[MS-NAPSO]:
Network Access Protection System Overview

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** Portions of this document were contributed by The Technical Committee (www.thetc.org) and are reproduced with permission. Other portions are covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft Open Specification Promise or the Community Promise. If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.
- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

This document provides an overview of the Network Access Protection 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	This document has been updated by The Technical

Date	Revision History	Revision Class	Comments
			Committee.

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 Microsoft Windows® client operating systems and selected Microsoft 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, Update NAP Client Configuration, describes how the configuration of the Network Access Protection client (NAP client) is updated.
- Section 5, Create and Send SoH Task, describes how the NAP Agent on the NAP Client creates the statement of health (SoH) when triggered by a system event or new NAP protocol connection, and how it sends it to the NAP transport protocol clients.
- Section 6, Proxy SoH Task, describes how the NAP Proxy on the policy enforcement point (NEP) sends NAP request data to the RNAP client.
- Section 7, Connect to NPS Task, describes how the Policy Engine on the NAP Health Policy Server receives NAP request data through either the Vendor-Specific RADIUS Attributes for Network Access Protection (NAP) Data Structure (RNAP) protocol server or the Compressed EAP Extension (CEAP) protocol server.
- Section 8, Process SoH Task, describes how the Policy Engine evaluates NAP request data.
- Section 9, Send SoHR Task, describes how the Policy Engine transmits the NAP response data through the RNAP or CEAP protocols.
- Section 10, Proxy SoHR Task, describes how the NAP Proxy sends NAP response data to the NAP transport protocol server for transport to the NAP Client.
- Section 11, Process SoHR Task, describes how the NAP Agent processes the SoHR message and the authentication response.
- Section 12, Security, describes system-wide security issues that are not otherwise described in the protocol documents related to the tasks covered in this document.

- Section 13, 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 (Client)
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-SOH]:

Health Certificate Enrollment Protocol (HCEP)
system health agent (SHA)
system health validator (SHV)

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

remediation
security updates

The following terms are specific to this document:

CEAP Client: A protocol client that implements a compressed version of EAP and includes Network Access Protection support, as specified in the [MS-CEAP] protocol.

CEAP Server: A protocol server that implements a compressed version of EAP and includes Network Access Protection support, as specified in the [MS-CEAP] protocol.

CEAP: See **Compressed Extensible Authentication Protocol (CEAP)**.

Compressed Extensible Authentication Protocol (CEAP): A compressed **Extensible Authentication Protocol (EAP)** extension that adds encryption services to the **EAP** methods, as specified in [MS-CEAP].

DHCPN Client: A DHCP client that supports Network Access Protection, as specified in the [MS-DHCPN] protocol.

DHCPN Server: A DHCP server that supports Network Access Protection, as specified in the [MS-DHCPN] protocol.

Dynamic Host Configuration Protocol (DHCP) Extensions for Network Access Protection (NAP) Protocol: A protocol that extends the DHCP protocol ([RFC2131]) to include support for Network Access Protection, as specified in [MS-DHCPN].

DHCPN: See **Dynamic Host Configuration Protocol (DHCP) Extensions for Network Access Protection (NAP) Protocol**.

EAPE Client: An EAP client that supports Network Access Protection, as specified in the [MS-EAPE] protocol.

EAPE Server: An EAP server that supports Network Access Protection, as specified in the [MS-EAPE] protocol.

EAPE: See **EAP Extension Protocol (EAPE)**.

EAP Extension Protocol (EAPE): An extension of the **Extensible Authentication Protocol (EAP)** [RFC3748] that include support for Network Access Protection, as specified in [MS-EAPE].

HCEP Client: A HTTP client that supports Network Access Protection and X.509 certificate processing, as specified in the [MS-HCEP] protocol.

HCEP Server: A HTTP server that supports Network Access Protection and X.509 certificate retrieval, as specified in the [MS-HCEP] protocol.

Health Certificate Enrollment Protocol (HCEP): A protocol that extends the HTTP protocol ([RFC2616]) to include support for Network Access Protection, as specified in [MS-HCEP].

HCEP: See **Health Certificate Enrollment Protocol (HCEP)**.

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 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.

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: The main software component on the **NAP Client**. 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.

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

NAP Enforcement Point (NEP): A computer acting as a server that enforces Network Access Protection. Examples of NEPs are VPN Servers, DHCP Servers, Terminal Services Gateways, 802.1x Routers and Health Registration Authority Servers.

NAP Health Policy Server (NPS): A computer acting as a server that stores **health** requirement policies and provides **health state** validation for NAP clients.

NAP Proxy: The main software component on the **NAP Enforcement Point (NEP)**. It is responsible for transferring data from the NAP transport protocol ([MS-DHCPN], [MS-HCEP], [MS-TSGU] and [MS-EAPE]) servers to the [MS-RNAP] client, or vice-versa.

NAP Transport Protocol: One of the NAP capable protocols that transport NAP information between the **NAP Client** and the **NAP Enforcement Point (NEP)**. These protocols consist of [MS-DHCPN], [MS-HCEP], [MS-TSGU] and [MS-CEAP]/[MS-EAPE].

Network Access Protection client (NAP client): The **NAP client** is the set of NAP components installed and running on a **NAP Client**. 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 **NAP Client**, composing the **health** information into a **SoH** [MS-SOH], and sending the **SoH** to a **NEP**.

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.

Policy Engine: The main software component on the **NAP Health Policy Server (NPS)**. It is responsible for executing the Connection, Network and Health policies, as well as sending the SoH message to the [MS-SOH] protocol server for validation.

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)**.

Terminal Services Gateway (TSG) client: A client that facilitates the access of authorized users of remote computers on the private network accessible via Internet, using the [MS-TSGU] protocol.

Terminal Services Gateway (TSG) server: A server that allows authorized remote users from the Internet to connect to resources on a private network via the [MS-TSGU] protocol.

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**.

TSGU Client: See **Terminal Services Gateway (TSG) client**.

TSGU Server: See **Terminal Services Gateway (TSG) server**.

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 **NAP Clients** 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 Specification documents 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.

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. Please check the archive site, http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624, as an additional source.

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

[MS-CEAP] Microsoft Corporation, "Compressed Extensible Authentication Protocol (CEAP) Specification".

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

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

[MS-EAPE] Microsoft Corporation, "EAP Extensions Protocol Specification".

[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".

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

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

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

[MS-SOH] Microsoft Corporation, "Statement of Health for Network Access Protection (NAP) Protocol Specification".

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

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

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

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

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

[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

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 Specification".

[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 computers 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 **NAP Agent** components to send and receive system health information.
- NAP Enforcement Points (NEPs), 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 (HCEP NEP), **Terminal Services Gateway server** (TSGU **NEP**), **virtual private network (VPN) server** and 802.1X capable devices using EAP Extension Protocol (EAPE NEP), and **DHCP** server (DHCP NEP).
- **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 HCEP). 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 8 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.

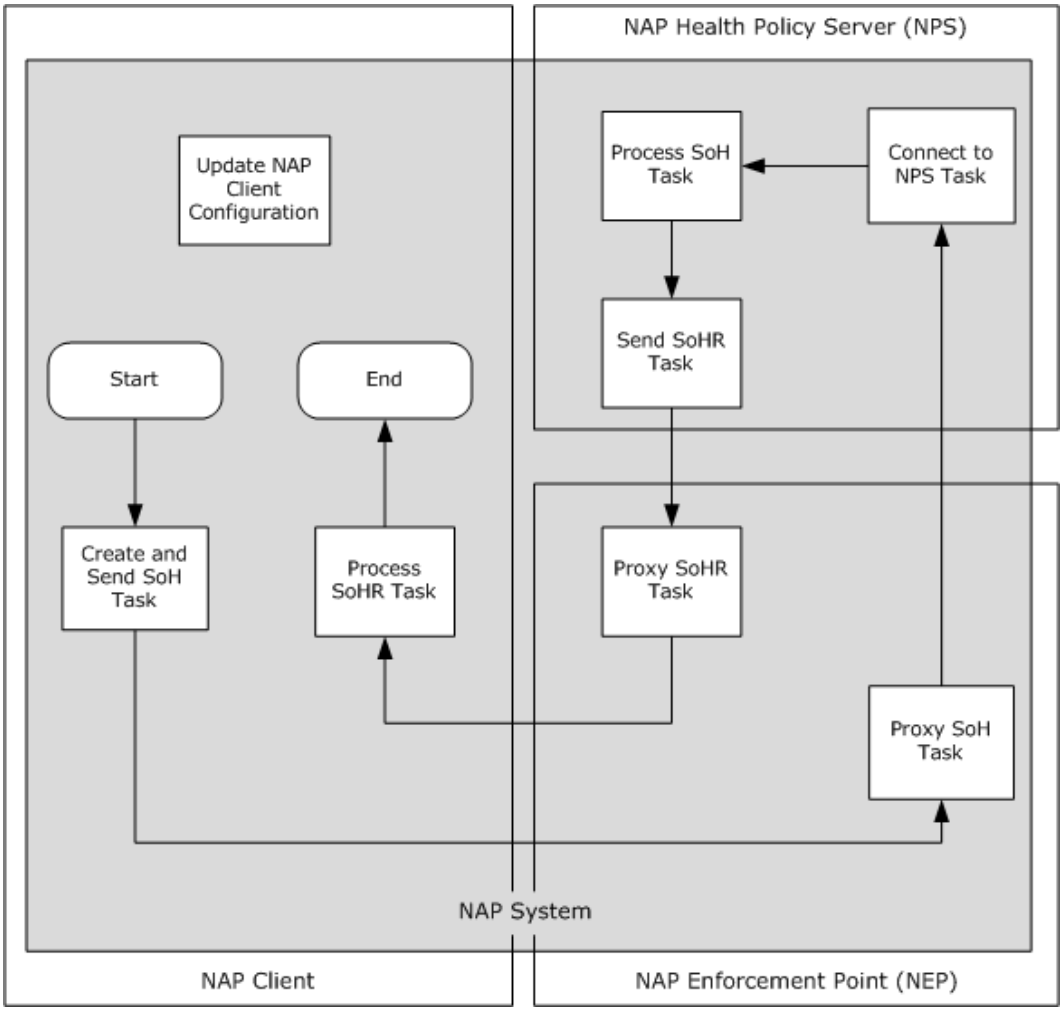

Figure 1: Process for health-validated access using eight defined tasks

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 (NEP)** if IPsec enforcement is used.

Create and Send SoH: This task describes the process of creating and sending the SoH on a NAP client computer.

Proxy SoH: This task describes the process of proxying the SoH via the PEP.

Connect to NHPS: This task describes the process of executing the connect policies on the NAP health policy server.

Process SoH: This task describes the process of evaluating the health and executing the network policy on the NAP health policy server.

Send SoHR: This task describes the process for sending the SoHR from the NAP health policy server.

Proxy SoHR: This task describes the process of proxying the SoHR via the PEP.

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

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 Microsoft Windows® operating system.

Statement of Health for Network Access Protection (NAP) Protocol Specification: As specified in [MS-SOH]. 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.

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

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

EAP Extension Protocol (PEAP) Specification: As specified in [MS-EAPE]. This protocol adds NAP support 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 Microsoft 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 **NAP Clients** 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 NAP Clients are compliant with health policy requirements.
- **Network access limitation:** Limits access for noncompliant network NAP Clients.
- **Automatic remediation:** Provides necessary updates to allow a noncompliant, network NAP Client to become compliant without user intervention.
- **Ongoing compliance:** Automatically updates compliant network NAP Clients 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 NAP Clients that are compliant with health policy.

The Microsoft 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 Microsoft 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-SOH] Statement of Health for Network Access Protection (NAP) Protocol Specification

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

- [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 NAP Clients 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.

3.3 NAP Architecture Details

This section contains information specific to the Microsoft Network Access Protection system.

3.3.1 NAP Client Architecture

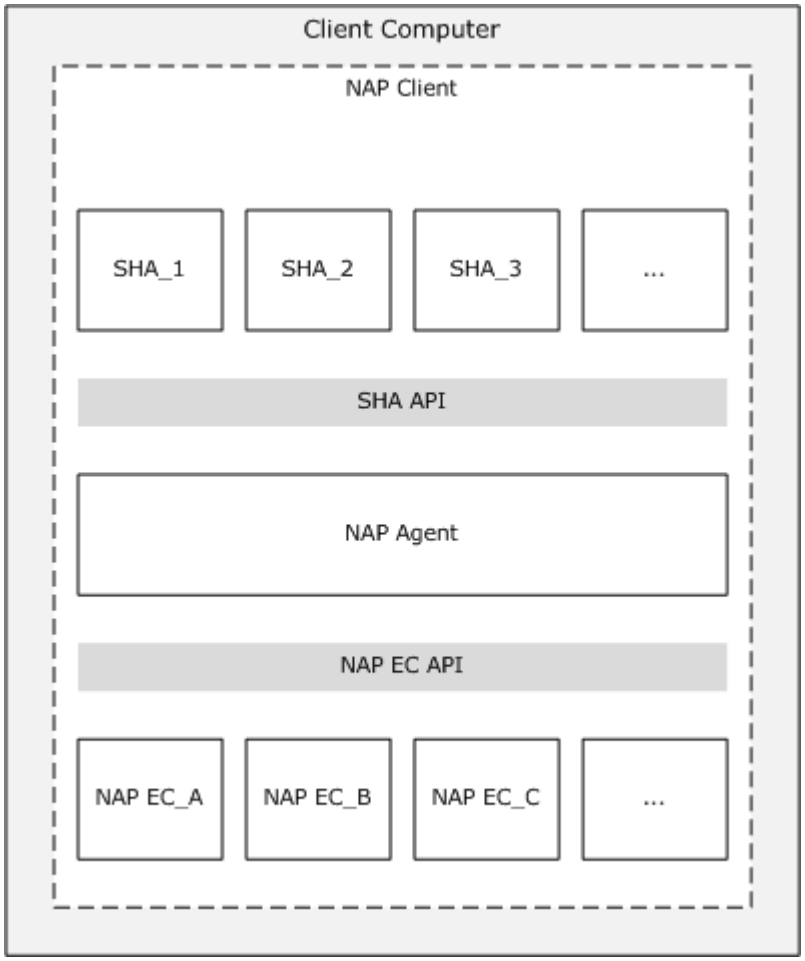

Figure 2: NAP client-side architecture

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), Microsoft 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 NAP Client 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

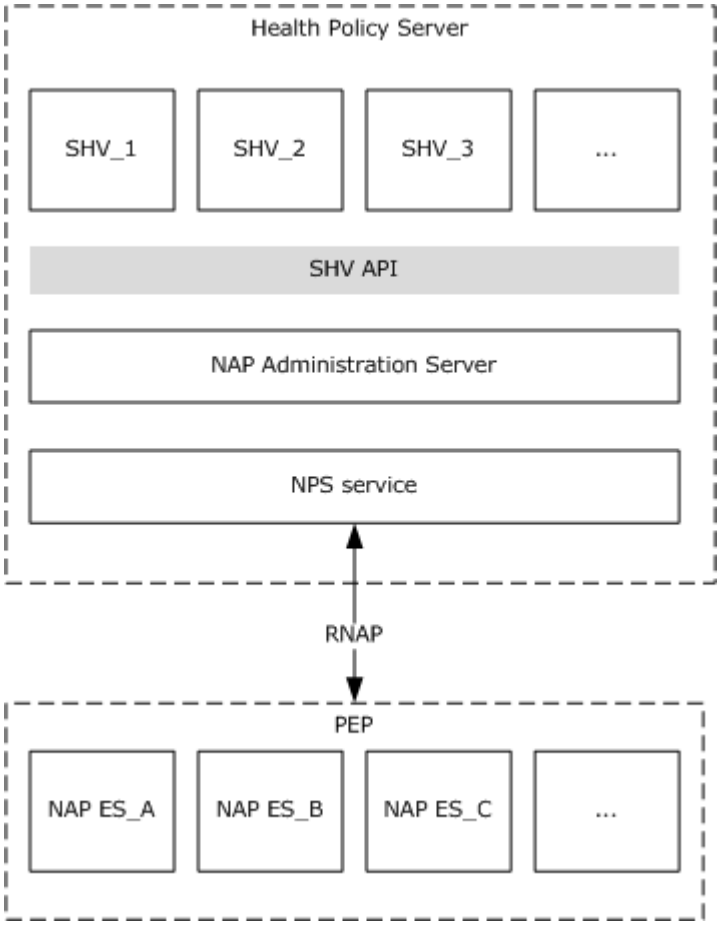

Figure 3: NAP server-side architecture

The NAP server-side architecture consists of NAP enforcement points and a NAP health policy server. A Microsoft 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

