SoH client.

Trigger: The arrival of a protocol message to the NAP EC.

Main Success Scenario:

1. A protocol message is successfully received from the PEP channel on the NAP EC.
2. The NAP EC extracts the SoHR message from the protocol message. More specifically, depending on the type of protocol that is being used by the NAP system, the following occurs:
   - If the NAP EC is connected to a TSG server, the NAP EC extracts the SoHR message from the encapsulation inside the TSG response, as described in [MS-TSGU] section 2.2.9.2.1.5.
   - If the NAP EC is connected to a DHCPN server, the NAP EC extracts the SoHR message from the encapsulation inside the DHCPN response as described in [MS-DHCPN] section 2.2.1.
   - If the NAP EC is connected to a PEAP server, the NAP EC extracts the SoHR message from the encapsulation inside the PEAP response as described in [MS-PEAP] section 2.2.8.1.3.
3. The NAP agent retrieves the SoHR message from the NAP EC.

4. The NAP agent sends the SoHR to the SoH client.

Extensions: None.

13.1.3.5 Use Case: Receive SoHR from HCEP HCEA -- NAP Agent

This use case is associated with the use case diagram in section 13.1.3.3.

Goal: To extract SoHR messages (see [TNC-IF-TNCCSPBSoH]) from the HCEP channel and deliver them to the SoH client, and to receive Computer Certificates accompanying an SoHR in the HCEP response and store them for use by any application.

Context of Use: This use case is initiated when an HCEP response protocol message arrives at the client computer.

Direct Actor: The direct actor in this use case is the NAP agent, which is the main software component on the NAP client that mediates between the NAP ECs, the SoH client, and the Certificate Storage Manager. It delivers received SoHR messages to the SoH client and passes Computer Certificates received from the HCEP HCEA to the Certificate Store Manager for storage. Its interests in this use case are:

- Deliver received SoHR messages to the SoH client.
- Pass Computer Certificates received from the HCEP HCEA to the Certificate Store Manager for storage.

Primary Actor: The primary actor role in this use case is performed by the HCEP HCEA defined in [MS-HCEP] section 3.1. The HCEP HCEA triggers this task when the message is received over the HCEP channel. The main interest of the HCEP HCEA is for this use case to process all message reception events.

Supporting Actors:

- **Certificate Store Manager:** The operating system component that implements the functionality of the Certificate Store, including storage and retrieval of the certificates. Specifically, it implements the **Persisted.ComputerCertificates** ADM element specified in [MS-CAESO] section 4.3.2.4. The main interest of the Certificate Store Manager in the Receive SoHR Task is to store the certificate accompanying an SoHR message originating from the HCEP HCEA. The client certificate is not consumed by any task described in this document. The client certificate is intended for use by other applications; for example, IPSec as described in [MS-WSO] section 3.1.1.6.
- **SoH client:** As defined in [TNC-IF-TNCCSPBSoH], the SoH client is responsible for syntactical validation and processing of SoHR messages. The main interest of the SoH client in the Receive SoHR Task is to process all SoHR messages received from the NAP ECs.

Stakeholders and Interests: None.

Preconditions:

- The NAP client components on the client computer are deployed and configured correctly by the client administrator.
- The HCEP channel is functional.

Minimal Guarantees:

- The NAP agent processes all message-reception events triggered by the HCEP HCEA.
- The NAP agent sends all received SoHR messages to the SoH client.
- The NAP agent sends all received certificates to the Certificate Storage Manager.

Success Guarantee: The SoHR message is recovered from the protocol message and passed to the SoH client. The certificate is stored by the Certificate Store Manager.

Trigger: The arrival of the HCEP response to the HCEP HCEA.

Main Success Scenario:

1. An HCEP response message is successfully received by the HCEP HCEA
2. The HCEP HCEA extracts the SoHR message from the HCEP Response as specified in [MS-HCEP] section 2.2.2.2.
3. The HCEP HCEA extracts the encapsulated certificate as described in [MS-HCEP] section 2.2.2.3.
4. The NAP agent retrieves the SoHR message from the HCEP HCEA.
5. The NAP agent retrieves the certificate from the HCEP HCEA.
6. The NAP agent sends the SoHR to the SoH client.
7. The NAP agent sends the extracted certificate to the Certificate Store Manager for storage.

Extensions: None.

13.2 Task Context

This section describes the relationship between this task and its environment.

13.2.1 Task Environment

This task is accomplished by the NAP EC in an environment where the NAP EC communicates with PEP or NAP health policy server using specific communication protocols, such as those defined in [MS-TSGU], [MS-HCEP], [MS-PEAP], and [MS-DHCPN]. The environment should meet the following requirement to support this task.

- **Requirement:** At least one NAP EC is correctly configured and enabled.
  - **Reason for requirement:** Correct configuration and enablement are required for the NAP EC to receive and extract SoHR messages and pass them to the NAP agent.
  - **Satisfying the requirement:** The NAP ECs are configured by the system administrator, for example, by means of a Group Policy (see [MS-GPNAP]).
  - **Verifying requirement is satisfied:** No errors related to configuration are logged by the NAP agent and NAP ECs.
  - **Consequences of not satisfying requirement:** The task is unable to receive messages from the PEP channels.
- **Requirement:** There is network connectivity between the PEP computer and the NAP client computer.

- **Reason for requirement:** The PEP computer communicates with the client computer.
- **Satisfying the requirement:**

  1. The network interface of the client computer is configured to operate on the local subnet.
  2. The components of the physical network path (network devices, Ethernet cables, and so on) between the local subnet and the PEP computer are connected.
  3. All network devices between the local subnet and the PEP computer are configured to allow packet flow between the two entities.
  4. The network infrastructure that provides name and address resolution and routing services is functional.

- **Verifying requirement is satisfied:** The client computer can successfully ping the PEP computer over the network.
- **Consequences of not satisfying requirement:** SoHR messages cannot be received.

Unless explicitly specified otherwise, the task implementation assumes its environment is properly configured and is not expected to verify that every requirement is satisfied. On the other hand, the task implementation should be able to gracefully handle errors possibly caused by environment misconfiguration or temporary dysfunction. The implementation should log these errors along with relevant information to allow troubleshooting.

13.2.2 Task Relationships

13.2.2.1 Black-Box Relationship Diagrams

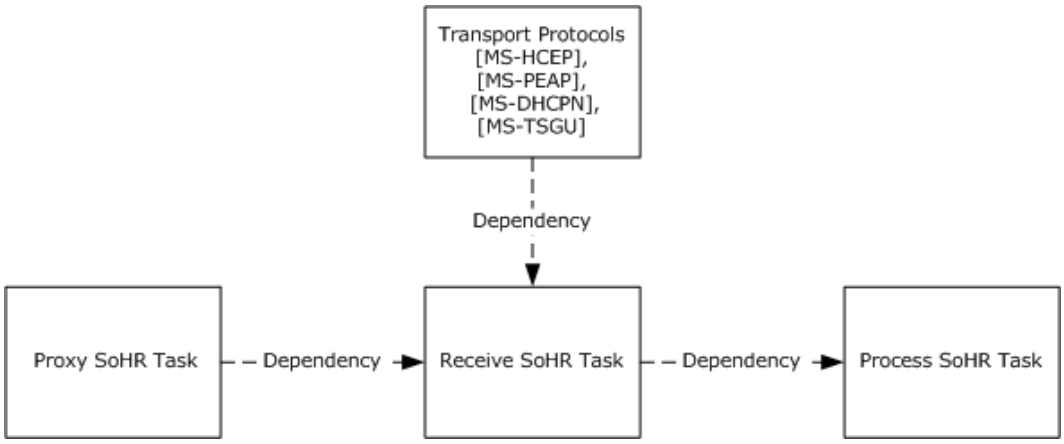

Figure 55: Receive SoHR Task black-box relationships

In this task, the NAP client receives encapsulated SoHR messages from the PEP via a PEP transport channel or from NAP health policy server via a HCEP transport channel.

13.2.2.2 Task Dependencies

The Receive SoHR Task has a dependency on the Proxy SoHR Task and Enforce NAP Policy Task. Without the SoHR messages sent from a NAP health policy server, there is no use of the Receive SoHR Task.

The Process SoHR Task has a dependency on the Receive SoHR Task. The NAP client must rely on the Receive SoHR Task to get the SoHR message first, until then it cannot start the process on the receive SoHR.

This task is also dependent on the various protocols that govern the PEP transport channels (such as those described in [MS-DHCPN], [MS-TSGU], [MS-PEAP]), or [MS-HCEP], depending on which EC is used in the specific NAP deployment.

13.2.2.3 Task Influences

None.

13.2.3 Task Assumptions and Preconditions

To accomplish this task, the NAP client has the following preconditions and assumptions:

- The operating system and hardware comprising on the client computer is trustworthy.
- The client administrators are trustworthy. The client administrators are responsible for enabling and configuring the NAP client correctly. They are also responsible for the integrity of executables that provide NAP client services.
- The underlying network infrastructures, such as the PEP channel(s) or HCEP Channel, name and address resolution, and routing services, are configured correctly.
- NAP client is enabled and correctly configured by the client administrator.
- The PEP is trustworthy and functioning correctly when using PEP channel to transport SoHR.
- The NAP health policy server is trustworthy when using HCEP Channel to transport SoHR.

13.2.4 Task Versioning and Capability Negotiation

The Receive SoHR Task does not define any versioning and capability negotiation beyond those described in the specifications of the protocols supported or used by the task, as listed in section 2.3.

13.3 Task Architecture

This section describes the structure of the Receive SoHR Task and the interrelationships among its parts.

13.3.1 Task Architectural Constraints

There can be more than one instance of the Receive SoHR Task on each client computer if multiple PEP channels are deployed. These task instances initialize themselves each time they start and run independently. Different instances of this task on different client computers also run independently. There are no constraints among these instances.

13.3.2 Task Abstract Data Model

This section describes state established, used, and maintained by processing rules of this task. State may be volatile or persisted. State may pertain to one, some, or all instances of the task. The task's state consists of the values of the named data elements (also called state variables) presented in this section. The overall organization of the data elements, with their names, is the Abstract Data Model. It is intended to facilitate the reader's conceptual understanding of the specification. While a

task's processing rules may depend upon associations established by the structure of its Abstract Data Model, such association can be achieved in other ways. Implementations may depart from this model so long as their external behavior remains consistent with that described in this document.

13.3.3 Task Abstract Parameters

This section describes data passed to an instance of this task at the time it is invoked or triggered. The parameters consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Parameter. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Parameters, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

SoHR: SoHR messages are received in the PEP channel.

Client Certificate: The client certificate conveyed in the HCEP response.

13.3.4 Task Abstract Results

This section describes data returned by an instance of this task to its caller. The results consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Result. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Results, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

The Receive SoHR Task is expected to receive and de-encapsulate the SoHR message sent from a PEP or NAP health policy server each time it is called. The SoHR de-encapsulation must follow the format defined in [MS-HCEP], [MS-DHCPN], [MS-TSGU], or [MS-PEAP], depending on the PEP channel(s) that are used in the specific NAP deployment.

13.3.5 White-Box Relationships

The white box relationships for the Receive SoHR Task are shown in the following figure.

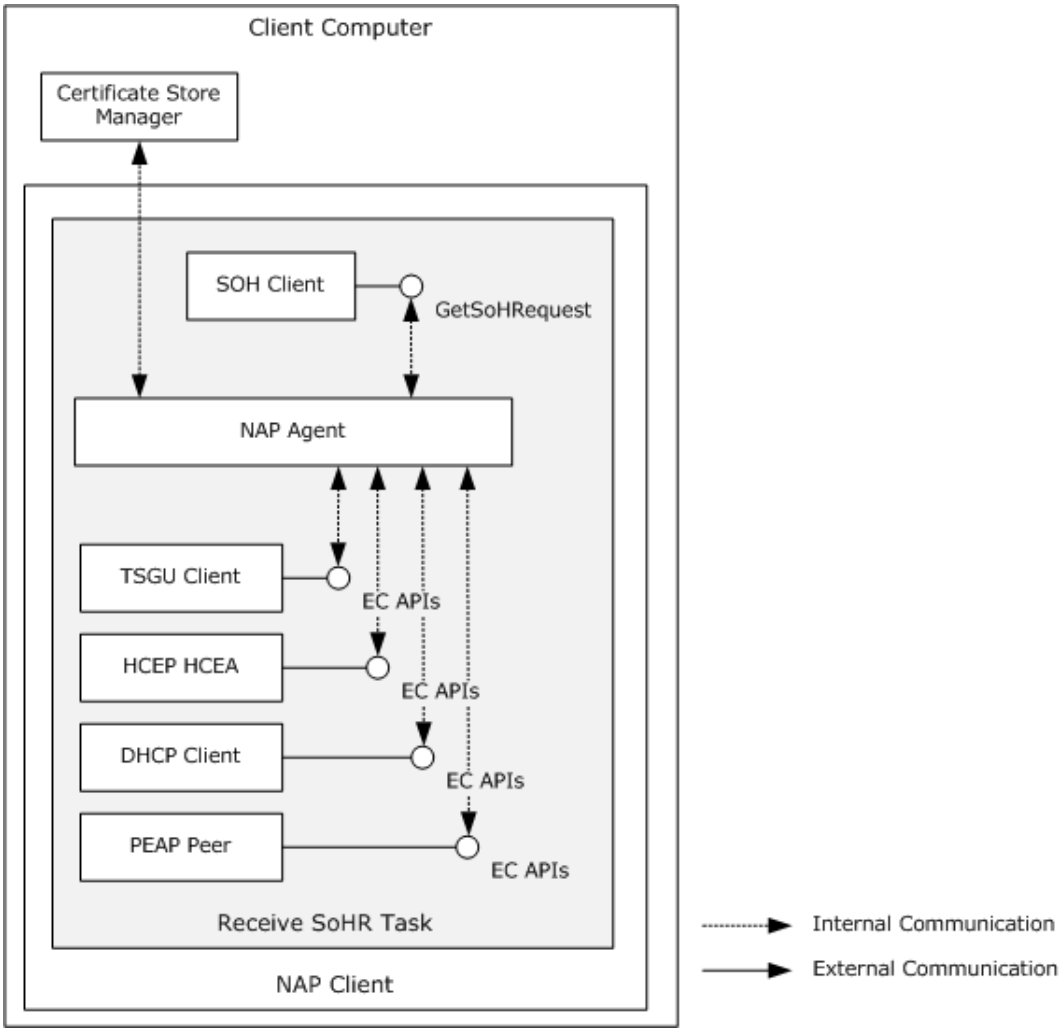

Figure 56: Receive SoHR Task white-box relationships

The Receive SoHR Task involves three major NAP client components: system health agent (SHA), NAP agent, and NAP EC.

From the Process SoHR Task's perspective, the Receive SoHR Task receives and prepares the SoHR messages so that they can be consumed later by the NAP agent. The NAP EC de-encapsulates the SoHR messages from the PEP-specific transport protocol messages, as defined in [MS-DHCPN], [MS-TSGU], [MS-PEAP], and [MS-HCEP]. These de-encapsulated and syntax validated SoHR messages are handed to and are finally consumed by the NAP agent using the Process SoHR Task.

13.3.6 Task Events

13.3.6.1 Task Timers

The Receive SoHR Task does not impose any additional timers to the outside entities other than the timers in the underlying transport system.

13.3.6.2 Task Non-Timer Events

This task does not use or respond to any additional non-timer events other than those in the underlying transport system.

13.3.7 Task Architecture and Communication

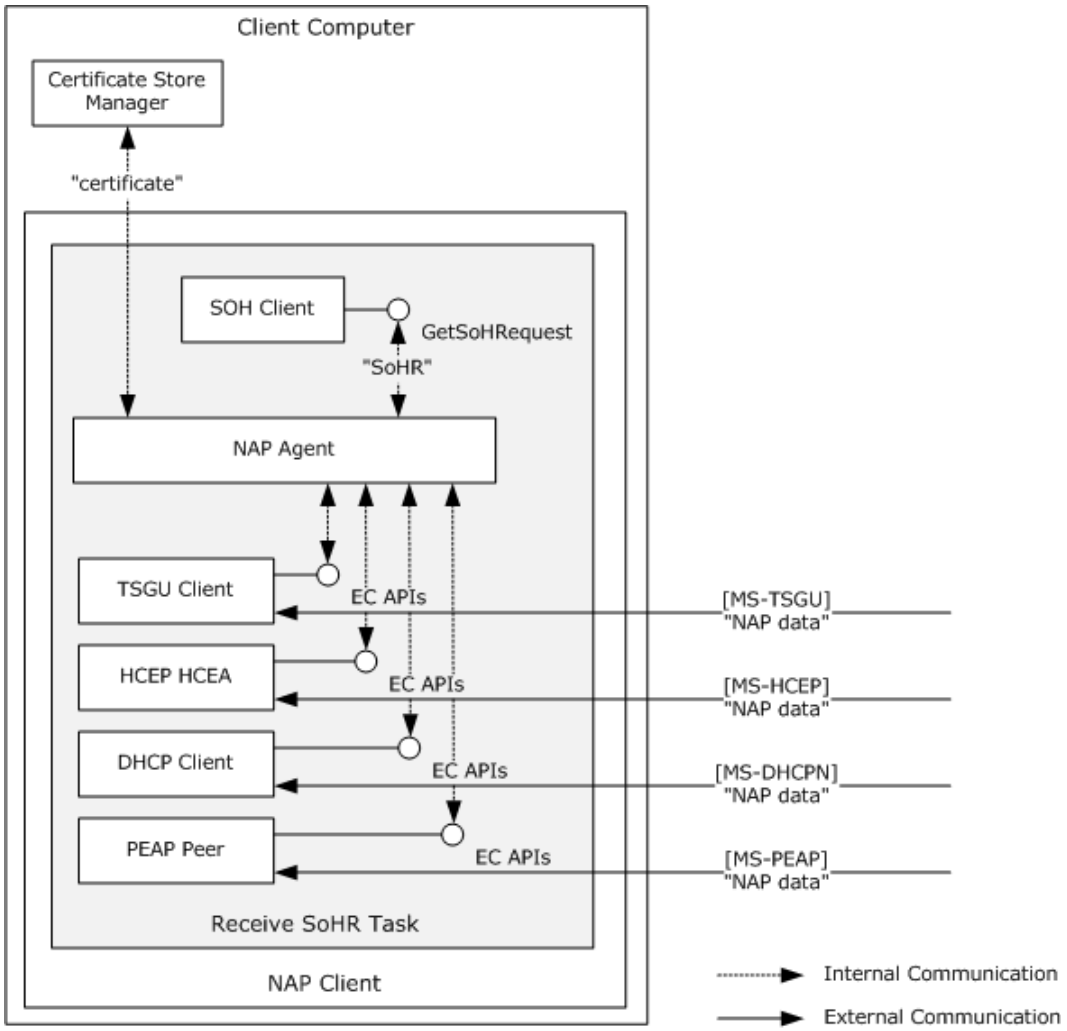

Figure 57: Receive SoHR Task architecture and communication

As shown in the preceding figure, the ECs on the NAP client communicate with PEP or NAP health policy server using various communication protocols depending on specific NAP deployment. After receiving the encapsulated SoHR messages, the ECs recover the SoHR messages using the specific communication protocol that was used for the encapsulation (such as those described in [MS-TSGU], [MS-HCEP], [MS-PEAP], or [MS-DHCPN]) and passes SoHR message to NAP agent.

13.3.8 Task Processing Rules

The following describes the operational flow of the Receive SoHR Task:

1. The EC receives the SoHR encapsulated in one of the supported transport protocols, including [MS-HCEP], [MS-DHCPN], [MS-TSGU], or [MS-PEAP].

2. The EC recovers the SoHR message from its encapsulation. For example, in the case where the PEP is an 802.1X NAS/NAD, the SoHR message is received from an 802.1X channel extended from a PEAP channel, and is recovered from the PEAP response [MS-PEAP].

3. If the HCEP protocol is used and the HCEP response contains a health certificate, the HCEA extracts the certificate and places it in the **Persisted.ComputerCertificates** ADM element specified in [MS-CAESO] section 4.3.2.4.

If an error is raised at any stage of the Receive SoHR Task, the task fails.