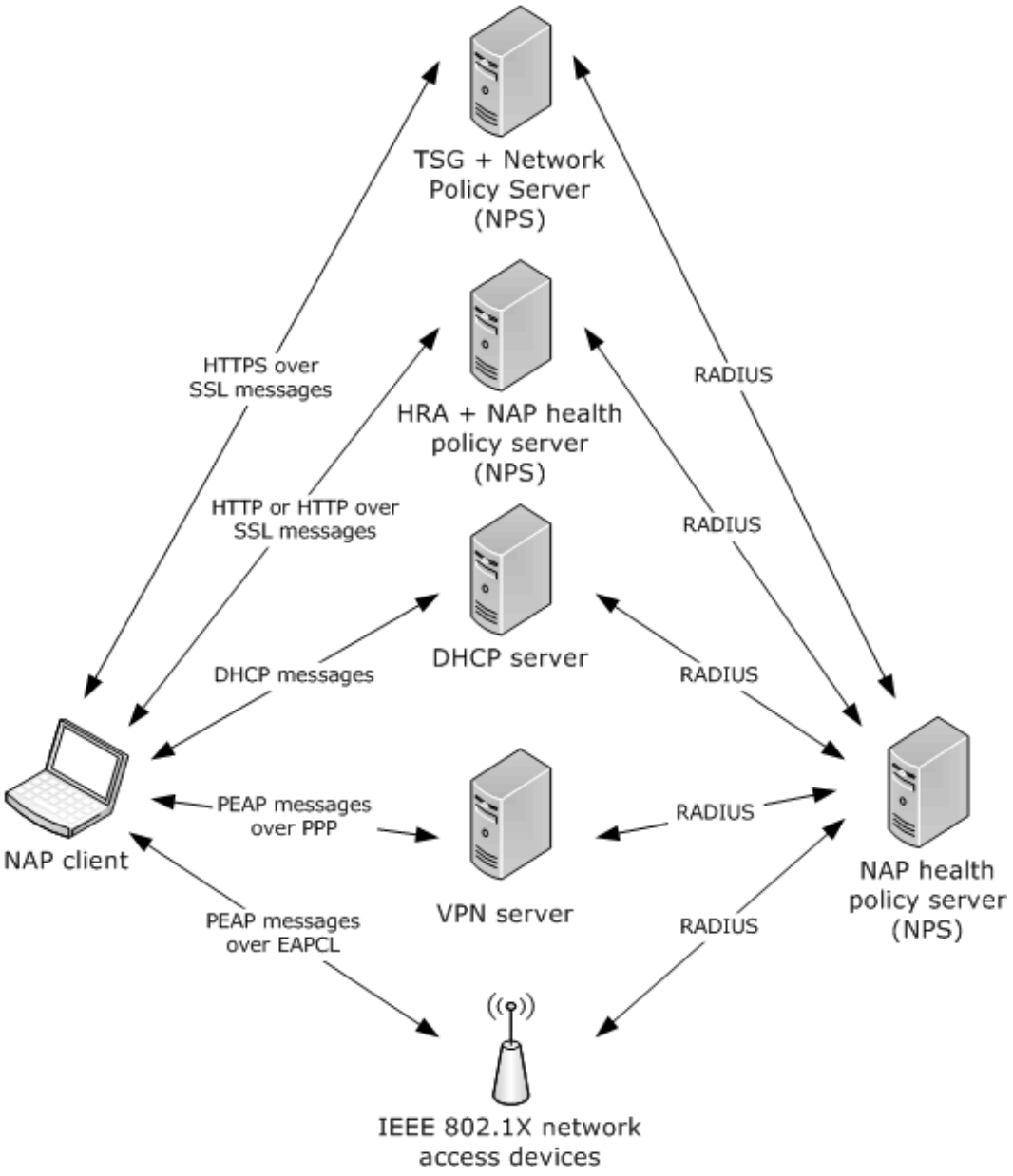

Figure 4: Interactions between NAP platform components

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 Microsoft 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 NEP.

- 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.

3.3.4 NAP System Architecture Details

The diagram below provides greater details of the components within Microsoft Windows® implementations of a Network Access Protection system.

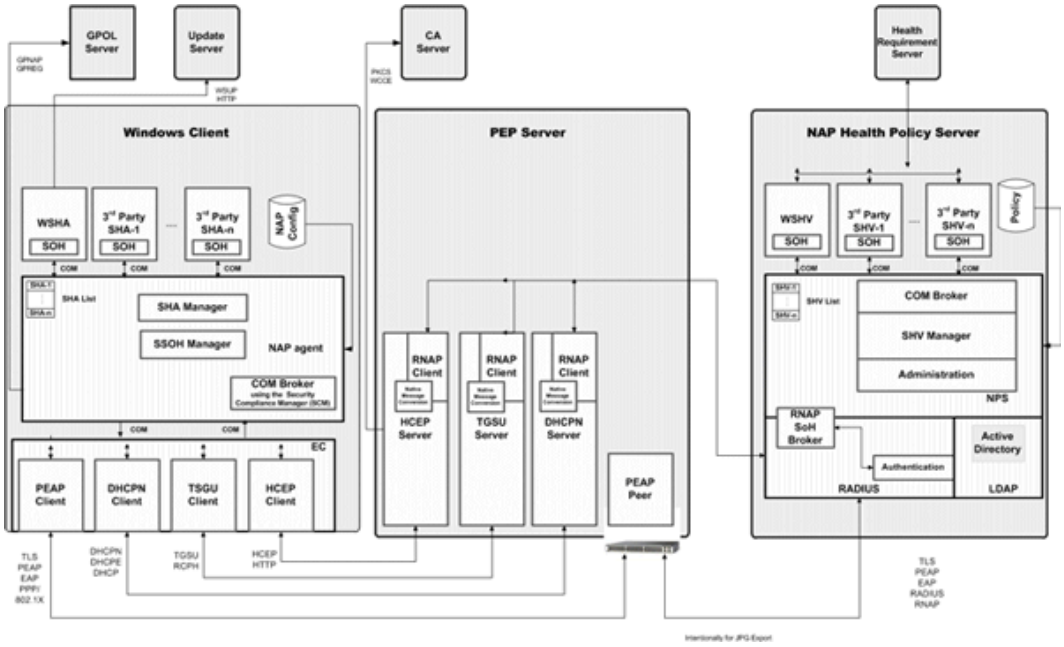

Figure 5: NAP System Detailed Architecture

3.4 Abstract Task Architectural Overview

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

3.4.1 NAP Client Architecture

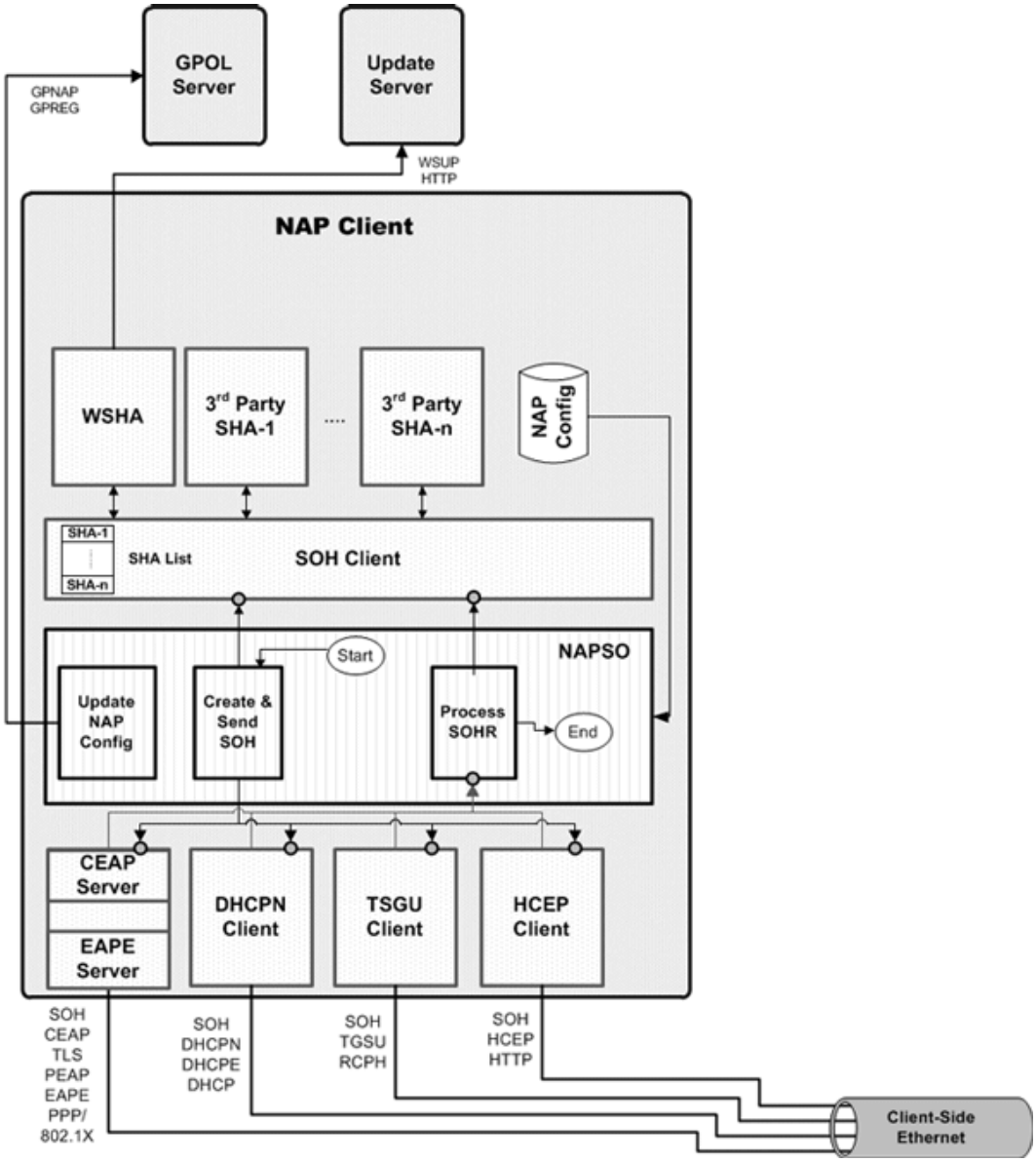

Figure 6: NAP Client Architecture

The NAP Client architecture consists of the NAP transport protocol clients, the NAPSO tasks, the SoH client, the SHA plug-ins (including WSHA) and the NAP Configuration store. There is also the GPOL server, which contains the group policy, and the update server, which contains MS Windows system and virus updates.

3.4.2 NAP Enforcement Point Architecture

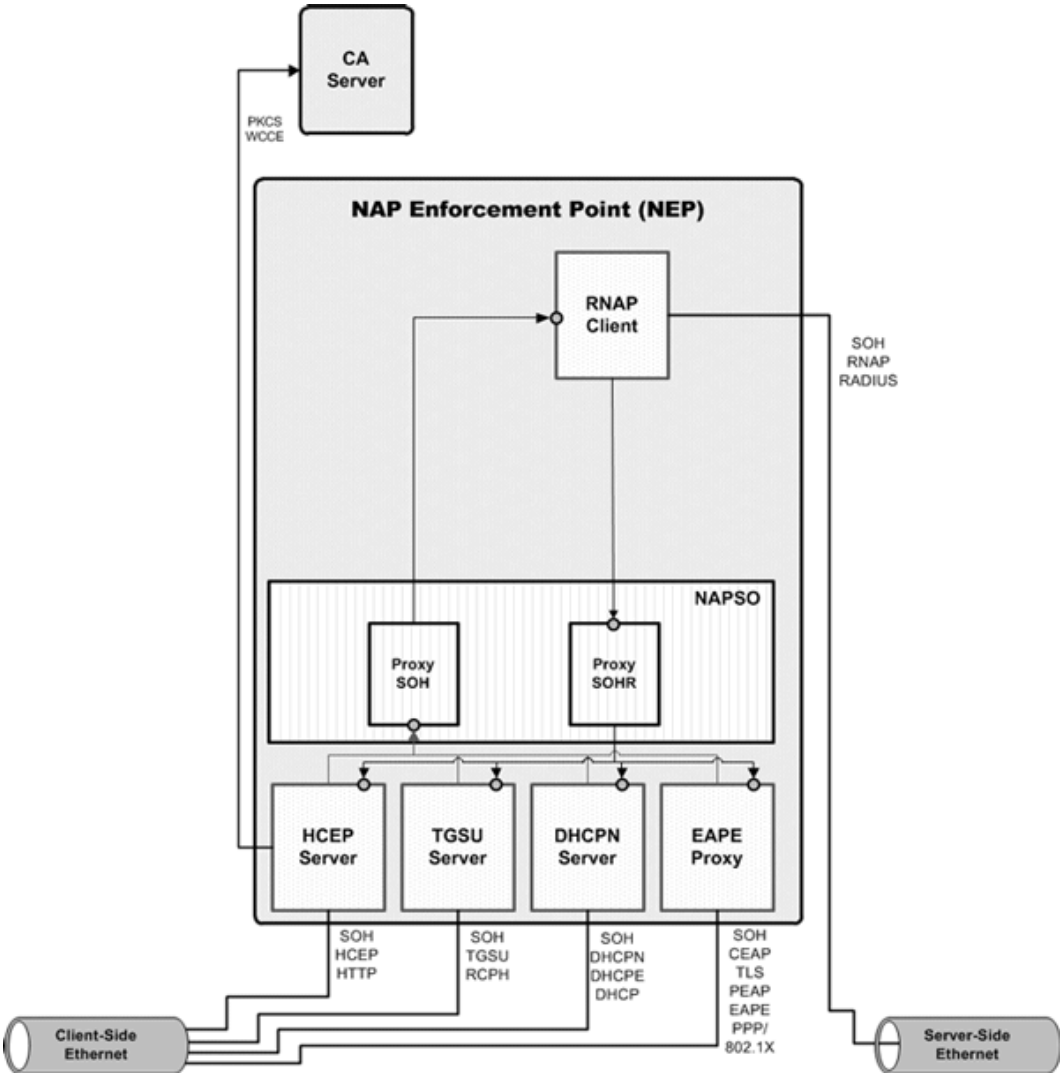

Figure 7: NAP Health Policy Server Architecture

The NAP Enforcement Point is a generic term to represent an amalgamation of the DHCP Server, the VPN Server, the HRA, the EAP Proxy and the 802.1x device. The NAP Enforcement Point consists of the NAP transport protocol servers, the NAPSO tasks and the RNAP client. There is also the CA server, which is used by the HCEP server to fetch certificates.

3.4.3 NAP Health Policy Server Architecture

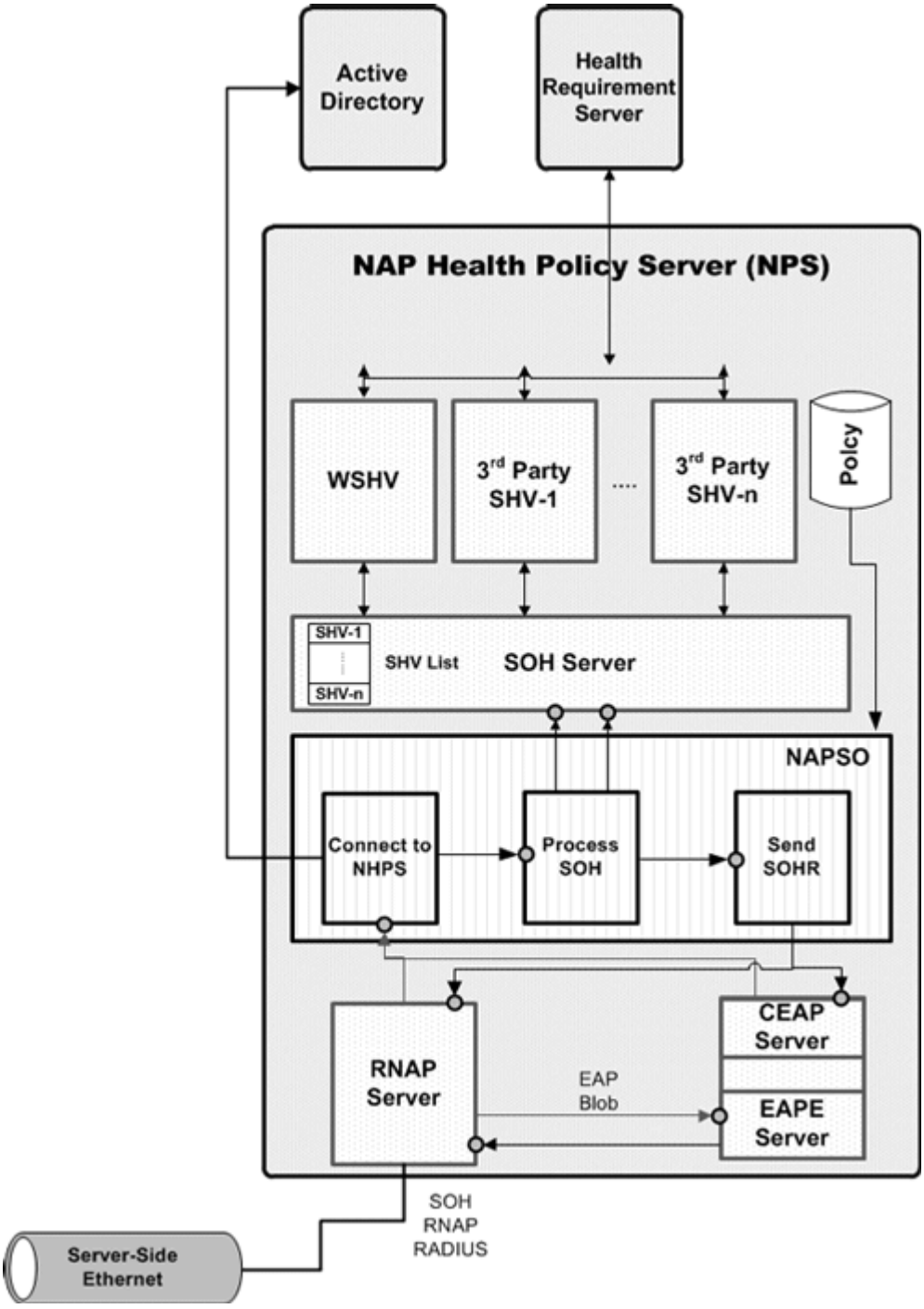

Figure 8: NAP Health Policy Server Architecture

The NAP Health Policy Server consists of the RNAP server, the CEAP/PEAP/EAPE server, the NAPSO tasks, the SoH server and the SHVs (including the WSHV). There is also the Active Directory server,

which performs authentication, and the Health Requirement Servers, which provides health requirement information to the SHVs.

4 Update NAP Client Configuration Task

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

4.1 Task Overview

4.1.1 Task Purpose

The purpose of this task is to update that the NAP configuration via the Group Policy System.

4.1.2 Task Applicability

This task is used when the NAP Client reboots, the NAP Client is first connected to the network or when the GPOL timer is triggered.

This task is not applicable if the NAP system is not deployed or the NAP Client is not joined to the domain.

4.1.3 Task Use Cases

4.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Update NAP Client Configuration Task (section 4) are:

GPOL Client: A component on the Group Policy system on the NAP Client that kicks off the fetching of GPO objects by group policy clients, such as GPNAP, GPIPSEC, GPSCMR, etc. Its interest in this task is that the GPNAP client will be invoked to fetch the GPO object related to NAP policy when triggered.

NAP agent: The main software component on the NAP Client. 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 interest of the NAP agent is to process the triggering event.

Create and Send SoH Task: The purpose of the Create and Send SoH Task (section 5) is to create SoH messages and send them, along with additional NAP data, to the NAP transport protocols, for eventual delivery to the NAP Health Policy Server. To accomplish this task, it uses the settings found in the NAP configuration. As such, its interests in this task are that valid NAP Configuration values are available at all times.

4.1.3.2 Supporting Actors and Task Interests Summary

NAP Configuration Manager: Manages the NAP Configuration store on the NAP Client. The NAP configuration (section 4.3.2 Task Abstract Data Model) contains elements representing all the settings found in [MS-GPNAP]. The task employs the NAP Configuration Manager to update values stored in the NAP Configuration.

GPNAP client: A Group Policy client that fetches NAP specific policy from the Group Policy System using [MS-GPREG], as described in [MS-GPNAP]. The task employs the GPNAP client to fetch NAP-specific policy settings from the **Group Policy server**.

4.1.3.3 Use Case Diagrams

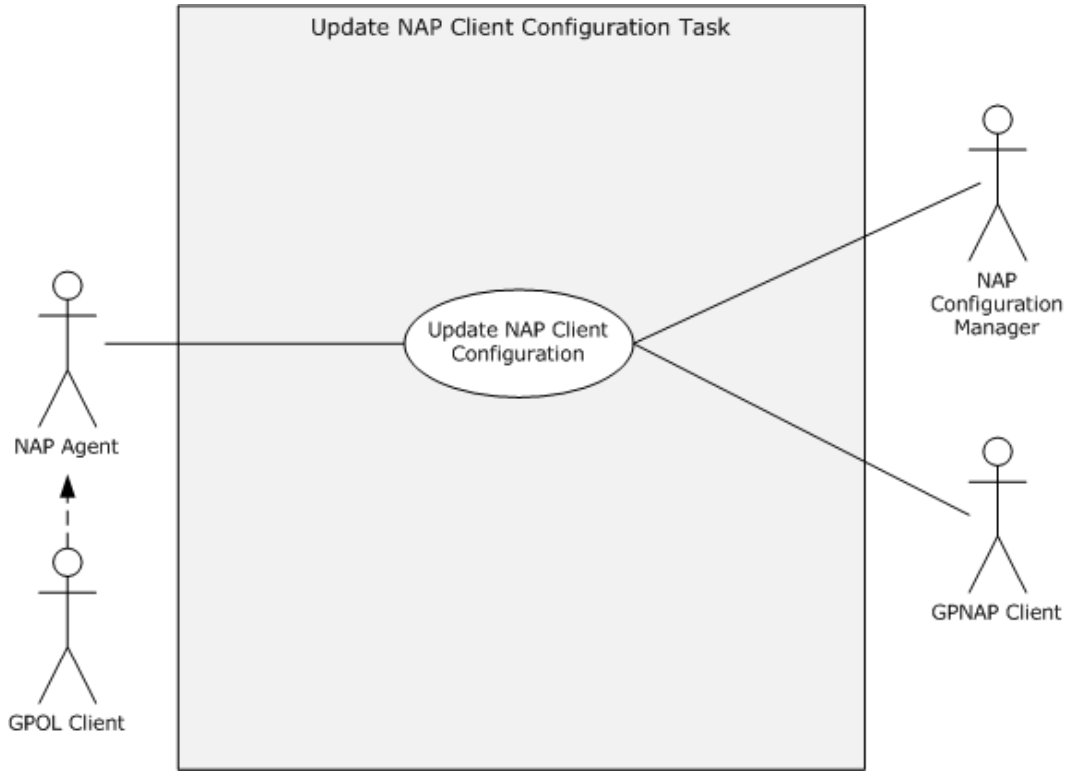

Figure 9: Update NAP Client Configuration use case diagram

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

Goal: To update the NAP Configuration on the NAP Client via Group Policy.

Context of Use: This use case is initiated when the NAP Client restarts, when the internal timer in the NAP agent triggers the task, or when there is a configuration update event. The direct actor is internal to the task.

Direct Actor: This role is performed by the NAP agent.

Primary Actor: This role is performed by the GPOL Client.

Supporting Actors:

NAP Configuration Manager: Manages the NAP Configuration store on the NAP Client. The use case employs the NAP Configuration Manager to update values stored in the NAP Configuration.

GPNAP client: A Group Policy client that fetches NAP specific policy from the Group Policy System using [MS-GPREG], as described in [MS-GPNAP]. The use case employs the GPNAP client to fetch NAP-specific policy settings from the Group Policy server.

Stakeholders and Interests:

- **Create and Send SoH Task**: The purpose of the Create and Send SoH Task (section 5) is to create SoH messages and send them, along with additional NAP data, to the NAP transport

protocols, for eventual delivery to the NAP Health Policy Server. To accomplish this task, it uses the settings found in the NAP configuration. As such, its interests in this task are that valid NAP Configuration values are available at all times.

Precondition: The NAP Client components are deployed and configured correctly. The integrity of the NAP Configuration storage is intact. The NAP Configuration contains valid values for all fields.

Minimal Guarantees:

1. The GPNAP client will always be invoked to attempt a fetch of the NAP policy.
2. The use case will always process the triggering event.
3. The NAP Configuration will always contain valid values.

Success Guarantee: The GPNAP client retrieves the NAP Group Policy from the Group Policy System and the NAP Configuration is updated with their values.

Trigger: The Task can be triggered by any of the following:

- The NAP Client restarts while connected to the network.
- The NAP Client is connected to the network, after being started unconnected.
- The GPOL periodic refresh timer is triggered.

Main Success Scenario:

1. The GPOL Client triggers the use case when one of the following events occurs:
   - The NAP Client restarts while connected to the network.
   - The NAP Client is connected to the network, after being started unconnected.
   - The GPOL periodic refresh timer is triggered.
2. The NAP Agent notifies the GPNAP Client that Group Policy retrieval is required.
3. The GPNAP client retrieves the NAP Group Policy from the Group Policy System.
4. The NAP Agent receives the new NAP Group Policy from the GPNAP Client.
5. The NAP Agent iterates through the new NAP Group Policy, doing the following:
   - The NAP Agent extracts a GPNAP value from the new NAP Group Policy.
   - The NAP Agent verifies the extracted GPNAP value is valid.
   - The NAP Agent sends the extracted GPNAP value to the NAP Configuration Manager.
   - The NAP Configuration Manager updates the corresponding GPNAP value in the NAP Configuration with the extracted GPNAP value, maintaining storage integrity at all times.
6. If the NAP Configuration Manager notes that any field value has changed from its previous value, the NAP Configuration Manager send an NAP configuration changed system event.
7. After the entire new NAP Group Policy is processed, the Nap Configuration is up to date.

Extensions: None.

4.2 Task Context

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

4.2.1 Task Environment

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

Requirement: The NAP Client has network and domain 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.
- **Means of satisfying the requirement:**

  1. The network interface of the NAP Client 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 NAP Client are configured to enable correct packet routing between the NAP Client and the Group Policy server.
  5. The NAP Client has joined the domain containing the Group Policy Server.

- **Means of knowing requirement satisfied**:

  1. The NAP Client can successfully ping the Group Policy server over the network.
  2. A user on the NAP Client can successfully login to the domain containing the Group Policy Server.

- **Consequences of not satisfying requirement**: The task is unable to receive Group Policy: NAP Extension (MS-GPNAP) values.

Requirement: The NAP Configuration store is uncorrupted and accessible.

- **Reason for requirement**: The NAP Configuration Manager will need to update the NAP Configuration store with Group Policy values.
- **Means of satisfying the requirement**:

  1. The NAP Configuration Manager is configured with the path and security settings to access the NAP Configuration store.
  2. The NAP Configuration Manager Service is started.
  3. The NAP Configuration Manager verifies the integrity of the NAP Configuration store, fixing any corrupted data as it is found.

- **Means of knowing requirement satisfied**:

  1. Every field in the NAP Configuration store can be accessed and the field value retrieved.

2. Each field value in the NAP Configuration store is within the range specified in section 2 Structures of [MS-GPNAP].

- **Consequences of not satisfying requirement**: The task will not be able to update the values in the NAP Configuration store with the values it receives from the GPNAP client.

4.2.2 Task Relationships

4.2.2.1 Black-Box Relationship Diagrams

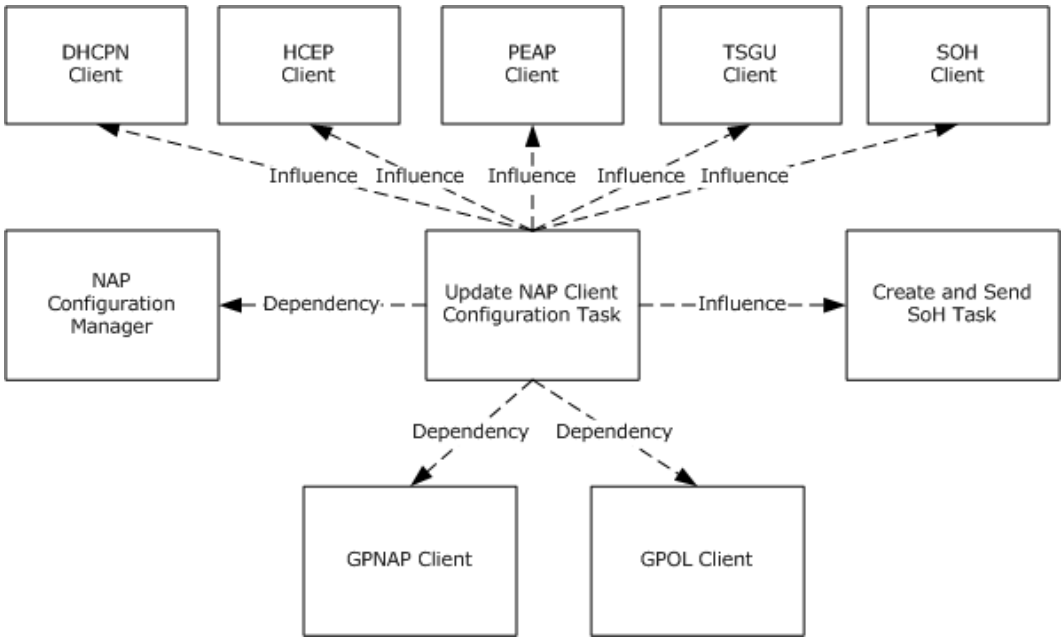

Figure 10: Update NAP Client Configuration Task black-box relationships

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

4.2.2.2 Task Dependencies

The Update NAP Client Configuration Task is dependent on the GPNAP Client for receiving current NAP configuration from the Group Policy Server.

The Update NAP Client Configuration Task is dependent on the GPOL Client for notifying the task when it is time to fetch new GPNAP values.

The Update NAP Client Configuration Task is dependent on the NAP Configuration Manager for updating the NAP configuration with values it receives from the GPNAP Client.

4.2.2.3 Task Influences

The Update NAP Client Configuration Task influences each of the transport protocol clients (DHCPN, HCEP, EAPE and TSGU) by updating the NAP Configuration, which determines whether the transport protocol clients carry SoH packets within their payload.

The Update NAP Client Configuration Task also influences the SoH client by updating the NAP Configuration, which determines the type of SoH packet that is created.

By updating the NAP Configuration, the Update NAP Client Configuration Task also influences the Create and Send Task, which uses the NAP Configuration values in its processing.

4.2.3 Task Assumptions and Preconditions

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

- The NAP Client components are deployed and configured correctly.
- The integrity of the NAP Configuration storage is intact.
- The NAP Configuration contains valid values for all fields.

4.2.4 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.

4.3 Task Architecture

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

4.3.1 Task Architectural Constraints

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

4.3.2 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.

NAP Configuration: The following fields constitute the NAP Configuration persistent data store. See [MS-GPNAP] section 2 Structures for additional information.

Name	Type	Description
Trace Settings		
EnableTracing*	DWORD	Indicates whether or not NAP tracing is enabled in a NAP Client: 0 = Disables NAP tracing on the client.

Name	Type	Description
		1 = Enables NAP tracing on the client.
TracingLevel*	DWORD	Indicates the level of NAP tracing in a NAP Client: 0 = Disables NAP tracing on the client. 1 = Sets NAP tracing on the client to basic level. 2 = Sets NAP tracing on the client to advanced level. 3 = Sets NAP tracing on the client to debug level.
User Interface Settings		
SmallText**	Unicode String	The user notification title displayed to the user.
LargeText**	Unicode String	The user notification sub-title displayed to the user.
ImageFile**	Octet Image Stream	Image that is displayed in the NAP Client user interface.
ImageFileName**	Unicode String	Image filename. Used to determine the format of the ImageData field (jpeg, tiff, etc).
Enforcement Client Settings		
DHCP Enforcement	DWORD	Indicates whether health policies should be enforced when a NAP Client attempts to obtain an IP address from a DHCP server: 0 = NAP enforcement disabled. 1 = NAP enforcement enabled.
Remote Access Enforcement	DWORD	Indicates whether health policies should be enforced when a NAP Client attempts to gain access to the network through a virtual private network (VPN) connection: 0 = NAP enforcement disabled. 1 = NAP enforcement enabled.
IPsec Enforcement	DWORD	Indicates whether health policies should be enforced when a NAP Client attempts to communicate with another computer using IPsec: 0 = NAP enforcement disabled. 1 = NAP enforcement enabled.
PlumbIpsecPolicy	DWORD	The IPsec NAP enforcement client can be configured to use the local predefined IPsec policy rather than the default IPsec policy: 0 = Allows the use of domain-based IPsec Group Policy on the client. 1 = Uses the local IPsec policy on the client.
Wireless EAPOL Enforcement	DWORD	Indicates whether health policies should be enforced when a NAP Client attempts to access a network through an 802.1X wireless connection:

Name	Type	Description
		0 = NAP enforcement disabled. 1 = NAP enforcement enabled.
RDG Enforcement	DWORD	Indicates whether health policies should be enforced when a NAP Client attempts to gain access to an RDG: 0 = NAP enforcement disabled. 1 = NAP enforcement enabled.
EAP Enforcement	DWORD	Indicates that health policies should be enforced when a NAP Client attempts to access a network through an 802.1X connection or an authenticating switch connection: 0 = NAP enforcement disabled. 1 = NAP enforcement enabled.
Health Registration Authority (HRA) Settings		
Cryptographic Service Provider (CSP)	Unicode String	The name of the cryptographic service provider (CSP) that is used to generate the key pair on the HCEP client. See [MS-GPNAP] section 2.4.1.1 for a description of the valid values.
Cryptographic Provider Type	DWORD	The type of the cryptographic service provider (CSP) that is used to generate the key pair on the HCEP client. See [MS-GPNAP] section 2.4.1.2 for a description of the valid values.
Public Key OID	Unicode String	An object identifier (OID) that identifies the algorithm of the public-private key pair associated with the certificate. See [MS-GPNAP] section 2.4.1.3 for a description of the valid values.
Public Key Length	DWORD	A 32-bit value consisting of the public key length. The minimum "Public Key Length" expected is 0x00000800.
Public Key Spec	DWORD	Indicates the type of public key (use with encryption, digital signatures, or both). By default, set to 0x00000001 (AT_KEYEXCHANGE).
Hash Algorithm OID	Unicode String	An OID identifying the hash algorithm used to sign the certificate request. See [MS-GPNAP] section 2.4.1.6 for a description of the valid values.
HRA Auto-Discovery	DWORD	HRA groups can be set by group policy or can be discovered automatically by the NAP client using DNS SRV lookup, as specified in [RFC2782]. Section 2.4.2 HRA Auto-Discovery in [MS-GPNAP] describes the process. 0 = Disables HRA auto discovery. 1 = Enables HRA auto discovery.
Use SSL	DWORD	An indicator that specifies whether SSL is enabled for communication with the HRA: 0 = Disables SSL. 1 = Enables SSL.
HRA URL List	List	An ordered list of HRA URLs which the HCEP client will attempt

Name	Type	Description
		to connect to. Each entry in the list has the following two values:
Server	Unicode String	Specifies the HRA URL to connect to.
Order	DWORD	The order of the HRA URL within the list.
Reconnect Attempts	DWORD	A 32-bit value representing time in minutes. The time (in minutes) that the client should wait before attempting to reconnect to an HRA in the event of a connection failure.
SoH Settings		
ShatimeoutInMsec	DWORD	A 32-bit value representing time in milliseconds. SHA timeout value.
BackwardCompatible	DWORD	Determines the version of the message header, as specified in figures 2 and 3 in [MS-SOH] section 2.2: 1 = SoH header as shown in Figure 2 (SSoH only) 2 = SoH header as shown in Figure 3 (SoH Mode Subheader plus SSoH)

Notes:

* Denotes elements only used for tracing of NAPSO functionality.