13.3.9 Task Failure Scenarios

13.3.9.1 NAP Agent Communication with EC

These failures are caused by an error with the initialization or registration of the enforcement client. The NAP System relies on the communication between the NAP agent service and an installed enforcement client to provide the enforcement client with health status and receive information about the level of network access granted to the client computer. A client experiencing this failure will not be able to receive and process SoHR messages sent from the NAP health policy server, which may make the client unhealthy or cause the client to be in a restricted state. These failures are not detected by the NAP System. The NAP System cannot recover from such a failure.

13.3.9.2 NAP Client and PEP Communication

These failures can be caused by:

- Misconfigurations on the NAP client and/or PEP.
- Network connectivity issues wherein the NAP client cannot communicate with the PEP.

If the NAP client cannot communicate with the PEP, the client may not have access to the network resources. The system may recover from certain types of failures (for example, the DHCP EC can attempt to connect to secondary a DHCP server if there is no response from the primary server) and cannot recover from various other failures (for example, if the NAP client cannot communicate with an 802.1x switch or VPN server then the NAP System cannot recover from this failure). The failures can be detected by the timers on the enforcement clients.

13.3.9.3 HCEA and NAP Health Policy Server Communication

These failures can be caused by:

- Misconfigurations on the NAP health policy server and/or HCEA.
- Network connectivity issues wherein the NAP health policy server cannot communicate with the HCEA.

If the NAP health policy server cannot communicate with the HCEA, the NAP health policy server will not send health certificates to the HCEA. The system cannot recover from this failure. This failure can be detected by the HCEA.

13.4 Task Details

This section contains the details that complete the descriptions in earlier sections of the document. These are needed to understand and implement this task.

13.4.1 Task Precondition Details

For the complete list of task preconditions and assumptions, see section 13.2.3. Details for some of the preconditions are as follows:

- The NAP agent service is started and initialized correctly on the client computer.
- ECs are correctly configured, enabled, and bound to the NAP agent so that the NAP agent has a complete EC list.
- Depending on the specific configuration, any of the required PEP channels (the HTTP/S channel, the PEAP channel, or the DHCP channel) are functioning correctly.

13.4.2 Task Initialization of External Entities

None.

13.4.3 Task Event Details

13.4.3.1 Task Timer Details

This task does not impose any additional timers. Timers are related to the underlying transports and are defined in [MS-TSGU], [MS-DHCPN], [MS-PEAP], and [MS-HCEP].

13.4.3.2 Task Non-Timer Event Details

This task does not impose any additional non-timer events. Non-timer events are related to the underlying transports and are defined in [MS-TSGU], [MS-DHCPN], [MS-PEAP], and [MS-HCEP].

13.4.4 Task Architectural Details

This section illustrates an example of a NAP client receiving an SoHR.

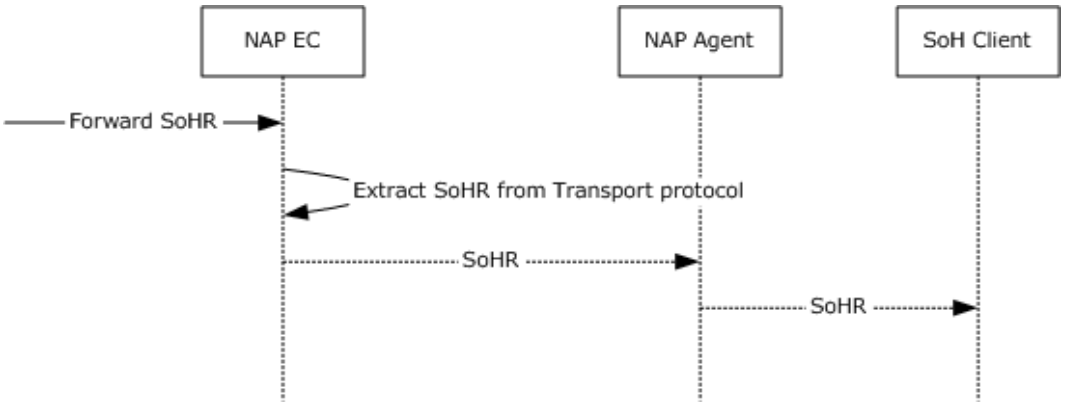

Figure 58: Sequence diagram for the main success scenario of the Receive SoHR Task

1. The NAP EC receives a PEP channel message containing the SoHR.

2. The NAP EC forwards the SoHR to the NAP agent.

3. The NAP agent forwards the SoHR to the SoH client.

13.4.5 Task Processing Rule Details

The following describes the operational details of the Receive SoHR Task:

1. The EC receives the SoHR via one of the supported transport protocols, including [MS-HCEP], [MS-DHCPN], [MS-TSGU], or [MS-PEAP].

2. The EC extracts the SoHR message from its encapsulation. The SoHR message is sent as an attribute in the encapsulating protocols as specified in [MS-HCEP] section 2.2.1.4, [MS-DHCPN] section 2.2.1.1, [MS-TSGU] section 2.2.9.2.1.4, or [MS-PEAP] section 2.2.8.2.

3. The EC passes the SoHR to the NAP agent using the EC APIs specified in section 3.2.1.

4. If the protocol used is HCEP, the response may contain an PKCS#7 certificate as specified in [MS-HCEP] section 2.2.2.4. If the certificate exists, the EC extracts the certificate from the HTTP message body (see [MS-HCEP] section 2.2.2.3) and places it in the **Persisted.ComputerCertificates** ADM element specified in [MS-CAESO] section 4.3.2.4.

13.5 Task Security

There are no task-specific security considerations. For additional information about security considerations, see section 16, as well as the Security sections of the referenced protocol Technical Documents.

14 Process SoHR Task

This section describes the task of processing SoHR messages on a client computer. This task is performed mainly by the SoH client and the NAP agent. This task processes the SoHR message passed by the enforcement client after the Receive SoHR Task (section 13) to the SoH client, as specified in [TNC-IF-TNCCSPBSoH]. If the SoHR indicates that remediation is required, the Remediate Client Health Task (section 15) is triggered.

Note All common information defined in section 4 is not applicable to this task.

14.1 Task Overview

14.1.1 Task Purpose

The purpose of this task is to ensure that the SoHR is valid and correctly processed, which results in the Remediate Client Health Task (section 15) being triggered when the computer is determined to be noncompliant and the NAP health policy server allows automatic remediation. The remediation process is defined in the Remediate Client Health Task.

14.1.2 Task Applicability

This task is used when an SoHR message has been successfully received by the NAP EC by using the Receive SoHR Task (section 13). This task is not applicable if the NAP System is not deployed.

14.1.3 Task Use Cases

14.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Process SoHR Task (section 14) are as follows:

NAP agent: The NAP agent is the main software component on the NAP client computer. It orchestrates the execution of NAP-related operations, such as receiving SoHR messages, passing SoHR messages internally to the SoH client, triggering remediation phases, displaying remediation results using a human-readable interface, and so on. The purpose of the NAP agent in this task is to ensure that remediation and enforcement are based on whether a valid SoHR was received in the transport payload.

DHCP client: This protocol client is used to send DHCP Extensions for NAP messages (as specified in [MS-DHCPN]) to a DHCP server on the PEP computer. When DHCP enforcement is used, it acts in the role of an enforcement client (EC) in this use case. Unlike other ECs, where enforcement occurs before this task and is protocol-specific, to achieve enforcement in IPv6, the DHCP client requires that health status information about the computer is obtained and the IPv6 remediation servers transport the SoHR as specified in [MS-DHCPN] section 3.1.5.2. The purpose of the DHCP client in the task is the analysis of the SoHR's validity.

Remediate Client Health Task: The Remediate Client Health Task (section 15) performs remediation of noncompliant components on the client computer, as necessary. Each execution of the Remediate Client Health Task receives an **SoHRReportEntry** ([TNC-IF-TNCCSPBSoH]) extracted from the SoHR message by the Process SoHR Task. The purpose of the Remediate Client Health Task in this task is to obtain only valid SoHR data received in the transport messages.

14.1.3.2 Supporting Actors and Task Interests Summary

SOH client: This actor uses the Statement of Health for Network Access Protection (NAP) Protocol processing rules specified in [TNC-IF-TNCCSPBSoH] to evaluate SoHR packets. This is accomplished by extracting the correlation ID, the SSoH header prepended to the SoHR packet, and the **SoHReportEntry** as specified in [TNC-IF-TNCCSPBSoH]. This task uses the SoH client to verify the validity of the structure and determine whether remediation is required.

NAP Human Interface: This actor displays NAP messages that describe the health status of the client computer in a human-readable format. The task employs this service to display health results to the user before a potential remediation phase.