** Denotes elements only used by the editor GUI used to modify [MS-GPNAP] values.

4.3.3 Task Abstract Parameters

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.

This task contains no abstract parameters.

4.3.4 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 never called by an external entity or returns results.

4.3.5 White-Box Relationships

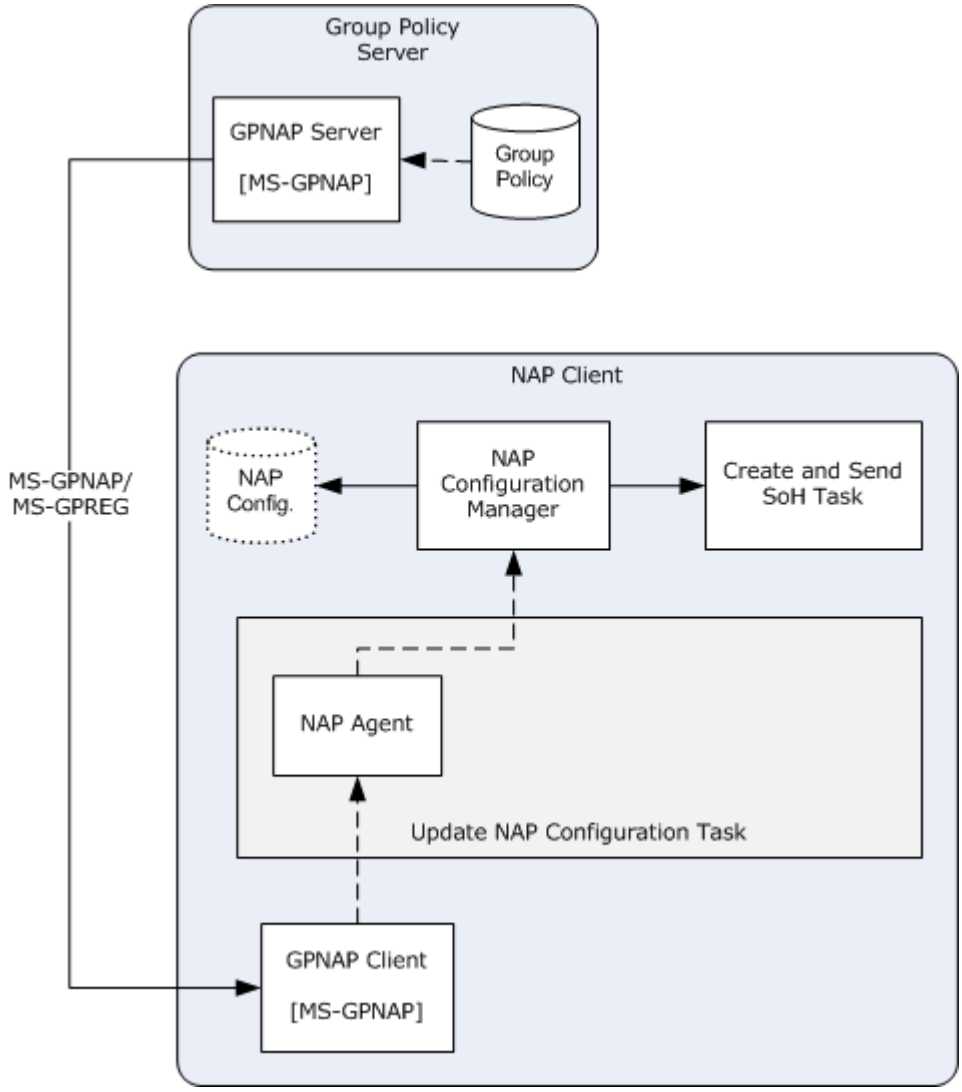

Figure 11: Update NAP Client Configuration Task white-box relationships

The NAP agent contacts the GPNAP client to fetch the latest group policy. The GPNAP client fetches the group policy from the Group Policy Server via [MS-GPNAP] over [MS-GPREG]. The [MS-GPNAP] settings are passed to the NAP Agent, who subsequently passes each setting to the NAP Configuration Manager for storage in the NAP Configuration store.

4.3.6 Task Events

4.3.6.1 Task Timers

The system does not define any task timers beyond those implemented in the member protocols.

4.3.6.2 Task Non-Timer Events

This task can receive three non-timer events from the GPOL client:

- The NAP Client restarts while connected to the network.
- The NAP Client is connected to the network, after being started unconnected.
- The GPOL periodic refresh timer is triggered.

4.3.7 Task Architecture and Communication

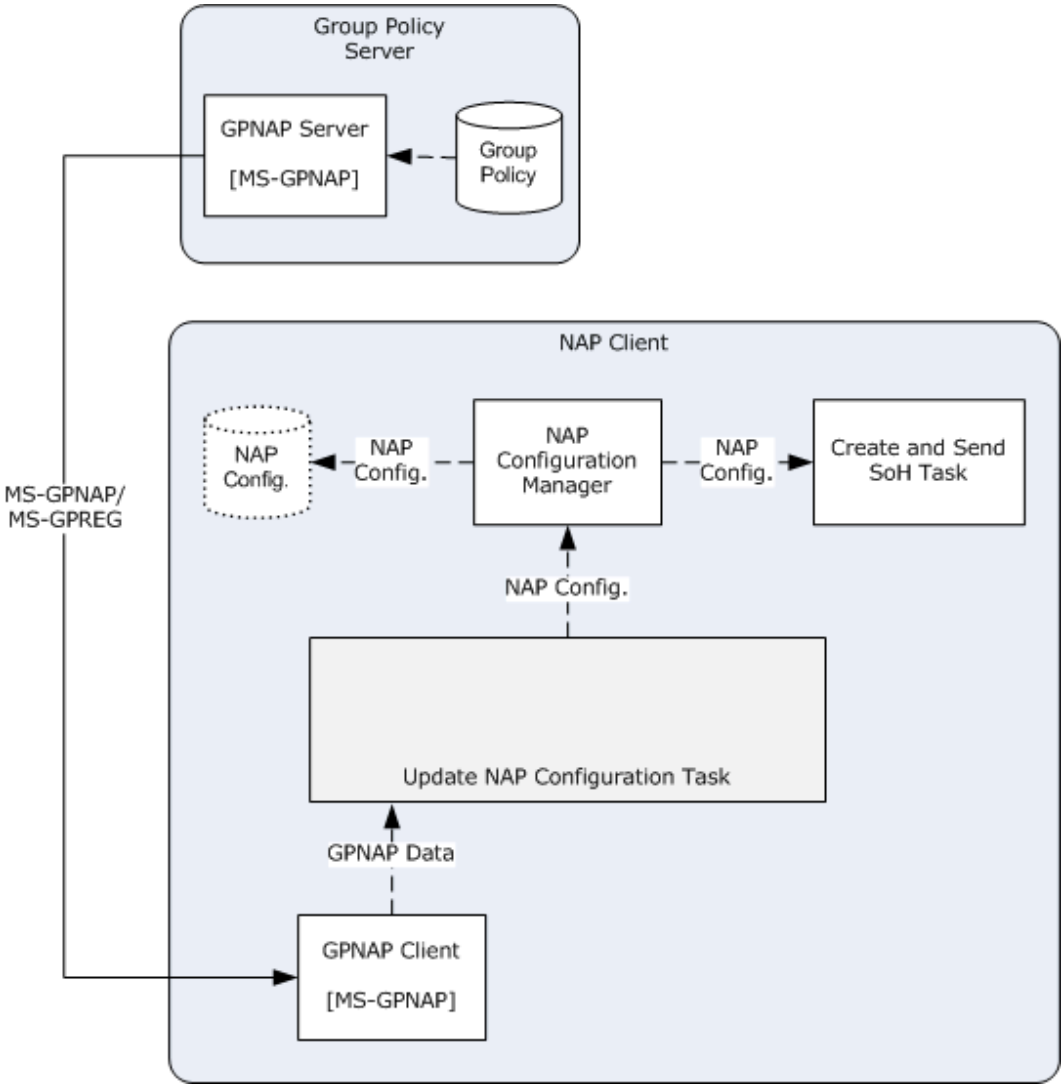

Figure 12: Update NAP Client Configuration Task architecture and communication overview

4.3.8 Task Processing Rules

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

1. The GPOL Client triggers this task when one of the following events occurs:
   - The NAP Client restarts while connected to the network, as described in [MS-GPOL].
   - The NAP Client is connected to the network, after being started unconnected, as described in [MS-GPOL].
   - The GPOL periodic refresh timer is triggered, as described in [MS-GPOL].
2. The NAP Agent triggers the GPNAP Client to retrieve the Group Policy, as described in [MS-GPNAP] and [MS-GPREG].
3. The NAP Agent receives the new NAP Group Policy from the GPNAP Client.
4. The NAP Agent iterates through the NAP Group Policy object, extracting the GPNAP value and processing each GPNAP value as follows:
   - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP].
   - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
   - The NAP Configuration Manager updates corresponding value in the NAP Configuration with the extracted GPNAP value.
5. If the NAP Configuration Manager notes that any field value has changed from its previous value, the NAP Configuration Manager invokes a NAP configuration changed system event.

4.3.9 Task Failure Scenarios

4.3.9.1 Task Fails to Receive System Configuration

If the NAP agent fails to receive the configuration from the GPNAP Client, such as when the GPNAP Client cannot access the Group Policy Server, the ADM elements passed to the Create and Send SoH Task may contain the wrong configuration for sending the SoH.

4.4 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.

4.4.1 Task Precondition Details

For the complete list of task preconditions and assumptions, see section 4.2.3. Details for some of the preconditions are as follows:

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

4.4.2 Task Initialization of External Entities

None.

4.4.3 Task Event Details

4.4.3.1 Task Timer Details

The system does not define any task timers beyond those implemented in the member protocols.

4.4.3.2 Task Non-Timer Event Details

This task can receive three non-timer events from the GPOL client:

- The NAP Client restarts while connected to the network.
- The NAP Client is connected to the network, after being started unconnected.
- The GPOL periodic refresh timer is triggered.

In each case, the GPOL client triggers this task to fetch new [MS-GPNAP] policy via the GPNAP client.

4.4.4 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.

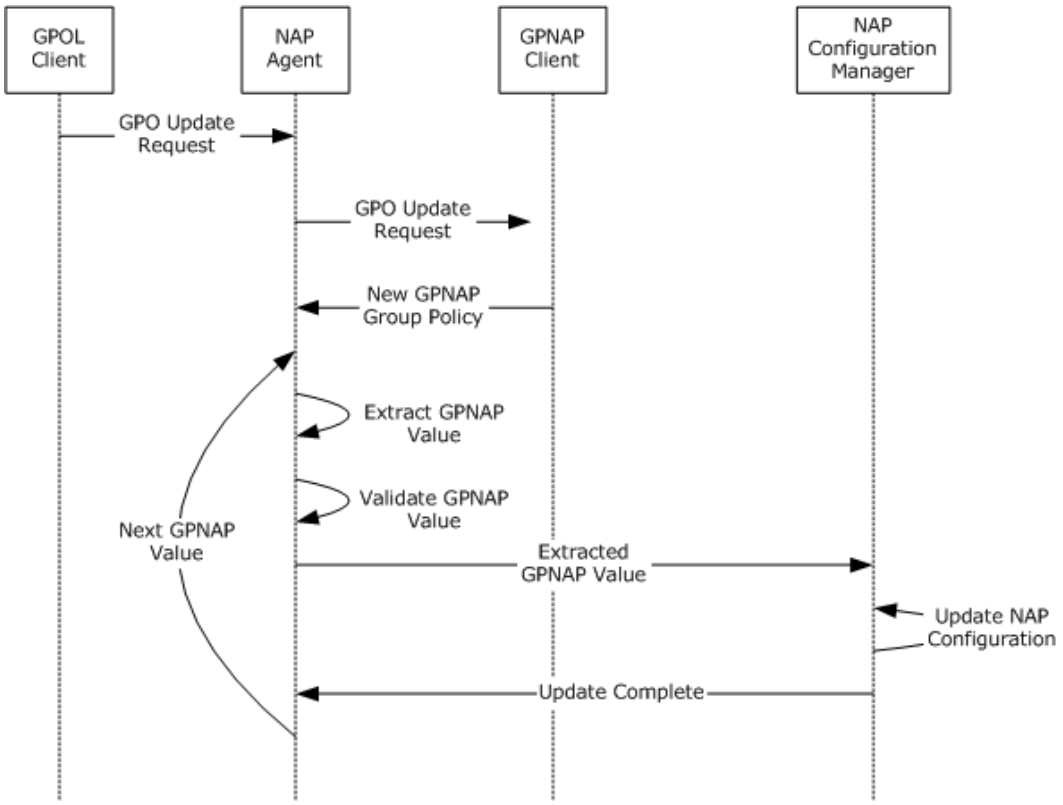

Figure 13: Sequence diagram for the main success scenario of the Update NAP Client Configuration Task

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.

4.4.5 Task Processing Rule Details

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

1. The GPOL Client triggers this task when one of the following events occurs:
   - The NAP Client restarts while connected to the network, as described in [MS-GPOL] section 3.2 Client Details.
   - The NAP Client is connected to the network, after being started unconnected, as described in [MS-GPOL] section 3.2 Client Details.

- The GPOL periodic refresh timer is triggered, as described in [MS-GPOL] section 3.2.2 Timers.

2. The NAP Agent triggers the GPNAP Client to retrieve the Group Policy, as described in [MS-GPNAP] section 1.3.2 Group Policy Extension Overview and [MS-GPREG] section 3.2 Client Plug-in Details.

3. The NAP Agent receives the new NAP Group Policy (see section 2 Structures in [MS-GPNAP]) from the GPNAP Client.

4. The NAP Agent iterates through the NAP Group Policy object, extracting the GPNAP value and processing the values as follows:

    1. For the GPNAP attribute Enable Tracing:

        - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.1.1 Enable Tracing.
        - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
        - The NAP Configuration Manager updates EnableTracing in the NAP Configuration with the extracted GPNAP value.

    2. For the GPNAP attribute Tracing Level:

        - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.1.2 Tracing Level.
        - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
        - The NAP Configuration Manager updates TracingLevel in the NAP Configuration with the extracted GPNAP value.

    3. For the GPNAP attribute SmallText:

        - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.2.1 SmallText.
        - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
        - The NAP Configuration Manager updates SmallText in the NAP Configuration with the extracted GPNAP value.

    4. For the GPNAP attribute LargeText:

        - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.2.2 LargeText.
        - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
        - The NAP Configuration Manager updates LargeText in the NAP Configuration with the extracted GPNAP value.

    5. For the GPNAP attribute ImageFile:

        - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.2.3 ImageFile.
        - The NAP agent passes the GPNAP value to the NAP Configuration Manager.

- The NAP Configuration Manager updates ImageFile in the NAP Configuration with the extracted GPNAP value.

6. For the GPNAP attribute ImageFileName:
   - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.2.4 ImageFileName.
   - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
   - The NAP Configuration Manager updates ImageFileName in the NAP Configuration with the extracted GPNAP value.
7. For the GPNAP attribute DHCP Enforcement:
   - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.3.1 DHCP Enforcement.
   - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
   - The NAP Configuration Manager updates DHCP Enforcement in the NAP Configuration with the extracted GPNAP value.
8. For the GPNAP attribute Remote Access Enforcement:
   - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.3.2 Remote Access Enforcement.
   - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
   - The NAP Configuration Manager updates Remote Access Enforcement in the NAP Configuration with the extracted GPNAP value.
9. For the GPNAP attribute IPsec Enforcement:
   - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.3.3 IPsec Enforcement.
   - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
   - The NAP Configuration Manager updates IPsec Enforcement in the NAP Configuration with the extracted GPNAP value.
10. For the GPNAP attribute PlumbIpsecPolicy:
    - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.3.3 IPsec Enforcement.
    - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
    - The NAP Configuration Manager updates PlumbIpsecPolicy in the NAP Configuration with the extracted GPNAP value.
11. For the GPNAP attribute Wireless EAPOL Enforcement:
    - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.3.4 Wireless EAPOL Enforcement.
    - The NAP agent passes the GPNAP value to the NAP Configuration Manager.

- The NAP Configuration Manager updates Wireless EAPOL Enforcement in the NAP Configuration with the extracted GPNAP value.

12. For the GPNAP attribute RDG Enforcement:

    - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.3.5 RDG Enforcement.
    - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
    - The NAP Configuration Manager updates RDG Enforcement in the NAP Configuration with the extracted GPNAP value.

13. For the GPNAP attribute EAP Enforcement:

    - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.3.6 EAP Enforcement.
    - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
    - The NAP Configuration Manager updates EAP Enforcement in the NAP Configuration with the extracted GPNAP value.

14. For the GPNAP attribute Cryptographic Service Provider (CSP):

    - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.4.1.1 Cryptographic Service Provider (CSP).
    - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
    - The NAP Configuration Manager updates Cryptographic Service Provider (CSP) in the NAP Configuration with the extracted GPNAP value.

15. For the GPNAP attribute Cryptographic Provider Type:

    - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.4.1.2 Cryptographic Provider.
    - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
    - The NAP Configuration Manager updates Cryptographic Provider in the NAP Configuration with the extracted GPNAP value.

16. For the GPNAP attribute Public Key OID:

    - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.4.1.3 Public Key OID.
    - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
    - The NAP Configuration Manager updates Public Key OID in the NAP Configuration with the extracted GPNAP value.

17. For the GPNAP attribute Public Key Length:

    - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.4.1.4 Public Key Length.
    - The NAP agent passes the GPNAP value to the NAP Configuration Manager.

- The NAP Configuration Manager updates Public Key Length in the NAP Configuration with the extracted GPNAP value.

18. For the GPNAP attribute Public Key Spec:

    - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.4.1.5 Public Key Spec.
    - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
    - The NAP Configuration Manager updates Public Key Spec in the NAP Configuration with the extracted GPNAP value.

19. For the GPNAP attribute Hash Algorithm OID:

    - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.4.1.6 Hash Algorithm OID.
    - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
    - The NAP Configuration Manager updates Hash Algorithm OID in the NAP Configuration with the extracted GPNAP value.

20. For the GPNAP attribute HRA Auto-Discovery:

    - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.4.2 HRA Auto-Discovery.
    - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
    - The NAP Configuration Manager updates HRA Auto-Discovery in the NAP Configuration with the extracted GPNAP value.

21. For the GPNAP attribute Use SSL:

    - The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.4.3 Use SSL.
    - The NAP agent passes the GPNAP value to the NAP Configuration Manager.
    - The NAP Configuration Manager updates Use SSL in the NAP Configuration with the extracted GPNAP value.

22. For each the GPNAP attribute HRA URL List:

    - The NAP agent verifies the GPNAP Server value against the range in [MS-GPNAP] section 2.4.4.1 Server.
    - The NAP agent verifies the GPNAP Server value against the range in [MS-GPNAP] section 2.4.4.2 Order.
    - The NAP agent passes both GPNAP values to the NAP Configuration Manager.
    - The NAP Configuration Manager updates Server value for the HRA URL in the NAP Configuration with the extracted GPNAP value.
    - The NAP Configuration Manager updates Order value for the HRA UR in the NAP Configuration with the extracted GPNAP value.

23.For the GPNAP attribute Reconnect Attempts:

- The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.4.5 Reconnect Attempts.
- The NAP agent passes the GPNAP value to the NAP Configuration Manager.
- The NAP Configuration Manager updates Reconnect Attempts in the NAP Configuration with the extracted GPNAP value.

24.For the GPNAP attribute Task Timer:

- The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.5.1 Task Timer.
- The NAP agent passes the GPNAP value to the NAP Configuration Manager.
- The NAP Configuration Manager updates ShatimeoutInMsec in the NAP Configuration with the extracted GPNAP value.

25.For the GPNAP attribute Backward Compatible:

- The NAP agent verifies the GPNAP value against the range in [MS-GPNAP] section 2.5.2 Backward Compatible.
- The NAP agent passes the GPNAP value to the NAP Configuration Manager.
- The NAP Configuration Manager updates BackwardCompatible in the NAP Configuration with the extracted GPNAP value.

5. If the NAP Configuration Manager notes that any field value has changed from its previous value, the NAP Configuration Manager invokes a NAP configuration changed system event.

4.5 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.

5 Create and Send SoH Task

This section describes how the NAP agent creates an SoH on a NAP Client. This task is initiated when there is a trigger for new health information on a NAP Client 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 5.1.3.4. The format of the SoH message is specified in [MS-SOH]. 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: [MS-SOH], [MS-WSH], [MS-HCEP], [MS-DHCPN], [MS-TSGU], [IEEE802.1X], and [MS-PEAP].

5.1 Task Overview

5.1.1 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 NAP Client when a health state change occurs. This task includes but is not limited to health status assessment and SoH composition.

5.1.2 Task Applicability

This task is used when a NAP Client attempts to access network-based resources and the NAP System is deployed on the NAP Client. 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 Create and Send SoH Task are as follows:

NAP Agent: The main software component on the NAP Client. 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 ability to perform its services is where the NAP agent's interests in this task are.

NAP Event Handler: A component on the NAP Client that receives events from the underlying layers and passes them to the NAP Agent. The interest of this actor in the task is that the event passed to the task is processed.

HCEP Client: This protocol client is used to send [MS-HCEP] messages to an HCEP server on the NEP computer. This is typically done when the NAP Client 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 interest of this actor in the task is that the QecConnection passed to the task is used when returning the NAP request data.

DHCP Client: This protocol client is used to send [MS-DHCPN] messages to a DHCPN server on the NEP computer. This is typically done when the NAP Client 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 interest of this actor in the task is that the QecConnection passed to the task is used when returning the NAP request data.

TSGU Client: This protocol client is used to send [MS-TSGU] messages to a TSGU server on the NEP computer. This is typically done when a user on the NAP Client attempts to RDP into a computer located on the corporate network, which requires going through a terminal services gateway. The interest of this actor in the task is that the QecConnection passed to the task is used when returning the NAP request data.

CEAP Peer: This protocol client is used to send [MS-CEAP] messages to a CEAP server on the NEP computer. There are two typical scenarios where a [MS-CEAP] protocol message exchange is triggered. The first scenario occurs when the NAP Client first boots up and authenticates against an 802.1X device, such as a gateway router. The second scenario occurs when a user on the NAP Client attempts to VPN into a resource located on the corporate network. The interest of this actor in the task is that the QecConnection passed to the task is used when returning the NAP request data.

Proxy SoH Task: The purpose of this stakeholder is to proxy the NAP request 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 NAP request data.

Network Administrator: The Network Administrator wants to limit the computers connected to the corporate network to those verified as healthy. To implement this restriction, the Network Administrator's interests in this specific task are to ensure that only valid health information about the client computer are sent within the SoH Packets, so that the correct health state of the computer can be determined.

5.1.3.2 Supporting Actors and Task Interests Summary

SOH Client: The purpose of this actor is to utilize the processing rules defined in [MS-SOH] 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 task employs this actor whenever a new SoH packet is required.

HCEP Client: This protocol client is used to send [MS-HCEP] messages to an HCEP server on the NEP computer. This is typically done when the NAP Client 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 task employs this actor whenever a SoH packet must be sent to the HCEP server.

DHCP Client: This protocol client is used to send [MS-DHCPN] messages to a DHCPN server on the NEP computer. This is typically done when the NAP Client 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 task employs this actor whenever a SoH packet must be sent to the DHCPN server.

TSGU Client: This protocol client is used to send [MS-TSGU] messages to a TSGU server on the NEP computer. This is typically done when a user on the NAP Client attempts to RDP into a computer located on the corporate network, which requires going through a terminal services gateway. The task employs this actor whenever a SoH packet must be sent to the TSGU server.

CEAP Peer: This protocol client is used to send [MS-CEAP] messages to a CEAP server on the NEP computer. There are two typical scenarios where a [MS-CEAP] protocol message exchange is triggered. The first scenario occurs when the NAP Client first boots up and authenticates against an

802.1X device, such as a gateway router. The second scenario occurs when a user on the NAP Client attempts to VPN into a resource located on the corporate network. The task employs this actor whenever a SoH packet must be sent to the CEAP server.

NAP Configuration Manager: Manages the NAP Configuration store on the NAP Client. The NAP configuration (see section 5.3.2 Task Abstract Data Model) contains elements representing all the settings found in [MS-GPNAP]. The task contacts the NAP Configuration Manager whenever it needs NAP configuration values.

Authentication Client: The Credential Cache is managed by this actor. The authentication data stored in this cache is obtained by an external actor, such as a Microsoft Windows® prompt, the Windows Registry, etc. The task employs this actor whenever authentication data is needed by the protocol clients for encapsulation within the protocol message.

5.1.3.3 Use Case Diagrams

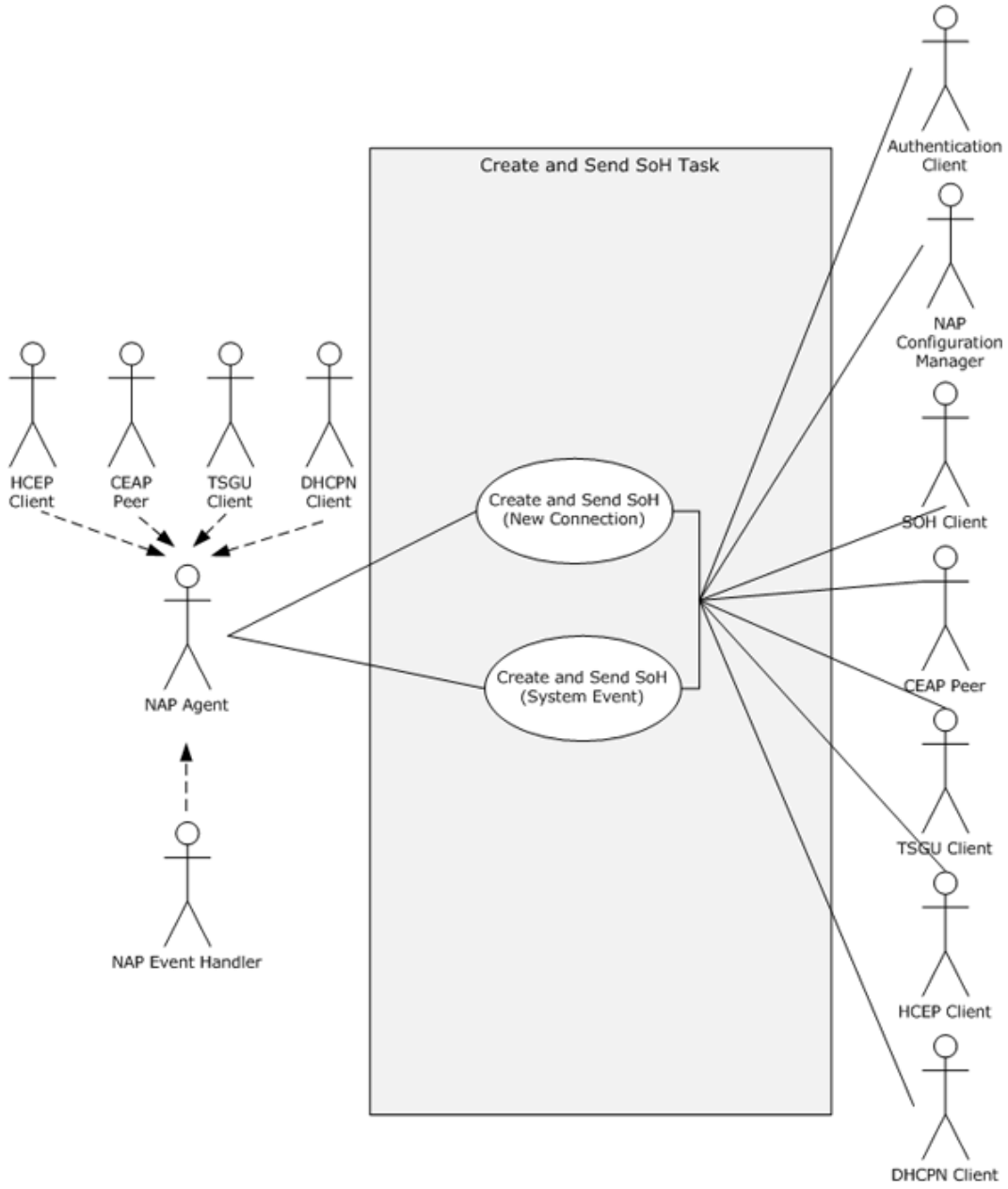

Figure 14: Create and Send SoH Task use case diagram

5.1.3.4 Use Case: Create and Send SoH (New Connection) - NAP Agent

This use case is associated with the use case diagram in section 5.1.3.3.

Goal: To create a SoH message [MS-SOH] containing health information about the NAP Client and send the SoH message to the NAP transport protocol ([MS-DHCPN], [MS-HCEP], [MS-CEAP] and [MS-TSGU]) client.

Context of Use: This use case is used when an external event triggers a new NAP transport protocol connection that requires NAP request data.

Direct Actor: This role is performed by the NAP Agent.

Direct Actor: This role is performed by the **NAP Agent**.

Primary Actor: This role is performed by four actors: the DHCPN Client, the HCEP Client, the TSGU Client and the CEAP Peer.

Supporting Actors:

SOH Client: The purpose of this actor is to utilize the processing rules defined in [MS-SOH] 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 Client: This protocol client is used to send [MS-HCEP] messages to an HCEP server on the NEP computer. This is typically done when the NAP Client 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 a SoH packet must be sent to the HCEP server.

DHCP Client: This protocol client is used to send [MS-DHCPN] messages to a DHCPN server on the NEP computer. This is typically done when the NAP Client 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 a SoH packet must be sent to the DHCPN server.

TSGU Client: This protocol client is used to send [MS-TSGU] messages to a TSGU server on the NEP computer. This is typically done when a user on the NAP Client attempts to RDP into a computer located on the corporate network, which requires going through a terminal services gateway. The use case employs this actor whenever a SoH packet must be sent to the TSGU server.

CEAP Peer: This protocol client is used to send [MS-CEAP] messages to a CEAP server on the NEP computer. There are two typical scenarios where a [MS-CEAP] protocol message exchange is triggered. The first scenario occurs when the NAP Client first boots up and authenticates against an 802.1X device, such as a gateway router. The second scenario occurs when a user on the NAP Client attempts to VPN into a resource located on the corporate network. The use case employs this actor whenever a SoH packet must be sent to the CEAP server.

NAP Configuration Manager: Manages the NAP Configuration store on the NAP Client. The NAP configuration (see section 5.3.2 Task Abstract Data Model) contains elements representing all the settings found in [MS-GPNAP]. The use case contacts the NAP Configuration Manager whenever it needs NAP configuration values.

Authentication Client: The Credential Cache is managed by this actor. The authentication data stored in this cache is obtained by an external actor, such as a Microsoft Windows® prompt, the Windows Registry, etc. The use case employs this actor whenever authentication data is needed by the protocol clients for encapsulation within the protocol message.

Stakeholders and Interests:

HCEP Client: This protocol client is used to send [MS-HCEP] messages to an HCEP server on the NEP computer. This is typically done when the NAP Client 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 interest of this actor in the use case is that the QecConnection passed to the task is used when returning the NAP request data.

DHCP Client: This protocol client is used to send [MS-DHCPN] messages to a DHCPN server on the NEP computer. This is typically done when the NAP Client 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 interest of this actor in the use case is that the QecConnection passed to the task is used when returning the NAP request data.

TSGU Client: This protocol client is used to send [MS-TSGU] messages to a TSGU server on the NEP computer. This is typically done when a user on the NAP Client attempts to RDP into a computer located on the corporate network, which requires going through a terminal services gateway. The interest of this actor in the use case is that the QecConnection passed to the task is used when returning the NAP request data.

CEAP Peer: This protocol client is used to send [MS-CEAP] messages to a CEAP server on the NEP computer. There are two typical scenarios where a [MS-CEAP] protocol message exchange is triggered. The first scenario occurs when the NAP Client first boots up and authenticates against an 802.1X device, such as a gateway router. The second scenario occurs when a user on the NAP Client attempts to VPN into a resource located on the corporate network. The interest of this actor in the use case is that the QecConnection passed to the task is used when returning the NAP request data.