14.1.3.3 Use Case Diagrams

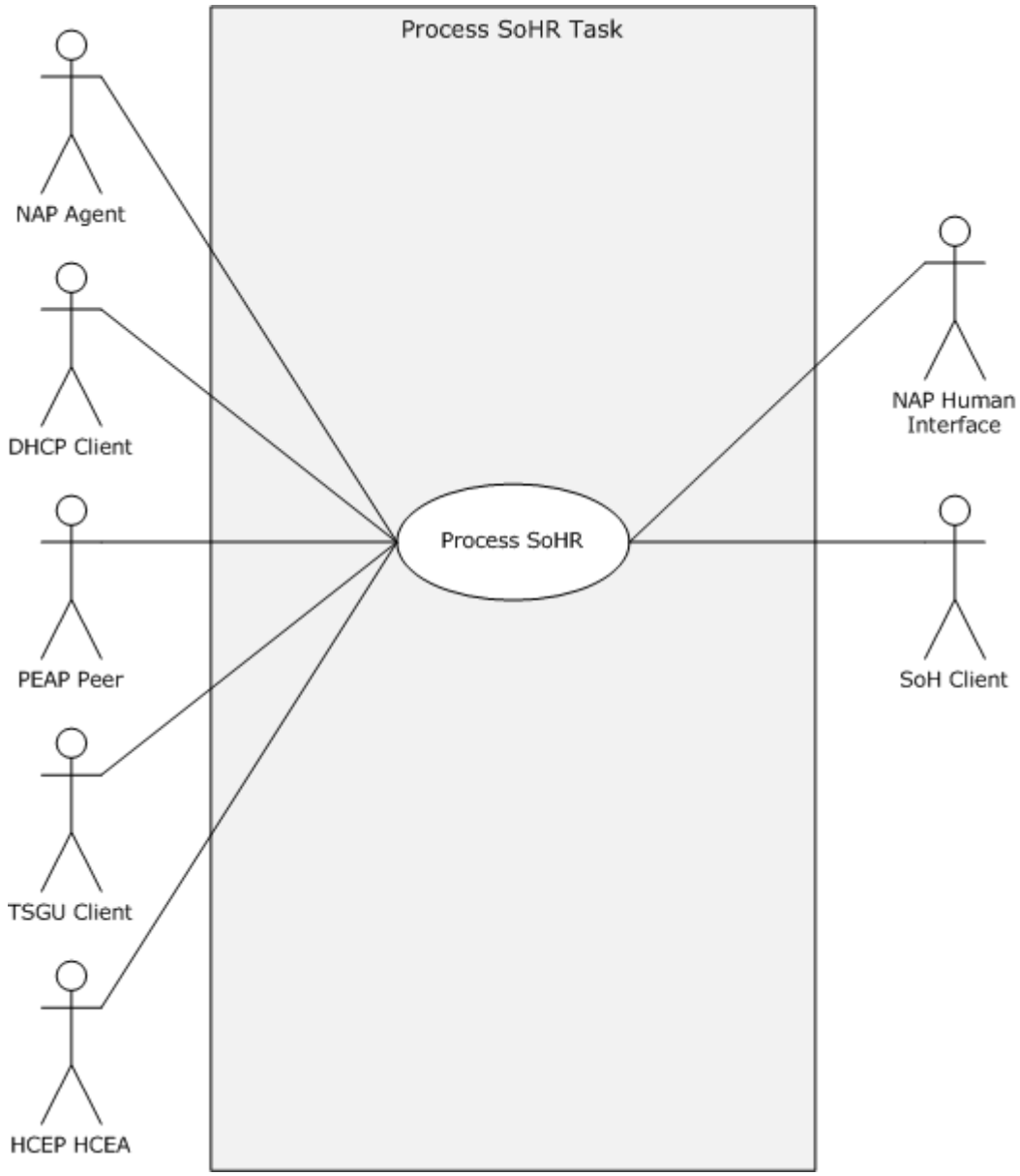

Figure 59: Process SoHR Task use case diagram

14.1.3.4 Use Case: Process SoHR - NAP Agent

This use case is associated with the use case diagram in section 14.1.3.3.

Goal: To evaluate the validity of SoHR messages and trigger automatic remediation if the client computer is noncompliant.

Context of Use: This use case is initiated when an SoHR has been received by the SoH client.

Direct Actor: The direct actor in this use case is the NAP agent.

Primary Actor: The primary actor is the NAP agent.

Supporting Actors: The supporting actors are as specified in section 14.1.3.2.

Stakeholders and Interests: All stakeholders are as follows:

- **DHCP client:** This protocol client is used to send DHCP Extensions for NAP messages, as specified in [MS-DHCPN], to a DHCP server on the PEP computer. When DHCP enforcement is used, it acts as an enforcement client (EC) in this use case. Unlike other ECs, where enforcement occurs before this task and is protocol-specific, to achieve enforcement in the case of IPv6, the DHCP client requires that health status information about the computer is obtained and the IPv6 remediation servers transport the SoHR as specified in [MS-DHCPN] section 3.1.5.2. The purpose of this actor in the task is in analyzing the validity of the SoHR.
- **Remediate Client Health Task:** The Remediate Client Health Task performs remediation of noncompliant components on the client computer, as necessary. Each execution of the Remediate Client Health Task receives an **SoHRReportEntry** as specified in [TNC-IF-TNCCSPBSoH] extracted from the SoHR message by the **Process SoHR Task**. The purpose of the Remediate Client Health Task in the Process SoHR Task is to obtain only valid SoHR data received in the transport messages.

Preconditions:

- The SoH client is deployed and configured correctly.
- The interface between the NAP EC and the SoH client is functioning correctly.
- The SoHR message received is valid according to [TNC-IF-TNCCSPBSoH].

Minimal Guarantees:

- The use case processes SoHR messages which are syntactically valid and the correlation ID corresponds to a sent SoH.
- The task does not alter the SoHR and filters out invalid SoHRs.

Success Guarantee: The SoH client triggers the Remediate Client Health Task (section 15) if the client is determined to be noncompliant in the SoHR and remediation is required. The EC regains control when enforcement based on the content of valid SoHRs is required, such as in the case of DHCP clients.

Trigger: The trigger is the invoking of the NAP agent by the EC.

Main Success Scenario:

1. The NAP agent is invoked by the EC.
2. The SoH client receives an SoHR message from the NAP agent.
3. The SoH client ensures that the SoHR message is syntactically correct and corresponds to a request created by the client by matching the unique correlation ID. If the SoHR is not syntactically correct or the correlation ID does not match, the SoH client ignores the SoHR and no additional processing is performed.
4. The SoH client detects the requirement for remediation and yields control to the NAP agent.
5. The NAP agent requests the NAP Human Interface to display the health status.
6. The NAP agent triggers the Remediate Client Health Task as required.
7. The NAP agent returns control to the EC.

14.2 Task Context

This section describes the relationship between this task and its environment.

14.2.1 Task Environment

This task is accomplished by the NAP client in an environment wherein client users log on to a client computer to request access to network resources under the control of devices or servers acting as PEPs ([RFC2753]). The environment should meet the following requirement to support this task.

- **Requirement:** The SoH client, NAP agent, and ECs are correctly configured as specified in Common Abstract Data Model (section 4.1.1).
  - **Reason for requirement:** Correct configuration of these components is required for the SoH client to compute remediation requirements based on the received SoHR.
  - **Satisfying the requirement:** The Update NAP Client Configuration Task (section 5) completes successfully.
  - **Verifying requirement is satisfied:**
    1. No errors related to configuration are logged by the SoH client, NAP agent, and the EC.
    2. If the DHCP client is enabled, IPv6 traffic will always be blocked regardless of the health results.
  - **Consequences of not satisfying requirement:** The task is unable to process the SoHR.

14.2.2 Task Relationships

14.2.2.1 Black-Box Relationship Diagrams

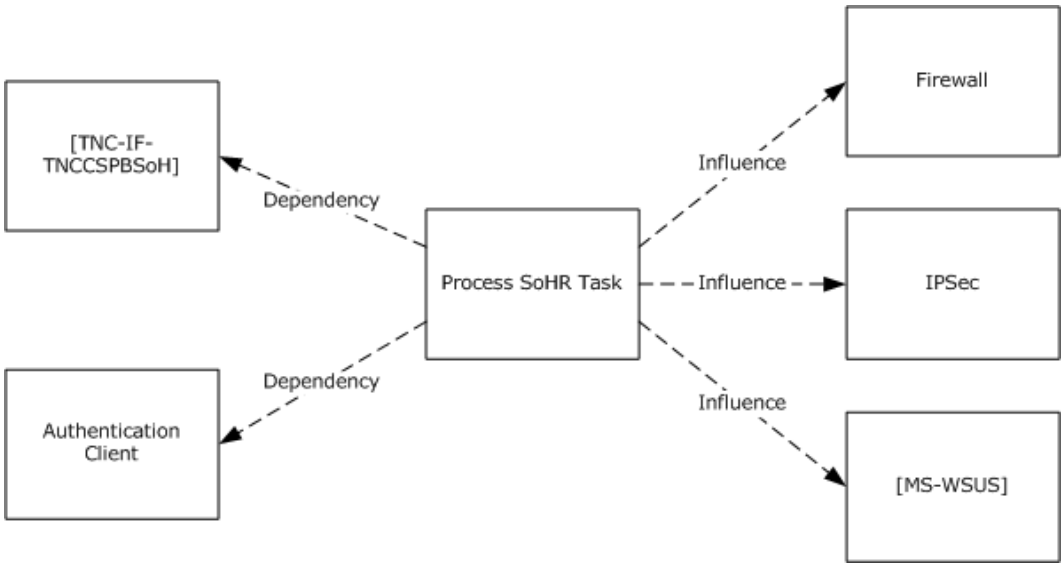

Figure 60: Process SoHR Task black-box relationships

14.2.2.2 Task Dependencies

The Process SoHR Task (section 14) depends on the Receive SoHR Task (section 13) because the SoHR is only processed after it is received.

As shown in the diagram in section 14.2.2.1, the Remediate Client Health Task potentially has a dependency on the Process SoHR Task. If the client is unhealthy, the Process SoHR Task will trigger this task. Also, the DHCP client endpoint depends on the Process SoHR Task to regain control with the validated SoHR to apply the IPv6 enforcement mechanism.

14.2.2.3 Task Influences

None.

14.2.3 Task Assumptions and Preconditions

To accomplish this task, the NAP client has the following preconditions and assumptions:

- The operating system and hardware comprising on the client computer is trustworthy.
- The client administrators are trustworthy. The client administrators are responsible for enabling and configuring the NAP client correctly. They are also responsible for the integrity of executable code that provides NAP client services.
- The NAP client is enabled and correctly configured by the client administrator.

14.2.4 Task Versioning and Capability Negotiation

The Process SoHR Task does not define any versioning and capability negotiation beyond those described in the specifications of the protocols supported or used by the task, as listed in section 2.3.

14.3 Task Architecture

This section describes the structure of the Process SoHR Task and the interrelationships among its parts.

14.3.1 Task Architectural Constraints

There should be only one instance of the Process SoHR Task on each client computer and this instance initializes itself each time it starts. Different instances of this task on different client computers can run independently. There are no constraints among these instances.

14.3.2 Task Abstract Data Model

This section describes the states that are established, used, and maintained by the processing rules of this task. State may be volatile or persisted and may pertain to one, some, or all instances of the task. The task's state consists of the values of the named data elements (also called state variables) presented in this section. The overall organization of the data elements, with their names, is the Abstract Data Model. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Data Model, such association can be achieved in other ways. Implementations can depart from this model so long as their external behavior remains consistent with that described in this document.

14.3.3 Task Abstract Parameters

This section describes data passed to an instance of this task at the time it is invoked or triggered. The parameters consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Parameter. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Parameters, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

SoHR: The task receives the SoHR as a parameter.

14.3.4 Task Abstract Results

This section describes data returned by an instance of this task to its caller. The results consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Result. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Results, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

The Process SoHR Task is expected to process the SoHR and indicate the validity of the SoHR. If the client is noncompliant, this task triggers the Remediate Client Health Task (section 15) based on the SoHR MS-Quarantine-State attribute and returns the **SoHRReportEntry** as specified in [TNC-IF-TNCCSPBSoH].

14.3.5 White-Box Relationships

The white box relationships for the Process SoHR Task are shown in the following figure.

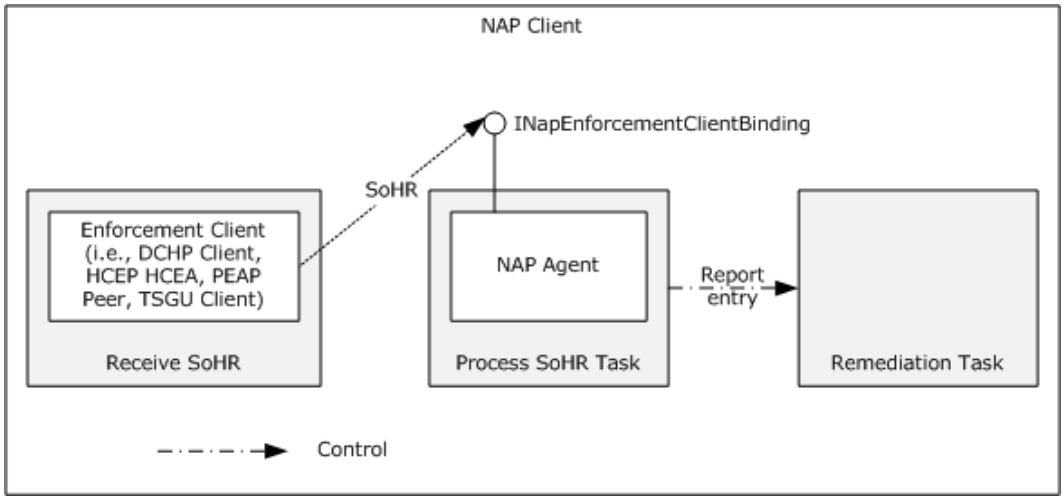

Figure 61: Process SoHR Task white-box relationships

After receiving the SoHR message, the NAP agent and SoH client process the SoHR following the format defined in the Protocol Bindings for SoH [TNC-IF-TNCCSPBSoH]. The NAP Human Interface, if started by the client user, collects health-related information from the NAP agent and displays it to the client user.

14.3.6 Task Events

14.3.6.1 Task Timers

None.

14.3.6.2 Task Non-Timer Events

This task does not use or respond to any additional non-timer events.

14.3.7 Task Architecture and Communication

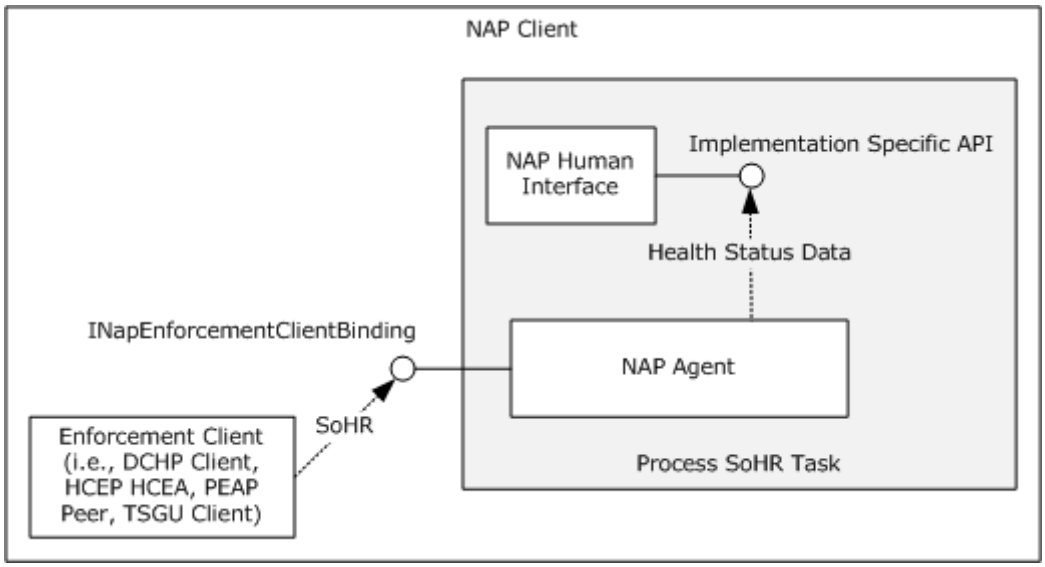

Figure 62: Process SoHR Task architecture and communication overview

14.3.8 Task Processing Rules

The following describes the operational flow of the Process SoHR Task:

1. The SoH client receives an SoHR message [TNC-IF-TNCCSPBSoH] from the NAP agent.
2. The SoH client ensures that the SoHR message is syntactically correct and corresponds to a request created by the client by matching the unique correlation ID. If the SoHR is not syntactically correct or the correlation ID does not match the **Correlation ID** ADM element (section 6.3.2) set by the Update NAP Client Configuration Task (section 5), the SoH client ignores the SoHR and no additional processing is performed. For more information, see section 14.4.5.
3. The SoH client extracts the **SoHRReportEntry** ([TNC-IF-TNCCSPBSoH]) from the SoHR message.
4. The SoH client checks for noncompliant SHV evaluation results in the SoHR as specified in [TNC-IF-TNCCSPBSoH].
5. For each noncompliant **SoHRReportEntry** ([TNC-IF-TNCCSPBSoH]), the SoH client determines whether remediation is required and yields control to the NAP agent. For more information about determining whether remediation is required, see section 14.4.5.
6. The NAP agent requests the NAP Human Interface to display the health status and triggers the Remediate Client Health Task (section 15) as required.
7. The NAP agent returns control to the EC that invoked this task.

14.3.9 Task Failure Scenarios

This task is executed by the NAP agent and SoH client, which are implemented in the same software component. The interface between the NAP agent and NAP Human Interface is implementation-

specific and SHOULD be designed in such a way that failures to execute its services do not compromise the execution of the NAP agent.

14.4 Task Details

This section contains the details that complete the descriptions in earlier sections of the document. These are needed to understand and implement this task.

14.4.1 Task Precondition Details

The NAP agent service is started and initialized correctly on the client computer.

For the complete list of task preconditions and assumptions, see section 14.2.3.

14.4.2 Task Initialization of External Entities

None.

14.4.3 Task Event Details

14.4.3.1 Task Timer Details

None.

14.4.3.2 Task Non-Timer Event Details

None.

14.4.4 Task Architectural Details

This section illustrates an example of a NAP client processing an SoHR. The client will utilize several NAP agent and SHA functions to accomplish the request.

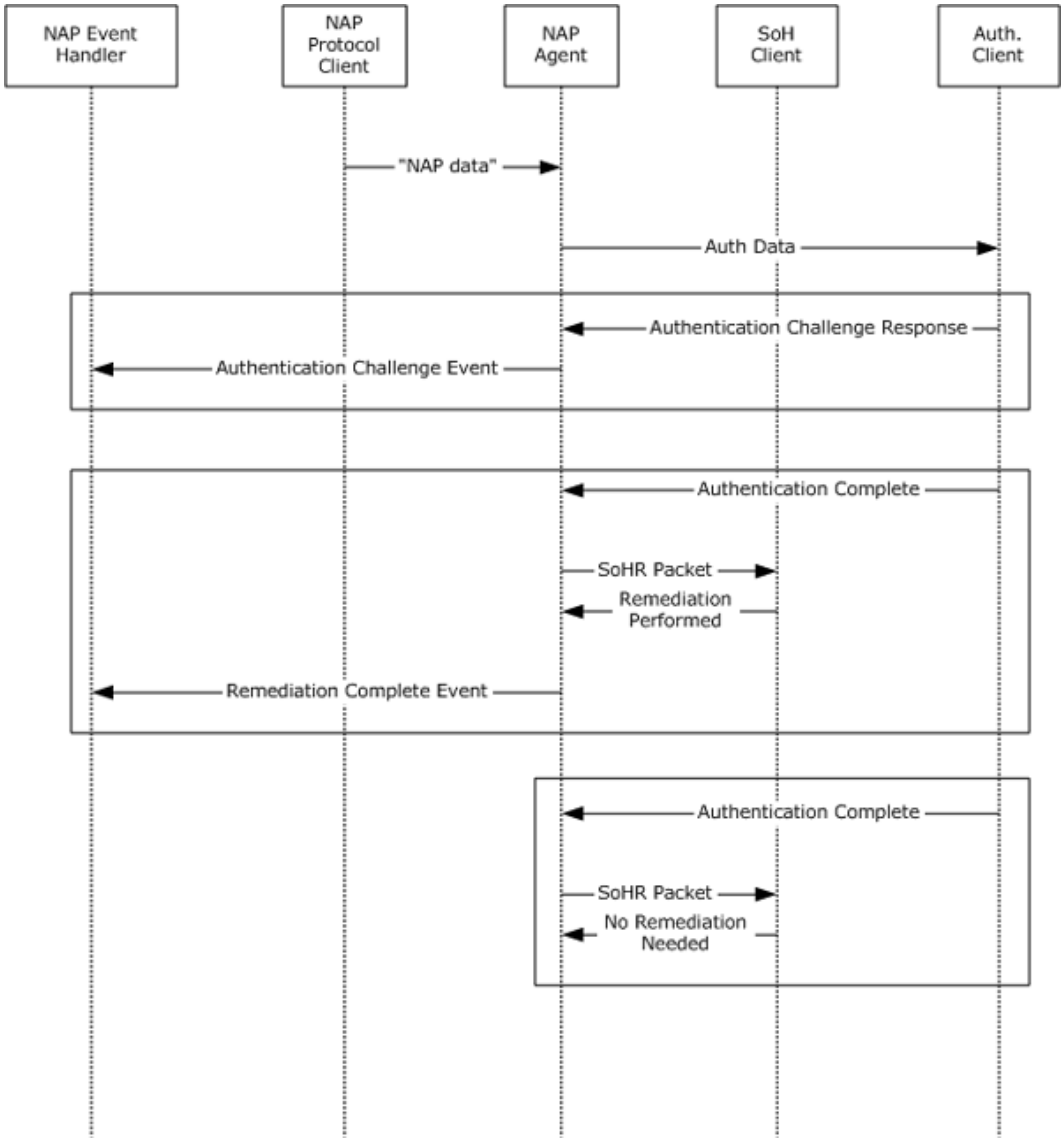

Figure 63: Sequence diagram for the main success scenario of the Process SoHR Task

The preceding diagram illustrates the task operational flow from EC enforcement to the NAP agent and then returning to the EC for use cases involving an invalid SoHR, compliant health status, noncompliant health status with remediation not required, and noncompliant health status with required remediation. For more information, see section 14.4.5.