Proxy SoH Task: The purpose of this stakeholder is to proxy the NAP request 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 use case only sends protocol messages that contain NAP request data.

Network Administrator: The Network Administrator wants to limit the computers connected to the corporate network to those verified as healthy. To implement this restriction, the Network Administrator's interests in this use case are to ensure that only valid health information about the client computer are sent within the SoH Packets, so that the correct health state of the computer can be determined.

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

Minimal Guarantees:

- NAP related operations are performed, without guaranteed success.
- The use case will always process the event it receives.
- The QecConnection parameter is always used when returning the NAP request data.
- No transport protocol messages are sent by the task without NAP request data.
- The task always sends only valid health information in the SoH Packets.

Success Guarantee: A SoH message is created with a valid health assessment and sent to the sent to the NAP transport protocol ([MS-DHCPN], [MS-HCEP], [MS-CEAP] and [MS-TSGU]) client.

Trigger: The Task can be triggered by any of the following:

- A DHCPN Client thread invokes the abstract interface of this task.

- A HCEP Client thread invokes the abstract interface of this task.
- A TSGU Server thread invokes the abstract interface of this task.
- A CEAP Peer thread invokes the abstract interface of this task.

Main Success Scenario:

1. The task is triggered by a thread associated with one of the following direct actors invoking the task's abstract interface:
   - The DHCPN Server.
   - The HCEP Server.
   - The TSGU Server.
   - The CEAP Peer.
2. The NAP Agent creates a new entry in the QecConnectList using the QecConnection and ProtocolType parameters.
3. The **NAP Agent** creates a unique correlation ID.
4. The **NAP Agent** requests and receives the current NAP configuration from the NAP Configuration Manager.
5. The **NAP Agent** obtains a SoH packet by calling calls the GetSoHRequest abstract interface in [MS-SOH].
6. The NAP agent requests and receives authentication data from the Authentication Client.
7. The NAP Agent invokes an abstract interface on the NAP transport protocol ([MS-DHCPN], [MS-HCEP], [MS-CEAP] and [MS-TSGU]) client corresponding to the QecConnection with the correlation ID, the SoH packet, and authentication data.

Extensions: None.

5.1.3.5 Use Case: Create and Send SoH (System Event) – NAP Agent

This use case is associated with the use case diagram in section 5.1.3.3.

Goal: To create an SoH message [MS-SOH] containing health information about the NAP Client and send the SoH message to the NAP transport protocol ([MS-DHCPN], [MS-HCEP], [MS-CEAP] and [MS-TSGU]) client.

Context of Use: This use case is used when an external event triggers a new NAP transport protocol connection that requires NAP request data.

Direct Actor: This role is performed by the NAP Agent.

Primary Actor: This role is performed by the **NAP Event Handler**.

Supporting Actors:

SOH Client: The purpose of this actor is to utilize the processing rules defined in [MS-SOH] 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 Client: This protocol client is used to send [MS-HCEP] messages to an HCEP server on the NEP computer. This is typically done when the NAP Client 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 a SoH packet must be sent to the HCEP server.

DHCP Client: This protocol client is used to send [MS-DHCPN] messages to a DHCPN server on the NEP computer. This is typically done when the NAP Client 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 a SoH packet must be sent to the DHCPN server.

TSGU Client: This protocol client is used to send [MS-TSGU] messages to a TSGU server on the NEP computer. This is typically done when a user on the NAP Client attempts to RDP into a computer located on the corporate network, which requires going through a terminal services gateway. The use case employs this actor whenever a SoH packet must be sent to the TSGU server.

CEAP Peer: This protocol client is used to send [MS-CEAP] messages to a CEAP server on the NEP computer. There are two typical scenarios where a [MS-CEAP] protocol message exchange is triggered. The first scenario occurs when the NAP Client first boots up and authenticates against an 802.1X device, such as a gateway router. The second scenario occurs when a user on the NAP Client attempts to VPN into a resource located on the corporate network. The use case employs this actor whenever a SoH packet must be sent to the CEAP server.

NAP Configuration Manager: Manages the NAP Configuration store on the NAP Client. The NAP configuration (see section 5.3.2 Task Abstract Data Model) contains elements representing all the settings found in [MS-GPNAP]. The use case contacts the NAP Configuration Manager whenever it needs NAP configuration values.

Authentication Client: The Credential Cache is managed by this actor. The authentication data stored in this cache is obtained by an external actor, such as a Microsoft Windows® prompt, the Windows Registry, etc. The use case employs this actor whenever authentication data is needed by the protocol clients for encapsulation within the protocol message.

Stakeholders and Interests:

Proxy SoH Task: The purpose of this stakeholder is to proxy the NAP request 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 use case only sends protocol messages that contain NAP request data.

Network Administrator: The Network Administrator wants to limit the computers connected to the corporate network to those verified as healthy. To implement this restriction, the Network Administrator's interests in this use case are to ensure that only valid health information about the client computer are sent within the SoH Packets, so that the correct health state of the computer can be determined.

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

Minimal Guarantees:

- NAP related operations are performed, without guaranteed success.

- The use case will always process the event it receives.
- No transport protocol messages are sent by the task without NAP request data.
- The task always sends only valid health information in the SoH Packets.

Success Guarantee: A SoH message is created with a valid health assessment and sent to the sent to the NAP transport protocol ([MS-DHCPN], [MS-HCEP] and [MS-TSGU]) client.

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 NAP Client has changed.
- The NAP Event Handler receives an event indicating that the NAP Client reboots or awakens from hibernation.
- The NAP Event Handler receives an event indicating that the SoH Client completes remediation.
- The NAP Event Handler receives an event indicating that another NAP authentication round is required.
- 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 NAP Client 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 NAP Client, 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 NAP Client has changed.
    - The NAP Event Handler receives an event indicating that the NAP Client has rebooted or awakens from hibernation.
    - The NAP Event Handler receives an event indicating that the SoH Client completes remediation.
    - The NAP Event Handler receives an event indicating that another NAP authentication round is required.
    - The NAP Event Handler receives an event indicating that a state change occurs on the NAP Client that would modify the current SoH values.
    - The NAP Event Handler receives an event indicating that the NAP configuration is modified.
2. The NAP Event Handler passes the event to the NAP Agent.
3. The NAP Agent creates a unique correlation ID.
4. The NAP Agent requests and receives the current NAP configuration from the NAP Configuration Manager.

5. The NAP Agent obtains a SoH packet by calling the GetSoHRequest abstract interface in [MS-SOH].

6. The NAP agent requests and receives authentication data from the Authentication Client.

7. The NAP Agent determines which transport protocol processing is required by reviewing the Enforcement Client Settings ADM and the trigger type.

8. If the NAP Agent determines the trigger is applicable to DHCPN, and DHCPN enforcement is enabled, the NAP Agent iterates through the QecConnectList table and calls the DhcpClientNotifySoHChange abstract interface in [MS-DHCPN] on all DHCPN connections in the table.

9. If the NAP Agent determines the trigger is applicable to HCEP, and HCEP enforcement is enabled, the NAP Agent iterates through the QecConnectList table and calls the HCEPEvaluateHealthRequest abstract interface in [MS-HCEP] on all HCEP connections in the table.

10. If the NAP Agent determines the trigger is applicable to TSGU, and TSGU enforcement is enabled, the NAP Agent iterates through the QecConnectList table and calls the NotifySoHChange abstract interface in [MS-TSGU] on all TSGU connections in the table.

11. If the NAP Agent determines the trigger is applicable to CEAP, and CEAP enforcement is enabled, the NAP Agent iterates through the QecConnectList table and calls the EvaluateHealthRequest abstract interface in [MS-CEAP] on all EAP connections in the table.

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 Agent in an environment where NAP Client request access to network resources under the control of devices or servers acting as Network Enforcement Points (NEP).

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

Requirement: The NAP Configuration store is uncorrupted and accessible.

- **Reason for requirement**: The NAP Configuration Manager will need to read the NAP Configuration store.
- **Means of satisfying the requirement**:

  1. The NAP Configuration Manager is configured with the path and security settings to access the NAP Configuration store.

  2. The NAP Configuration Manager Service is started.

  3. The NAP Configuration Manager verifies the integrity of the NAP Configuration store, fixing any corrupted data as it is found.

- **Means of knowing requirement satisfied**:

1. Every field in the NAP Configuration store can be accessed and the field value retrieved.

2. Each field value in the NAP Configuration store is within the range specified in section 2 Structures of [MS-GPNAP].

- **Consequences of not satisfying requirement**: The task will not be able to read the values from the NAP Configuration store.

Requirement: The HCEP Client is running and has the ability to forward [MS-HCEP] packets to the HCEP server on the Network Enforcement Point (NEP).

- **Reason for requirement:** The HCEP client is used to send NAP request data (including the SoH) encapsulated within [MS-HCEP] packets to the NEP.

- **Means of satisfying the requirement**:

1. The HCEP client is configured with the HCEP server settings described in section 2.4 Health Registration Authority (HRA) Settings in [MS-GPNAP].

2. The HCEP client has network access to the HCEP Server:

    1. The network interface of the NAP Client is configured to operate on the local subnet.

    2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NEP is connected.

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

    4. The routing tables in the NAP Client are configured to enable correct packet routing between the NAP Client and the NEP.

3. The HCEP client service has been started.

- **Means of knowing requirement satisfied**:

1. The NAP Client can successfully ping the NEP over the network.

2. The HCEP client is shown as running within the list of services.

3. A sniffer trace performed during a Health Certificate Enrollment event, shows HTTP packets traveling between the NAP Client and the NEP. These HTTP packets must contain [MS-HCEP] fields and reflect the HRA settings from [MS-GPNAP].

4. No errors are logged by the HCEP client.

- **Consequences of not satisfying requirement**: The task is unable to send NAP request data via the [MS-HCEP] protocol.

Requirement: The DHCPN Client is running and has the ability to forward [MS-DHCPN] packets to the DHCP server on the Network Enforcement Point (NEP).

- **Reason for requirement**: The DHCPN client is used to send NAP request data (including the SoH) encapsulated within [MS-DHCPN] packets to the NEP.

- **Means of satisfying the requirement**:

1. The DHCPN client is configured with the DHCP settings from the registry.

2. The DHCPN client has network access to the DHCP Server:
    1. The network interface of the NAP Client is configured to operate on the local subnet.
    2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NEP is connected.
    3. All network devices between the local subnet and the NEP are configured to allow packet flow between the two entities.
    4. The routing tables in the NAP Client are configured to enable correct packet routing between the NAP Client and the NEP.
3. The DHCPN client service has been started.

- **Means of knowing requirement satisfied**:
    1. The NAP Client can successfully ping the NEP over the network.
    2. The DHCPN client is shown as running within the list of services.
    3. An attempt to renew the IP address lease on each NIC completes successfully.
    4. A sniffer trace performed during a DHCP lease renewal event, shows DHCP packets traveling between the NAP Client and the NEP. These DHCP packets must contain [MS-DHCPN] fields.
    5. No errors are logged by the DHCPN client.
- **Consequences of not satisfying requirement**: The task is unable to send NAP request data via the [MS-DHCPN] protocol.

Requirement: The TSGU Client is running and has the ability to forward [MS-TSGU] packets to the TSGU server on the Network Enforcement Point (NEP).

- **Reason for requirement**: The TSGU client is used to send NAP request data (including the SoH) encapsulated within [MS-TSGU] packets to the NEP.
- **Means of satisfying the requirement**:
    1. The TSGU client is configured with the RDP settings from the registry.
    2. The TSGU client has network access to the TSGU Server:
        1. The network interface of the NAP Client is configured to operate on the local subnet.
        2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NEP is connected.
        3. All network devices between the local subnet and the NEP are configured to allow packet flow between the two entities.
        4. The routing tables in the NAP Client are configured to enable correct packet routing between the NAP Client and the NEP.
    3. The TSGU client service has been started.
- **Means of knowing requirement satisfied**:
    1. The NAP Client can successfully ping the NEP over the network.

2. The TSGU client is shown as running within the list of services.

3. An attempt to connect to a remote desktop via the Terminal Services Gateway completes successfully.

4. A sniffer trace performed during a RDP connection event, shows TSGU packets traveling between the NAP Client and the NEP. These TSGU packets must contain [MS-TSGU] fields.

5. No errors are logged by the TSGU client.

- **Consequences of not satisfying requirement**: The task is unable to send NAP request data via the [MS-TSGU] protocol.

Requirement: The CEAP peer is running and has the ability to forward [MS-CEAP] packets to the EAPE server on the Network Enforcement Point (NEP).

- **Reason for requirement**: The CEAP peer is used to send NAP request data (including the SoH) encapsulated within [MS-CEAP] packets to the NEP.

- **Means of satisfying the requirement**:

1. The CEAP peer is configured with the EAP/PEAP settings from the registry.

2. The CEAP peer has network access to the EAPE Proxy:

    1. The network interface of the NAP Client is configured to operate on the local subnet.

    2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NEP is connected.

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

    4. The routing tables in the NAP Client are configured to enable correct packet routing between the NAP Client and the NEP.

3. The CEAP peer service has been started.

- **Means of knowing requirement satisfied**:

1. The NAP Client can successfully ping the NEP over the network.

2. The EAPE peer is shown as running within the list of services.

3. An attempt to VPN to a computer on the corporate network completes successfully.

4. A sniffer trace performed during a VPN connection event, shows EAPE packets traveling between the NAP Client and the NEP. These EAPE packets must contain [MS-EAPE] fields.

5. No errors are logged by the EAPE peer.

- **Consequences of not satisfying requirement**: The task is unable to send NAP request data via the [MS-EAPE] protocol.

Requirement: The SoH client is operational and has the ability to create [MS-SoH] packets.

- **Reason for requirement**: The SoH client is used to gather health information about the NAP Client and create a SoH message, which will be forwarded to the NEP by the NAP transport protocols.

- **Means of satisfying the requirement**:

  1. The SoH client is configured with the settings described in section 2.5 SoH Settings in [MS-GPNAP].

  2. The SoH client service has been started.

  3. All required SHA plug-ins are installed and configured.

  4. All required SHA plug-ins have initialized and reported to the SoH client.

- **Means of knowing requirement satisfied**:

  1. The SoH client is shown as running within the list of services.

  2. A sniffer trace performed during a DHCP lease renewal event, shows DHCPN packets traveling between the NAP Client and the NEP containing a [MS-SoH] message.

  3. No errors are logged by the SoH client.

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

Requirement: The Authentication client is operational and has the ability to process user credential requests.

- **Reason for requirement**: The Authentication client is used to manage the credential cache on the NAP Client and provide authentication data, which will be forwarded to the Authentication Server by the NAP transport protocols.

- **Means of satisfying the requirement**:

  1. The Authentication client is configured from the registry.

  2. The Authentication service is started.

  3. The credential cache is initialized and is operational.

  4. The Authentication service is connected to the credential cache.

- **Means of knowing requirement satisfied**:

  1. The Authentication service is shown as running within the list of services.

  2. A user on the NAP client is able to login.

  3. A sniffer trace performed during a Health Certificate Enrollment event, shows HTTP packets traveling between the NAP Client and the NEP:

     1. The HTTP packets must contain an authentication blob in the Authorization field.

     2. The final server response is "200 OK".

  4. No errors are logged by the Authentication client.

- **Consequences of not satisfying requirement**: The task is unable to send authentication data to the NEP.

Requirement: The task triggers described in section 5.1.3.4 are functioning correctly.

- **Reason for requirement**: The triggers initiate the task.
- **Satisfying the requirement**:
  - Configure the OS to fire an event message to the NAP event handler for the following events:
    1. The network status of the NAP Client has changed.
    2. One of the DHCP leases expire.
    3. One of the IPSec certificates expire.
    4. The NAP Client awakens from hibernation.
    5. The HCEP client, the DHCPN client, the TSGU client or the CEAP peer signals the task that a new connection is being created.
    6. The NAP Agent signals that remediation has completed via the Process SoHR Task.
    7. The NAP Agent signals that a new authentication round is needed via the Process SoHR Task.
    8. The SoH values change due to an internal or external event. (A state change occurs in the NAP Client firewall, Microsoft Windows® Server Update Service, antivirus or antispyware software, and so on.)
    9. There is a change in the NAP configuration of the NAP Client, for example a NAP Group Policy change.
- **Verifying requirement is satisfied**:
  1. Fire each of the triggers and check if the task is executed.
  2. 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.