14.4.5 Task Processing Rule Details

The following describes the operational details of the Process SoHR Task:

1. The NAP agent receives the SoHR from the EC via the **INapSystemHealthAgentCallback::ProcessSoHResponse** method, which is part of the SHA API. For more information about this API, see section 3.3.1 and [MSDN-NAPAPI].

2. The NAP agent internally invokes the SoH client to verify whether the SoHR is syntactically correct and corresponds to a request created by the client by matching the unique correlation ID. If the SoHR is not syntactically correct or the correlation ID does not match the **Correlation ID** ADM element (section 6.3.2) set by the Update NAP client Configuration Task (section 5), the SoH client ignores the SoHR and no additional processing is performed. For more information, see section 14.4.5. The SoH client also extracts the **SoHReportEntry** ([TNC-IF-TNCCSPBSoH]) from the SoHR message and verifies it as specified in [TNC-IF-TNCCSPBSoH] to continue processing.

3. The SoH client extracts and reviews the MS-Quarantine-State VSA to check the compliance state of the client.

4. If the client is compliant, processing ends and control is returned to the caller. If the client is noncompliant, the SoH client determines whether remediation is required by reviewing the f bit in the MS-Quarantine-State message of the SoHR, as described in [TNC-IF-TNCCSPBSoH].

5. If remediation is required, the SoHR returns control to the NAP agent which initiates the Remediate Client Health Task (section 15).

6. The NAP agent returns control to EC that invoked the agent. Processing continues based on how the NAP agent was invoked:

   - If the NAP agent was invoked from the DHCPN client, the DHCPN client performs ulterior, implementation-specific enforcement for IPv6 traffic, as specified in [MS-DHCPN] section 3.1.5.2, based on the value of the MS-Quarantine-State VSA and the list of **IPv6-Fixup-Servers** as specified in [TNC-IF-TNCCSPBSoH].

   - If the NAP agent was invoked from the HCEP HCEA, and the received HCEP message contained a PKCS#7 certificate as specified in section 13.4.5, the HCEP HCEA examines the registry value of PlumbIpsecPolicy ([MS-GPNAP] section 2.3.3). If the value of this setting is 0x00000001, the HCEP HCEA plumbs a prescanned IPSec Policy by using the Windows Filtering Platform Management API described in [MSDN-MGMTFUNCS]. This setting does not affect wire behavior of any of the described protocols.

14.5 Task Security

There are no task-specific security considerations. For additional information about security considerations, see section 16, as well as the Security sections of the referenced protocol Technical Documents.

15 Remediate Client Health Task

This section describes the Remediate Client Health Task. This task is performed on the NAP client. This task is expected to be used by the client computer and the client administrator.

Note This task uses the **ShaTimeoutInMsec** ADM elements (section 4.1.1). All other common information defined in section 4 is not applicable to this task.

15.1 Task Overview

15.1.1 Task Purpose

The purpose of this task is to remediate the client computer based on the health evaluation results. The exact remediation steps depend on the specific SHA/SHV that is used to monitor, validate, and correct system health.

A NAP client creates an SoH for health validation. This SoH is transported to the PDP for evaluation. The PDP uses the installed SHVs and the configured policies to determine whether the NAP client is compliant and if it is not, the PDP determines the remediation actions that must be taken to achieve compliance.

The PDP creates an SoHR, which indicates whether the NAP client is compliant or noncompliant and includes remediation steps that the client needs to take to correct the situation. The PDP passes the SoHR back to the NAP client through a PEP channel. The NAP client uses the information in SoHR to remediate its health state and create an updated SoH, and the health validation process begins again.

15.1.2 Task Applicability

This task is used whenever a NAP health evaluation results in an unhealthy or non-compliant client. After the PDP processes the contents of the SoH against the configured health requirement policies, it creates and sends the SoHR back to the NAP client. If the client is non-compliant, the SoHR contains the steps to remediate the client.

NAP health evaluations can occur at the initial connection to the network or network resources periodically, when network state changes, or when an element of system health that is being monitored by SHAs running on the NAP client changes.

15.1.3 Task Use Cases

15.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Remediate Client Health Task are as follows:

NAP agent: The main software component on the NAP client computer. It maintains the current health state information of the NAP client and facilitates communication between the NAP EC and the SoH Client. The ability to perform its services is where the NAP agent's interests in this task are.

Client Administrator: The individual who configures and administers the client computer. The Client Administrator ensures that the NAP client is remediated successfully and has full access to the network. The primary interest of the Client Administrator is to obtain the required remediation steps from the NAP Human Interface.

Create and Send SOH Task: The purpose of the Create and Send SoH Task is to ensure that the health information is correctly gathered and that the SoH is correctly created on a client computer

when a health state change occurs. Therefore, the Create and Send SoH Task has to be assured that the NAP agent will be notified regarding a status change caused by the remediation.

15.1.3.2 Supporting Actors and Task Interests Summary

SoH Client: The purpose of this actor is to utilize the processing rules defined by the Statement of Health for NAP Protocol [TNC-IF-TNCCSPBSoH] to perform remediation actions. The SoH Client is responsible for interpreting the remediation information found in an SoHR and for informing the NAP agent about the remediation results. The use case employs this actor whenever a remediation is required based on information found in an SoHR packet.

NAP Human Interface: The purpose of this actor is to guide the Client Administrator by displaying the remediation steps. The remediation steps have to be followed in order to attain compliance. The NAP Human Interface informs the administrator about the result of automatic remediation, successful or unsuccessful. In the case of unsuccessful remediation, the NAP Human Interface informs the Client Administrator about the unsuccessful remediation and provides instructions for the manual steps required for automatic remediation to be possible or to manually remediate the state of the device. When no remediation is required, the NAP Human Interface informs the Client Administrator that the state of the device is healthy. The use case employs this actor when a manual intervention is required to remediate the NAP client.

15.1.3.3 Use Case Diagrams

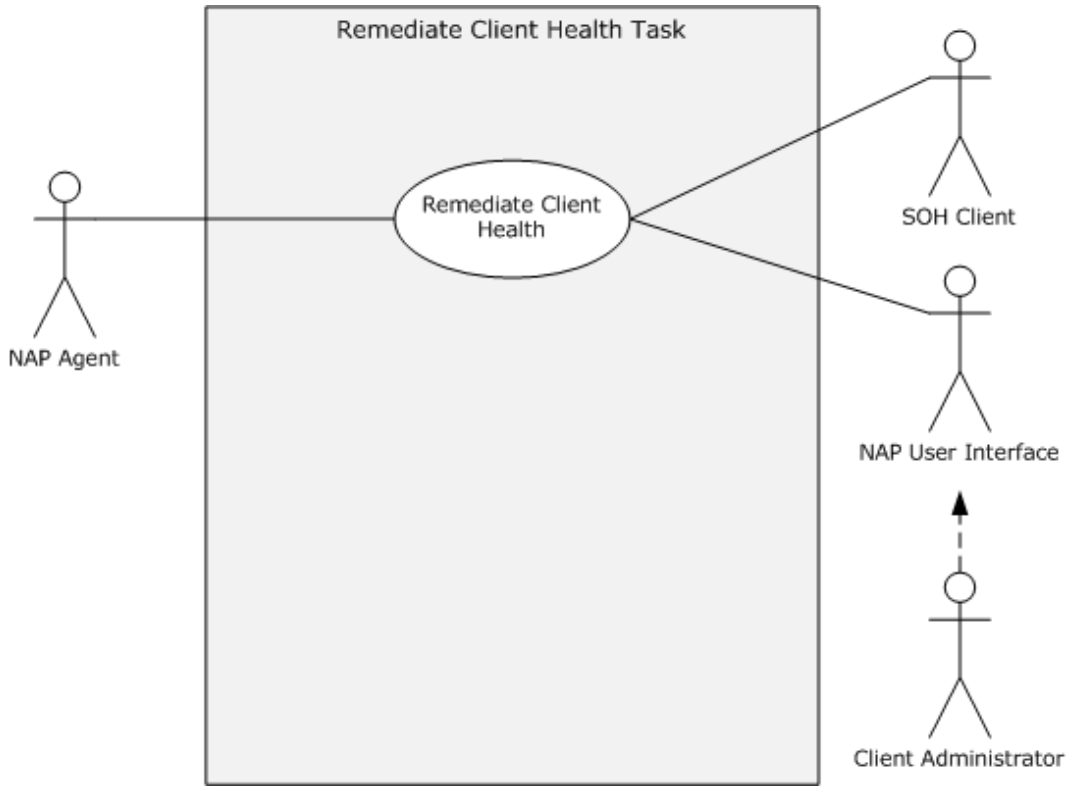

Figure 64: Remediate Client Health Task use case diagram

Both Manual Remediation and Automatic Remediation use cases extend the remediation of the client health by using the SHA's functionality of remediation. The exact method of remediation depends on the specifics of the SHA/SHV.

15.1.3.4 Use Case: Client Remediation – NAP Agent

This use case is associated with the use case diagram in section 15.1.3.3.

Goal: To remediate the client based on the health evaluation results found in the SoHR.

Context of Use: This use case is employed when the client is noncompliant and therefore requires remediation.

Direct Actor: The direct actor in this use case is the NAP agent. The NAP agent is the main software component on the NAP client computer. It maintains the current health state information of the NAP client and facilitates communication between the NAP EC and the SoH Client. The main interest of the NAP agent in this task is the ability to perform its services.

Primary Actor: The primary actor is the same as the direct actor.

Supporting Actors: The supporting actors are as defined in section 15.1.3.2.

Stakeholders and Interests:

- **Client Administrator:** The individual who configures and administers the client computer. The client administrator ensures that the NAP client is remediated successfully and has full access to the network. The primary interest of the Client Administrator is to obtain the required remediation steps from the NAP Human Interface.
- **Create and Send SOH Task:** The purpose of the Create and Send SoH Task is to ensure that the health information is correctly gathered and that the SoH is correctly created on a client computer when a health state change occurs. Therefore, the Create and Send SoH Task has to be assured that the NAP agent will be notified regarding a status change caused by the remediation.

Preconditions: The SoHR indicates that the client is noncompliant as specified in [TNC-IF-TNCCSPBSoH].

Minimal Guarantees:

- A remediation attempt will be performed.
- The information displayed by the NAP Human Interface reflects the remediation steps.
- The use case will always notify the NAP agent about the status of the system after remediation.

Success Guarantee: The client computer will be remediated.

Trigger: This use case is triggered when the Process SoHR Task (section 14) identifies that remediation is required on the client.

Main Success Scenario:

1. The task is triggered by the Process SoHR Task (section 14).
2. The NAP agent passes the SoHR message to the SoH Client.
3. The SoH Client determines which remediation steps have to be performed.

4. If automatic remediation is enabled, the SoH Client tries to remediate the NAP client automatically. If automatic remediation is not enabled, or if automatic remediation fails, the SoH Client notifies the Client Administrator using the NAP Human Interface and guides the administrator to follow the steps to remediate the NAP client.

5. The SoH Client informs the NAP agent regarding the outcome of the remediation actions.

Extensions: None.

15.2 Task Context

This section describes the relationship between this task and its environment.

15.2.1 Task Environment

To accomplish this task, the NAP client requires the following from its environment:

- **Requirement:** The remediation servers, if any are required for remediation, are reachable by the clients in the restricted network to enable the remediation process.
  - **Reason for requirement:** Some remediation actions might require access to the remediation server.
  - **Satisfying the requirement:**
    1. The network interface of the client computer is configured to operate on the local subnet.
    2. The physical network path (network devices, Ethernet cables, and so on) between the local subnet and the remediation server is connected.
    3. All network devices between the local subnet and the remediation server are configured to allow packet flow between the two entities in the restricted network.
  - **Verifying that requirement is satisfied:** The client computer can successfully ping the remediation server in the restricted network.
  - **Consequences of not satisfying requirement:** If the task is unable to reach the remediation server, the client will not be remediated and will remain in a noncompliant state.
- **Requirement:** Client Administrator intervention might be required on the client computer.
  - **Reason for requirement:** If automatic remediation fails, the Client Administrator has to manually remediate the client by following the steps provided by the NAP Human Interface.
  - **Satisfying the requirement:** The Client Administrator has to follow the remediation steps provided by the NAP Human Interface and ensure that all the software and updates are installed on the client computer.
  - **Verifying that requirement is satisfied:** The client computer will be quarantined.
  - **Consequences of not satisfying requirement:** The client computer will not be remediated.

Unless explicitly specified otherwise, the task implementation may assume that its environment is properly configured and is not expected to verify that every requirement is satisfied. On the other hand, the task implementation should be able to gracefully handle errors which may be caused by environment misconfiguration or temporary dysfunction. The implementation should log these errors along with relevant information to allow troubleshooting.

15.2.2 Task Relationships

15.2.2.1 Black-Box Relationship Diagrams

This task consists of remediating the NAP client based on the SoHR. The following diagram illustrates the task to enable the functioning of the NAP System.

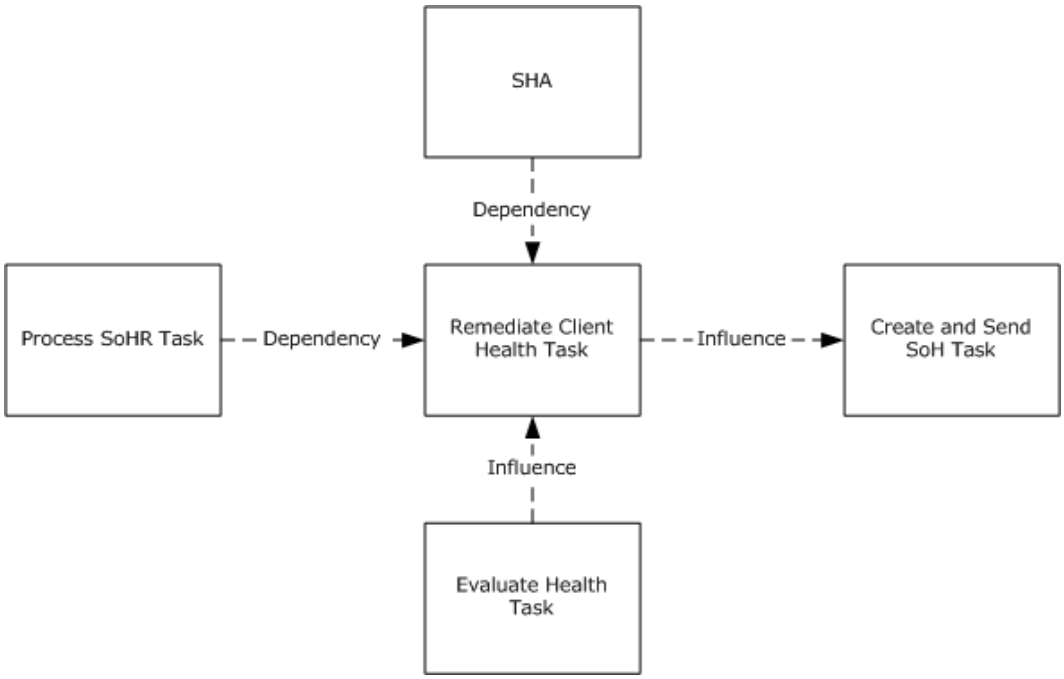

Figure 65: Remediate Client Health Task black-box diagram

A noncompliant client receives an SoHR that indicates whether the client should automatically attempt to remediate its noncompliant health state and also the steps needed to remediate.

15.2.2.2 Task Dependencies

The Remediate Client Health Task depends on the Process SoHR Task to provide evaluation results for remediation. This task also depends on the SoH Client to perform the remediation on the client computer.

15.2.2.3 Task Influences

The Create and Send SoHR Task (section 10) and Process SoH Task (section 9) influence the Remediate Client Health Task. The Process SoH Task determines the health of the client. Based on this decision, the client may have to be remediated. The Process SoH Task identifies the steps that are needed for the client to remediate itself.

The Remediate Client Health Task influences the Create and Send SoH Task (section 6) as the Remediate Client Health Task may change the client computer firewall, Windows Server Update Service, or antivirus or antispyware software, and as a result, trigger the Create and Send SoH Task.

15.2.3 Task Assumptions and Preconditions

To accomplish this task, the NAP client has the following preconditions and assumptions:

- The operating system and hardware comprising on the client computer is trustworthy.
- The client administrators are trustworthy. The client administrators are responsible for enabling and configuring NAP client settings correctly. They are also responsible for the integrity of the executable that provides NAP client services.
- The NAP client is enabled and correctly configured by the client administrator.
- The remediation servers are configured and are reachable for the clients in the restricted network.

15.2.4 Task Versioning and Capability Negotiation

The system does not define any versioning or capability negotiation beyond those described in the specifications of the protocols supported by the system.

15.3 Task Architecture

15.3.1 Task Architectural Constraints

None.

15.3.2 Task Abstract Data Model

This section describes state established, used, and maintained by processing rules of this task. State may be volatile or persisted. State may pertain to one, some, or all instances of the task. The task's state consists of the values of the named data elements (also called state variables) presented in this section. The overall organization of the data elements, with their names, is the Abstract Data Model. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Data Model, such association can be achieved in other ways. Implementations may depart from this model so long as their external behavior remains consistent with that described in this document.

Remediate client state: Client health remediation occurs when an SoHR has a NotHealthy evaluation result. This ADM element specifies whether client health remediation will be performed. A value of Active (0x00000001) indicates remediation, and a value of Inactive (0x00000000) indicates no remediation.

After the task is triggered, its state changes to Active and client health remediation is performed, which is implementation-specific for every SHA.

15.3.3 Task Abstract Parameters

This section describes data passed to an instance of this task at the time it is invoked or triggered. The parameters consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Parameter. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Parameters, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

Health evaluation result: The task receives the SoHRReportEntry ([TNC-IF-TNCCSPBSoH]) as a parameter.

15.3.4 Task Abstract Results

This section describes data returned by an instance of this task to its caller. The results consist of the values of the named data elements presented in this section. The organization of a data element, with its names, is an Abstract Result. It is intended to facilitate the reader's conceptual understanding of the specification. While a task's processing rules may depend upon associations established by the structure of its Abstract Results, such association can be achieved in other ways. Implementations may depart from this abstraction so long as their external behavior remains consistent with that described in this document.

The Remediate Client Health Task is expected to perform SoH Client remediation of its monitored resources.

15.3.5 White-Box Relationships

SoHR messages contain MS-Quarantine-State message to indicate the state of compliance of the client and includes remediation actions for the client. The NAP client uses the information in the SoHR to remediate its health state.

The "f bit" in the MS-Quarantine-State message of the SoHR, as described in [TNC-IF-TNCCSPBSoH], indicates to the NAP client that it should remediate. The SoH Client uses this flag along with the individual health response to calculate the actual required remediation actions.

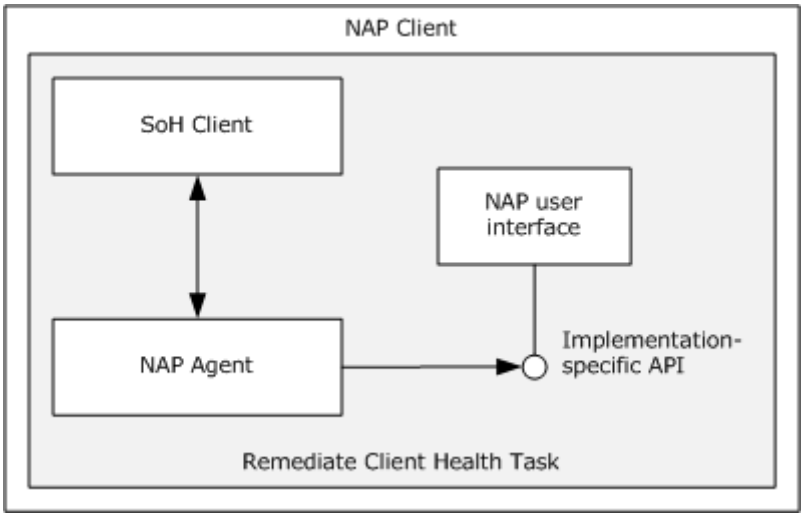

Figure 66: Remediate Client Health Task white-box diagram

15.3.6 Task Events

15.3.6.1 Task Timers

There are no additional timers on outside entities imposed by this task other than the timers in the underlying transport system. However, inside this task there is a timer associated with all function calls that the NAP agent changes into SHAs. This timer known as **ShaTimeoutInMsec** (see section 4.1.1) determines how soon these function calls must return.