5.2.2 Task Relationships

5.2.2.1 Black-Box Relationship Diagrams

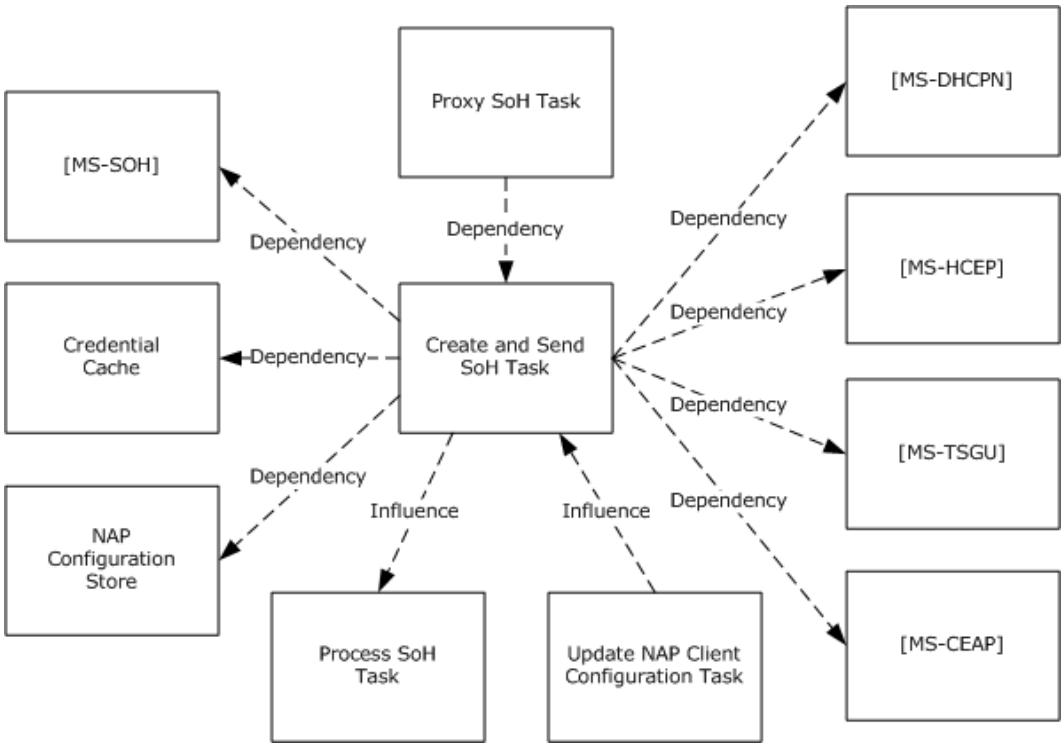

Figure 15: Create and Send SoH Task black-box relationships

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

5.2.2.2 Task Dependencies

The dependencies in the relationship diagram are as follows:

The Create and Send SoH Task depends on the clients of the NAP transport protocols (MS-HCEP, MS-TSGU, MS-CEAP and MS-DHCPN) to accept the NAP request data and to transport the data to its corresponding server on the NAP Enforcement Point.

The Create and Send SoH Task depends on the credential cache for the authentication data it sends to the clients of the NAP transport protocols (MS-HCEP, MS-TSGU, MS-CEAP and MS-DHCPN).

The Create and Send SoH Task depends on the NAP Configuration store for information on how to perform its processing steps and for configuration it sends to the clients of the NAP transport protocols (MS-HCEP, MS-TSGU, MS-CEAP and MS-DHCPN).

The Create and Send SoH Task depends on the MS-SOH client, and the MS-SOH processing rules, to create the SoH message it sends to the clients of the NAP transport protocols (MS-HCEP, MS-TSGU, MS-CEAP and MS-DHCPN).

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 send NAP request data to the NAP transport protocol (MS-HCEP, MS-TSGU, MS-CEAP and MS-DHCPN) clients, for inclusion within their protocol messages. If the Create and Send SoH Task does not send the NAP request data, the NAP Proxy has no data to process.

5.2.2.3 Task Influences

The Create and Send SoH Task is influenced by the Update NAP Client Configuration Task, which modifies the values contained in the NAP Configuration ADM.

The Create and Send SoH Task influences the Process SoH Task, as the Process SoH Task processes the NAP request data this task sends out.

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 NAP Client 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 that provide NAP client services.
- The underlying network infrastructures, such as the EC NEP 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.

5.2.4 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.

5.3 Task Architecture

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

5.3.1 Task Architectural Constraints

There is only one instance of the Create and Send SoH Task on each NAP Client and this instance initializes itself each time it starts. Different instances of this task on different NAP Clients 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.

Name	Type	Description
QecConnectList	Table with three columns: 1. QecConnection of type Object Reference (see section 5.3.3). 2. ProtocolType of type String (see section 5.3.3). 3. ToSend of type Boolean.	A lookup table indexed by the ProtocolType string.
NAP Configuration	Persistent Data Store	Defined in section 4.3.2 Task Abstract Data Model

5.3.3 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.

Name	Type	Description
QecConnection	Object Reference	A connection object reference used to identify the protocol connection (RDP connection, VPN session, etc).
ProtocolType	String	"DHCPN" = DHCPN Client "CEAP" = CEAP Peer "HCEP" = HCEP Client "TSGU" = TSGU Client

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.

This task is always run asynchronously and never returns values to the caller.

5.3.5 White-Box Relationships

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

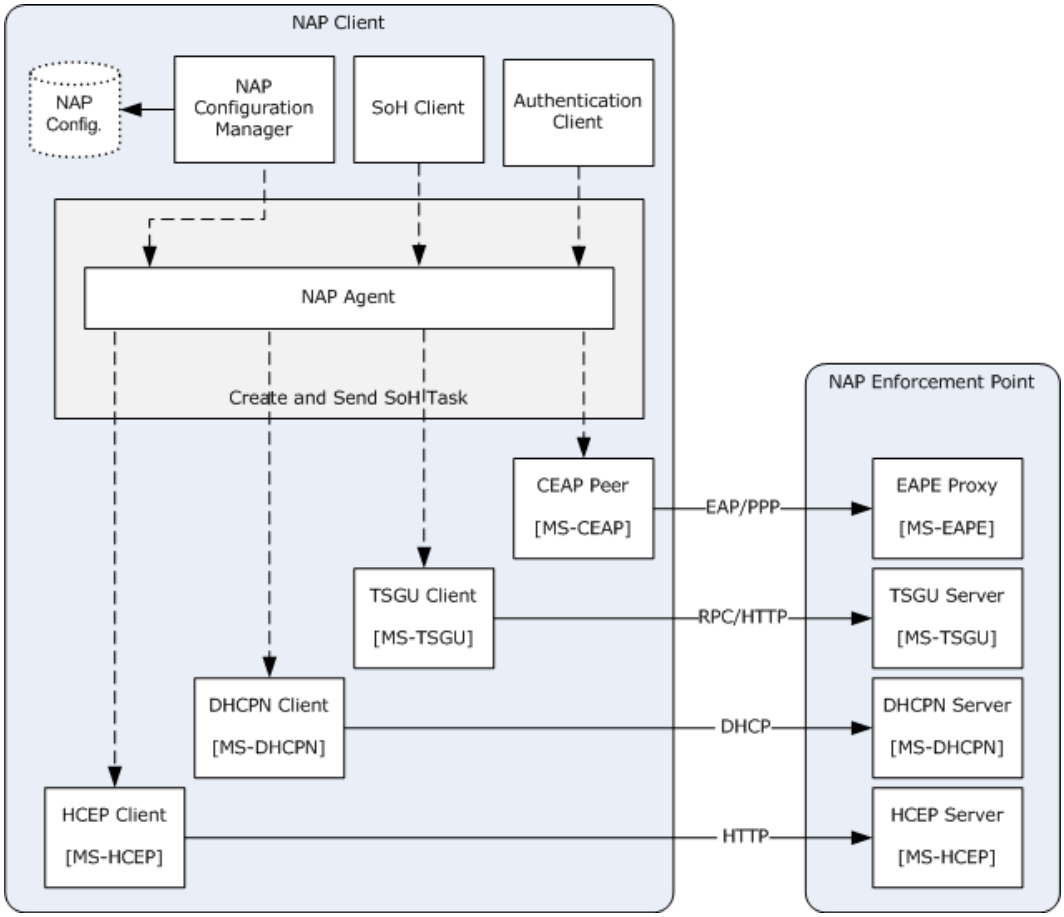

Figure 16: Create and Send SoH Task white-box relationships

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 [MS-SOH] and the encapsulating of the SoH messages by the EC into NEP-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 (NEP).

5.3.6 Task Events

5.3.6.1 Task Timers

There are no additional timers on outside entities imposed by this task other than the timers in the underlying transport system.

5.3.6.2 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 5.4.3.2.

5.3.7 Task Architecture and Communication

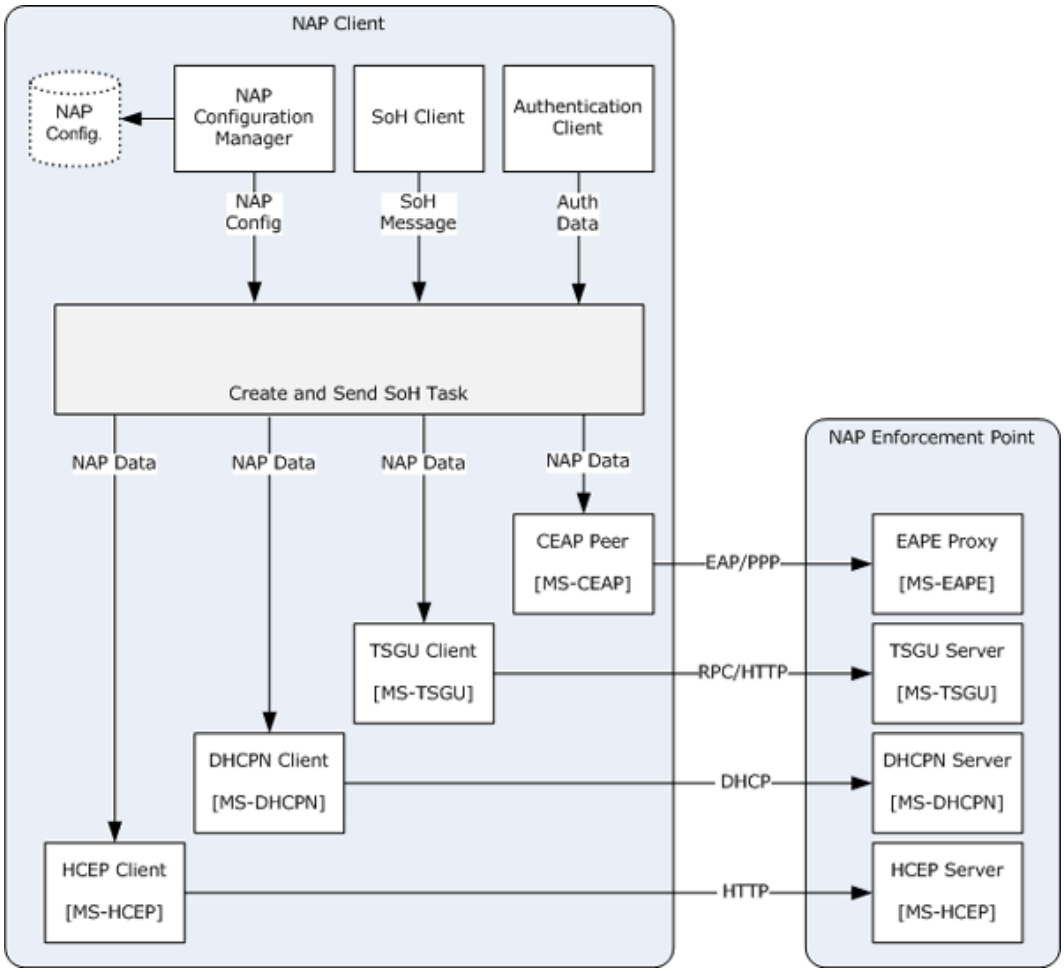

Figure 17: Create and Send SoH Task architecture and communication

5.3.8 Task Processing Rules

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

1. The task can be triggered in one of two ways:

    1. A NAP transport protocol ([MS-DHCPN], [MS-HCEP], [MS-CEAP] and [MS-TSGU]) client is triggered to create a new connection, or a DHCP lease expires in [MS-DHCPN] or a IPSec certificate expires in [MS-HCEP].

    2. A system event that is handled by the NAP Event Handler is received that pertains to an existing NAP transport protocol connection.

2. The NAP Agent requests and receives the current NAP configuration from the NAP Configuration Manager.

3. The NAP agent requests and receives authentication data from the Authentication Client.

4. The NAP Agent iterates through the QecConnectList and sets all ToSend values to false.

5. If the trigger is due to a new NAP transport protocol connection, or a DHCP lease expires or an IPSec certificate expires, the following actions occur:

    1. A thread associated with a NAP transport protocol client invokes the task after creating a QecConnection object:

        - The DHCPN Client invokes the task as described in section 3.1.7.1 DhcpClientGetSoH in [MS-DHCPN].
        - The HCEP Client invokes the task as described in section 3.1.4 Higher-Layer Triggered Events in [MS-HCEP].
        - The TSGU Client invokes the task as described in section 3.2.4 Message Processing Events and Sequencing Rules in [MS-TSGU].
        - The CEAP Peer invokes the task as described in section 3.2.5 Message Processing Events and Sequencing Rules in [MS-CEAP].

    2. The NAP Agent searches the QecConnectList for an entry that matches QecConnection. If found, the ToSend value is set to true. Otherwise, the NAP Agent creates a new entry in the QecConnectList, setting the QecConnection and ProtocolType values to the passed in parameters, and the ToSend value to true.

6. If the trigger is due to a system event received by the NAP Event Handler, the following actions occur:

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

    2. The NAP Agent processes each events as follows:

        - If the event indicates that the network status of the NAP Client has changed, the NAP Agent iterates through the QecConnectList and set the ToSend flag to true for all entries.
        - If the event indicates that the SoH Client has completed remediation, the NAP Agent iterates through the QecConnectList and set the ToSend flag to true for all entries.
        - If the event indicates that the SoH values have changed due to an internal or external event (such as firewall disabled, antivirus update, etc.), the NAP Agent iterates through the QecConnectList and set the ToSend flag to true for all entries.
        - If the event indicates that the NAP configuration has changed, the NAP Agent iterates through the QecConnectList and set the ToSend flag to true for all entries.
        - If the event indicates that the NAP Client has rebooted or awaken from hibernation, the NAP Agent iterates through the QecConnectList and set the ToSend flag to true for all the "DHCPN" entries.

7. The NAP Agent creates a unique correlation ID.

8. The NAP Agent requests and receives the current NAP configuration from the NAP Configuration Manager.

9. The NAP Agent obtains a SoH packet by calling the GetSoHRequest abstract interface in [MS-SOH].

10. The NAP agent requests and receives authentication data from the Authentication Client.

11. The NAP Agent determines which transport protocol processing is required by reviewing the Enforcement Client Settings ADM and the trigger type.

12. If the NAP Agent determines the trigger is applicable to DHCPN, and DHCPN enforcement is enabled, the NAP Agent iterates through the QecConnectList table and calls the DhcpClientNotifySoHChange abstract interface in [MS-DHCPN] on all DHCPN connections in the table.

13. If the NAP Agent determines the trigger is applicable to HCEP, and HCEP enforcement is enabled, the NAP Agent iterates through the QecConnectList table and calls the HCEPEvaluateHealthRequest abstract interface in [MS-HCEP] on all HCEP connections in the table.

14. If the NAP Agent determines the trigger is applicable to TSGU, and TSGU enforcement is enabled, the NAP Agent iterates through the QecConnectList table and calls the NotifySoHChange abstract interface in [MS-TSGU] on all TSGU connections in the table.

15. If the NAP Agent determines the trigger is applicable to CEAP, and CEAP enforcement is enabled, the NAP Agent iterates through the QecConnectList table and calls the EvaluateHealthRequest abstract interface in [MS-CEAP] on all EAP connections in the table.

5.3.9 Task Failure Scenarios

5.3.9.1 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 NEP, which may make the client machine **unhealthy**. The failures are detected by a timer monitored by a SoH Client (section 5.3.6.1). 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)**.

5.3.9.2 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 5.1.3.4. 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.

5.3.9.3 EC and NEP Communication

These failures can be caused by:

- Misconfigurations on the EC and/or NEP.
- Network connectivity issues in which the EC cannot communicate with the NEP.

If the EC cannot communicate with the NEP, 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.

5.4 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.

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 agent service is started and initialized correctly on the NAP Client.
- 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 NEP 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.

5.4.2 Task Initialization of External Entities

None.

5.4.3 Task Event Details

5.4.3.1 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.

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

5.4.3.2 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 NAP Client wakes up from sleep or hibernation, the NAP Event Handler on the NAP Client 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 NAP Client: 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 Microsoft 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 NAP Client 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.

5.4.4 Task Architectural Details

This section illustrates an example of a NAP client creating an SoH and sending it to the NEP. 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].