15.3.6.2 Task Non-Timer Events

The system does not define any task non-timer events beyond those described in the specifications of the protocols supported by the system.

15.3.7 Task Architecture and Communication

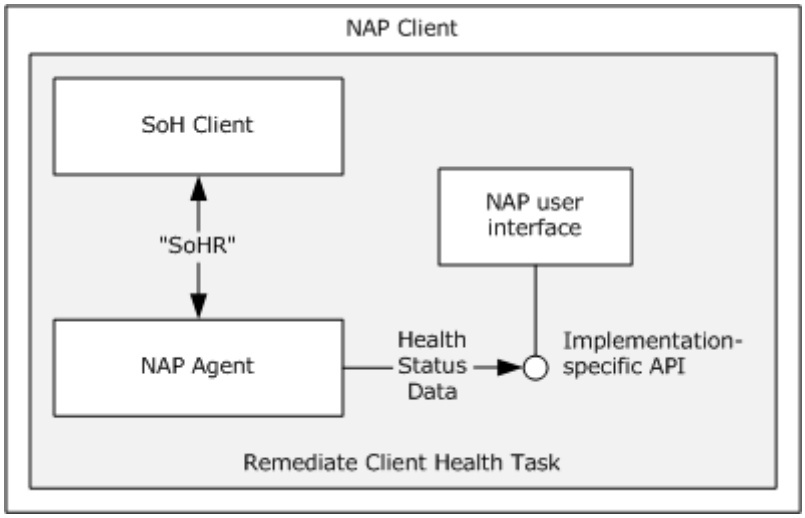

Figure 67: Remediate Client Health Task architecture and communication

The diagram shows the architectural details and the interworking between the SoH Client and an SHA-monitored resource component and/or an external remediation server to accomplish the Remediate Client Health Task.

The Remediate Client Health Task is indirectly triggered by the evaluation of health on the PDP. The PDP, consisting of the SHV and the Policy Engine, evaluates the health of the NAP client and creates the SoHR with the MS-Quarantine-State message as described in [TNC-IF-TNCCSPBSoH].

The Remediate Client Health Task is triggered on the NAP client upon the receipt of the SoHR if the SoHR indicates that the client is noncompliant and must remediate. The SoH Client receives and processes the SoHR. The SoH Client uses the "f bit" in the MS-Quarantine-State of the SoHR, as described in [TNC-IF-TNCCSPBSoH], to indicate to the SHAs that they should remediate. The SoH Client uses a flag along with the health response to communicate this to the SHAs and to initiate remediation.

15.3.8 Task Processing Rules

The following describes the operational flow of the Remediate Client Health Task:

1. The task is triggered by the Process SoHR Task (section 14).
2. The NAP agent passes the SoHR message to the SoH Client.
3. The SoH Client determines which remediation steps have to be performed.
4. The SoH Client changes state to active as described in section 15.3.2.
5. If automatic remediation is enabled, the SoH Client attempts to perform automatic remediation based on the health evaluation result codes, as specified in section 15.4.5.

6. If automatic remediation is not enabled or if automatic remediation fails, the SoH Client notifies the Client Administrator using the NAP Human Interface and guides the administrator to follow the steps to remediate the NAP client.

7. The SoH Client notifies the NAP agent about the status of remediation.

15.3.9 Task Failure Scenarios

15.3.9.1 Failures in SHA and SoH Client Communication with SHA

These failures are caused by an error with the initialization, registration, or binding of the SHA. The SoH Client relies on its ability to communicate with the installed SHAs in order to manage the health status that is monitored and reported by a SHA and to remediate the noncompliant client.

In this failure scenario, the SoH Client fails to communicate the health status of the properties that are monitored by the SHA and the remediation steps. The client experiencing this failure will not be able to initiate steps to remediate the client, which may keep the client noncompliant.

The failures are detected by a timer monitored by the SoH Client. The NAP System provides an error code enabling the administrator to configure fragility settings to detect and override the health policy decision on the PDP.

15.3.9.2 Failures in SHA and Remediation Server Communication

These failures can be caused by the following:

- Misconfiguration on the NAP client and/or PDP.
- Network connectivity issues wherein the NAP client cannot communicate with the remediation server.

If the NAP client cannot communicate with the remediation server, the client may not be able to remediate itself. These failures are not detected by the NAP System.

15.4 Task Details

This section contains the details that complete the descriptions in earlier sections of the document. These are needed to understand and implement this task.

15.4.1 Task Precondition Details

For the complete list of task preconditions and assumptions, see section 15.2.3. Details for some of the preconditions are as follows:

- The NAP agent service is started and initialized correctly on the client computer.
- SHAs are correctly configured, registered, and bound to the SoH Client so that the SoH Client has a complete SHA list.
- The Auto-Remediation settings are correctly configured on the NAP health policy server.

15.4.2 Task Initialization of External Entities

None.

15.4.3 Task Event Details

15.4.3.1 Task Timer Details

None.

15.4.3.2 Task Non-Timer Event Details

None.

15.4.4 Task Architectural Details

This section illustrates an example of a NAP client remediating itself. The client will utilize the SoH Client to accomplish the request.

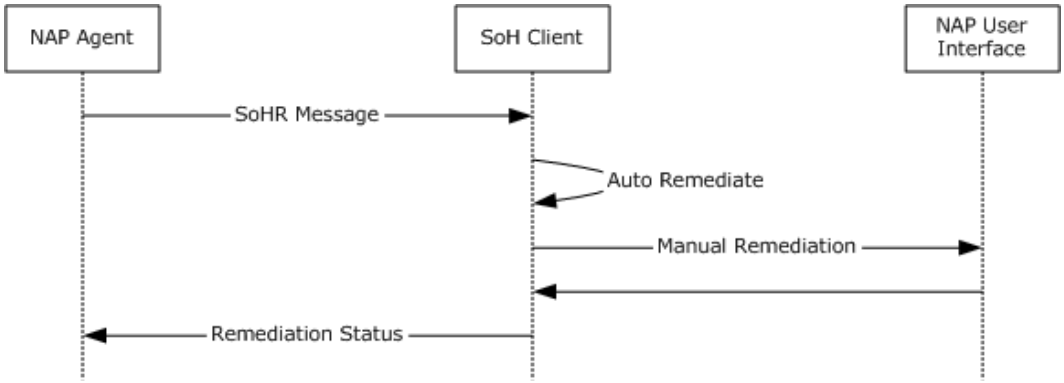

Figure 68: Sequence diagram for the main success scenario of the Remediate Client Health Task

1. The SoH Client receives the remediation information from the NAP agent.
2. The SoH Client attempts to remediate the client automatically. If automatic remediation fails, manual remediation has to be performed. (The SHA INapSoHProcessor API is called to start manual remediation. For a complete list of SHA APIs, see [MSDN-NAPAPI].)
3. The SoH Client notifies the NAP agent about the remediation status.

15.4.5 Task Processing Rule Details

The following describes the operational details of the Remediate Client Health Task:

1. The task is triggered by the Process SoHR Task (section 14).
2. The NAP agent passes the SoHR message to the SoH Client.
3. The SoH Client determines which remediation steps have to be performed.
4. The SoH Client sets the state of the **Remediate client state** ADM element (section 15.3.2) to active.
5. If automatic remediation is enabled (the 'f' bit of the SoHR message is set to 1), the SoH Client attempts automatic remediation as follows:

1. The SoH Client reviews the value of every **HealthClassID** field and checks its corresponding **Compliance-Result-Codes field** (see [TNC-IF-TNCCSPBSoH]).

2. If the value of the **Compliance-Result-Codes** field indicates that the Compliance state is NotCompliant, the SoH Client performs remediation based on the error code provided by the SHV. For a list of WSHA error codes, see [MS-WSH] sections 2.2.13, 2.2.14, and 3.1.5.

6. If automatic remediation is not enabled, or if automatic remediation fails, the SoH Client notifies the Client Administrator using the NAP Human Interface and guides the administrator to follow the steps to remediate the NAP client.

7. The SoH Client informs the NAP agent regarding the outcome of the remediation actions.

15.5 Task Security

There are no task-specific security considerations. For additional information about security considerations, see section 16, as well as the Security sections of the referenced protocol Technical Documents.

16 Security

This section documents security issues common to all tasks that are not otherwise described in the Technical Documents (TDs) for the protocols used in the task. It does not duplicate what is already in the protocol TDs unless there is some unique aspect that applies to the system as a whole.

The NAP System is designed to ensure that compliant (healthy) clients remain compliant and when full enforcement is used as specified in the Enforce NAP Policy Task in section 11, to ensure that noncompliant and possibly compromised clients cannot access and attack compliant clients.

This is illustrated in the following figure, where the client is shown communicating with the PDP for network access.

Figure 69: Message flow between the NAP client and the NAP health policy server

Enforcement in the case of an IPsec-protected client is as follows:

Protection of communication for IPsec-protected NAP clients is achieved by dropping incoming communication attempts that are sent from computers that cannot negotiate IPsec protection using health certificates. Unlike 802.1X and VPN enforcement, IPsec enforcement is performed by each individual computer, rather than at the point of entry into the network.

The NAP/Client system does not provide any security mechanism against tampering, spoofing, and replay attacks of the SoH message [TNC-IF-TNCCSPBSoH] or its contents sent to the NAP health policy server (NPS). The NPS blindly trusts the SoH messages received on the PEP channel and has no means to verify the integrity of the SoH message.

17 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Windows XP operating system Service Pack 3 (SP3)
- Windows Vista operating system
- Windows Server 2008 operating system
- Windows 7 operating system
- Windows Server 2008 R2 operating system
- Windows 8 operating system
- Windows Server 2012 operating system
- Windows 8.1 operating system
- Windows Server 2012 R2 operating system

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

18 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

19 Index

D

E

F

G

I

L

N

R

S

T

U

V

W