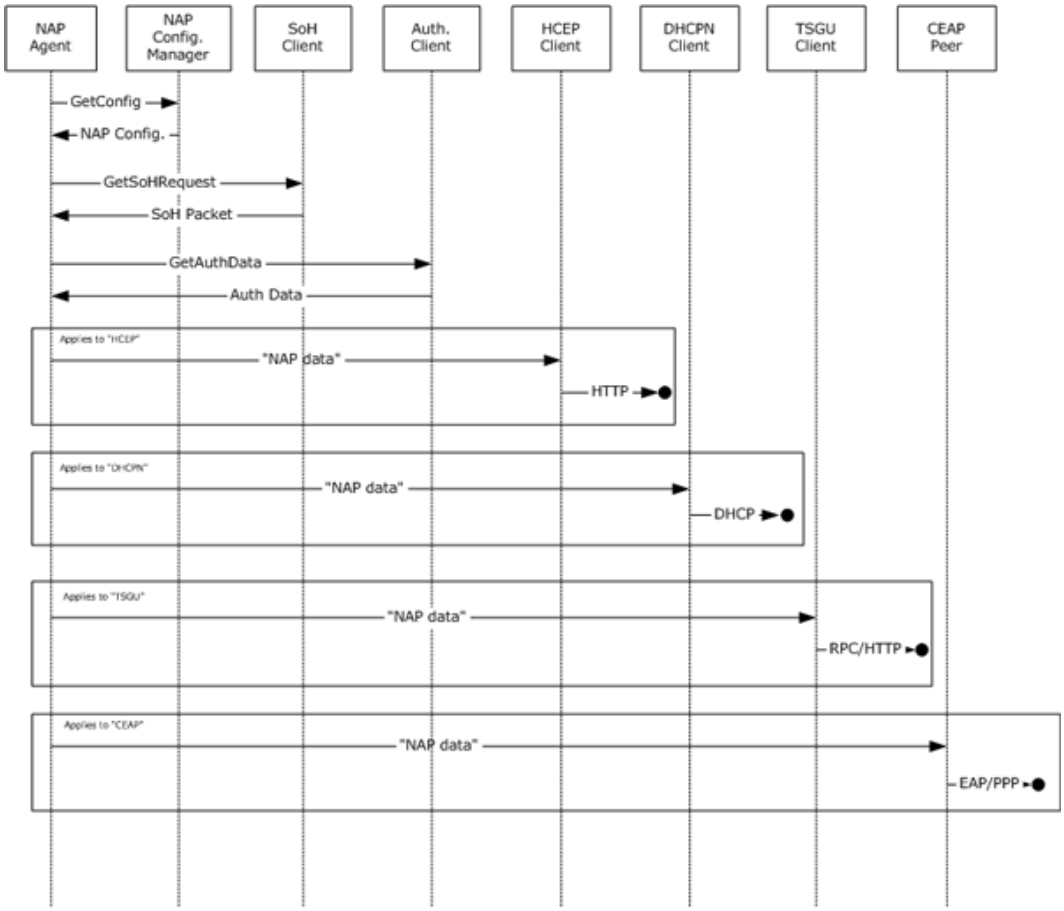

Figure 18: Sequence diagram for the main success scenario of the Create and Send SoH Task

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 Microsoft Windows® Server Update Services) from the SHAs in the **NAP Available SHAs List** ADM element.

4. The SoH Client composes an SoH message with the health information collected from the SHAs [MS-SOH] 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 NEP.

5.4.5 Task Processing Rule Details

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

1. The task can be triggered in one of two ways:
    1. A NAP transport protocol ([MS-DHCPN], [MS-HCEP], [MS-CEAP] and [MS-TSGU]) client is triggered to create a new connection, or a DHCP lease expires in [MS-DHCPN] or a IPSec certificate expires in [MS-HCEP].
    2. A system event that is handled by the NAP Event Handler is received that pertains to an existing NAP transport protocol connection.
2. The NAP Agent requests and receives the current NAP configuration from the NAP Configuration Manager.
3. The NAP agent requests and receives authentication data from the Authentication Client.
4. The NAP Agent iterates through the QecConnectList and sets all ToSend values to false.
5. If the trigger is due to a new NAP transport protocol connection, or a DHCP lease expires or an IPSec certificate expires, the following actions occur:
    1. A thread associated with a NAP transport protocol client invokes the task after creating a QecConnection object:
        - The DHCPN Client invokes the task as described in section 3.1.7.1 DhcpClientGetSoH in [MS-DHCPN].
        - The HCEP Client invokes the task as described in section 3.1.4 Higher-Layer Triggered Events in [MS-HCEP].
        - The TSGU Client invokes the task as described in section 3.2.4 Message Processing Events and Sequencing Rules in [MS-TSGU].
        - The CEAP Peer invokes the task as described in section 3.2.5 Message Processing Events and Sequencing Rules in [MS-CEAP].
    2. The NAP Agent searches the QecConnectList for an entry that matches QecConnection. If found, the ToSend value is set to true. Otherwise, the NAP Agent creates a new entry in the QecConnectList, setting the QecConnection and ProtocolType values to the passed in parameters, and the ToSend value to true.
6. If the trigger is due to a system event received by the NAP Event Handler, the following actions occur:
    1. The NAP Event Handler passes the event to the NAP Agent.
    2. The NAP Agent processes each events as follows:
        - If the event indicates that the network status of the NAP Client has changed, the NAP Agent iterates through the QecConnectList and set the ToSend flag to true for all entries.
        - If the event indicates that the SoH Client has completed remediation, the NAP Agent iterates through the QecConnectList and set the ToSend flag to true for all entries.

- If the event indicates that the SoH values have changed due to an internal or external event (such as firewall disabled, antivirus update, etc.), the NAP Agent iterates through the QecConnectList and set the ToSend flag to true for all entries.
- If the event indicates that the NAP configuration has changed, the NAP Agent iterates through the QecConnectList and set the ToSend flag to true for all entries.
- If the event indicates that the NAP Client has rebooted or awaken from hibernation, the NAP Agent iterates through the QecConnectList and set the ToSend flag to true for all the "DHCPN" entries.

7. The NAP Agent creates a unique correlation ID, as described in [MS-SoH].
8. The NAP Agent obtains a SoH packet by calling the GetSoHRequest abstract interface (see section 3.2.7.1 GetSoHRequest in [MS-SOH]) with correlationID, ShatimeoutInMsec and BackwardCompatible (see section 4.3.2) as input parameters and receives the sohRequest as an output parameter.
9. The NAP agent requests and receives authentication data from the Authentication Client.
10. The NAP Agent then iterates through the QecConnectList and processes each entry that has its ToSend flag set to true as follows:
    1. If the ProtocolType equals "DHCPN" and the DHCPN enforcement ADM is enabled, the NAP agent calls the DhcpClientNotifySoHChange abstract interface (see section 3.1.7.3 DhcpClientNotifySoHChange in [MS-DHCPN]) on the QecConnection value with following parameters: sohRequest, authentication data and correlationID.
    2. If the ProtocolType equals "HCEP" and the HCEP enforcement ADM is enabled, the NAP agent calls the HCEPEvaluateHealthRequest abstract interface (see section 3.1.4 Higher-Layer Triggered Events in [MS-HCEP]) on the QecConnection value with following parameters: sohRequest, authentication data, correlationID, 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.
    3. If the ProtocolType equals "CEAP" and the EAP enforcement ADM is enabled, the NAP agent calls the EvaluateHealthRequest abstract interface (see section 3.1.7.3 EvaluateHealthRequest in [MS-CEAP]) on the QecConnection value with following parameters: sohRequest, authentication data and correlationID.
    4. If the ProtocolType equals "TSGU" and the RDG enforcement ADM is enabled, the NAP agent calls the NotifySoHChange abstract interface (see section 3.1.7.3 DhcpClientNotifySoHChange in [MS-DHCPN]) on the QecConnection value with following parameters: sohRequest, authentication data and correlationID.

5.5 Task Security

The only security consideration for this task is in the case of HCEP EC Enabled and the NAP Client requires that the X.509 certificate use SSL as specified in [MS-TLSP]. For additional information about security considerations, see section 12, as well as the Security sections of the referenced protocol Technical Documents.

6 Proxy SoH Task

This section describes the task of proxying the NAP specific data from NAP transport protocol servers to the RNAP client. The NAP request data can arrive on a number of different transport protocols (MS-HCEP, MS-TSGU, MS-EAPE and MS-DHCPN). The NAP request data is received from the incoming transport protocol and passed to the NAP Proxy. The NAP request data is then passed by the NAP Proxy to the RNAP client, by calling abstract interfaces provided by MS-RNAP.

6.1 Task Overview

6.1.1 Task Purpose

The purpose of this task is to proxy NAP request data from the NAP transport protocol ([MS-DHCPN], [MS-HCEP], [MS-TSGU] and [MS-EAPE]) servers to the RNAP client.

6.1.2 Task Applicability

This task is used when NAP specific data is received by the NAP Proxy. This task is not applicable if a NAP System is not deployed.

6.1.3 Task Use Cases

6.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Proxy SoH Task are as follows:

HCEP Server: This protocol server is used to process HCEP Protocol [MS-HCEP] messages received from an HCEP client on the NAP Client. It also acts as a policy enforcement point, using protocol specific behaviors in its enforcement. The interest of this actor in the task is that the NAP request data passed to the task is processed.

DHCPN Server: This protocol server is used to process (DHCP) Extensions for NAP [MS-DHCPN] messages received from a DHCPN client on the NAP Client. It also acts as a policy enforcement point, using protocol specific behaviors in its enforcement. The interest of this actor in the task is that the NAP request data passed to the task is processed.

TSGU Server: This protocol server is used to process TSGU Protocol [MS-TSGU] messages received from a TSGU client on the NAP Client. It also acts as a policy enforcement point, using protocol specific behaviors in its enforcement. The interest of this actor in the task is that the NAP request data passed to the task is processed.

EAPE Proxy: This protocol server is used to process EAPE Protocol [MS-EAPE] messages received from an EAPE peer on the NAP Client. It also acts as a policy enforcement point, using protocol specific behaviors in its enforcement. The interest of this actor in the task is that the NAP request data passed to the task is processed.

NAP Proxy: The NAP Proxy is used to proxy NAP request data (including the SoH) passed into the task abstract parameters by the NAP transport protocols to the RNAP Client. The NAP Proxy's interest in the task is that the task parameters are passed on to a [MS-RNAP] abstract interface for valid NAS server types.

SoH Client: The purpose of this actor is to utilize the processing rules defined in [MS-SOH] to create SoH packets, which will be consumed by the SoH Server to determine the health of the NAP Client. The SoH Client's interest in the task is that any proxied NAP request data includes the [MS-SOH] message created in the Create and Send SoH Task.

WSH Client: The purpose of this actor is to utilize the processing rules defined in [MS-WSH] to create SoHEntry packets, which will be consumed by the WSH Server. The WSH Client's interest in the task is that any proxied NAP request data includes the [MS-WSH] message created in the Create and Send SoH Task.

6.1.3.2 Supporting Actors and Task Interests Summary

RNAP client: This protocol client is used to send Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] messages to the NAP health policy server. This task employs this actor whenever NAP request data must be sent to the NAP health policy server via [MS-RNAP].

6.1.3.3 Use Case Diagrams

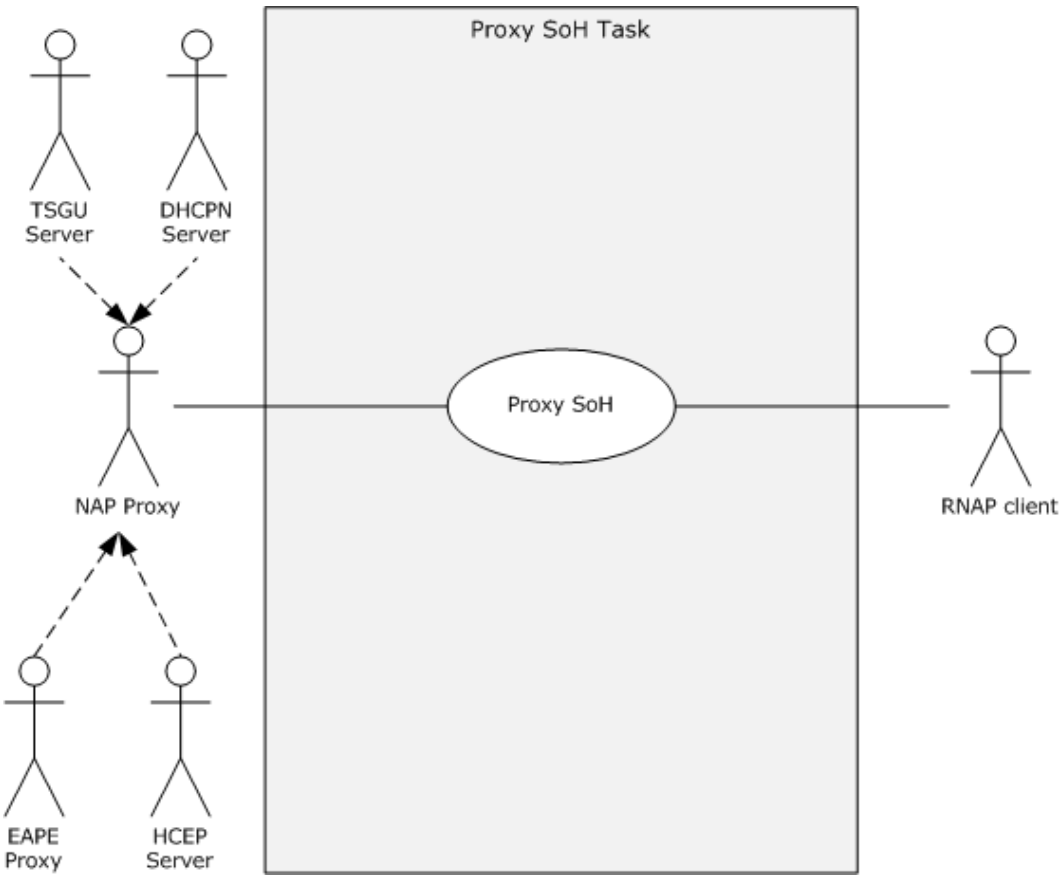

Figure 19: Proxy SoH Task use case diagram

6.1.3.4 Use Case: Proxy SoH - NAP Proxy

Goal: To proxy the NAP request data from the DHCPN Server, the HCEP Server or the TSGU Server to the RNAP Client.

Context of Use: This use case is used when NAP request data is passed by a protocol server to the NAP Proxy. The direct actor is internal to the task.

Direct Actor: This role is performed by the NAP Proxy.

Primary Actor: This role is performed by four actors: the DHCPN Server, the HCEP Server the TSGU Server and the EAPE Proxy.

Supporting Actors:

RNAP client: This protocol client is used to send Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] messages to the NAP health policy server. This use case employs this actor whenever NAP request data must be sent to the NAP health policy server via [MS-RNAP].

Stakeholders and Interests: The stakeholders are defined as follows:

SoH Client: The purpose of this actor is to utilize the processing rules defined in [MS-SOH] to create SoH packets, which will be consumed by the SoH Server to determine the health of the NAP Client. The SoH Client's interest in the use case is that any proxied NAP request data includes the [MS-SOH] message created in the Create and Send SoH Task.

WSH Client: The purpose of this actor is to utilize the processing rules defined in [MS-WSH] to create SoHEntry packets, which will be consumed by the WSH Server. The WSH Client's interest in the use case is that any proxied NAP request data includes the [MS-WSH] message created in the Create and Send SoH Task.

Precondition: The NAP transport protocol ([MS-DHCPN], [MS-HCEP] and [MS-TSGU]) server successfully receive the transport protocol message and extracted the NAP request data from it.

Minimal Guarantees:

- The use case will always process the task abstract parameters passed to it.
- A [MS-RNAP] abstract interface will be invoked, with properly mapped parameters, for valid NAS server types.
- The [MS-SOH] message, if present, will be proxied with the rest of the NAP data.
- The [MS-WSH] message, if present, will be proxied with the rest of the NAP data.

Success Guarantee: A new entry is created in the NasTypeTable and the NAP request data is sent to the RNAP Client.

Trigger: The Task can be triggered by any of the following:

- A DHCPN Server thread invokes the abstract interface of this task.
- A HCEP Server thread invokes the abstract interface of this task.
- A TSGU Server thread invokes the abstract interface of this task.
- An EAPE Proxy thread invokes the abstract interface of this task.

Main Success Scenario:

1. The task is triggered by a thread associated with one of the following direct actors invoking the task's abstract interface:
   - The DHCPN Server.
   - The HCEP Server.
   - The TSGU Server.

- The EAPE Proxy.

2. The NAP Proxy receives the task abstract parameters and records a reference to the caller thread.
3. The NAP Proxy invokes an abstract interface on the RNAP Client based on the passed on the NAS server type.
4. The NAP Proxy passes values to the invoked [MS-RNAP] abstract interface by mapping the task abstract parameters to the parameter listing of the [MS-RNAP] abstract interface.
5. The NAP Proxy creates an entry in the NasTypeTable using the passed in correlation ID and caller reference.

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 Proxy in an environment where the NAP request data is transferred from the NAP transport protocol ([MS-DHCPN], [MS-HCEP], [MS-TSGU] and [MS-EAPE]) servers to the RNAP client. The environment should meet the following requirement to support this task.

Requirement: The HCEP server is running and has the ability to receive [MS-HCEP] packets from the HCEP client on the NAP Client.

- **Reason for requirement**: The HCEP server is used to receive NAP request data (including the SoH) encapsulated within [MS-HCEP] packets from the NAP Client.
- **Means of satisfying the requirement**:

  1. The HCEP server is configured with the HCEP server settings from the registry.
  2. The HCEP server has network access to the HCEP client:

     1. The network interface of the NEP is configured to operate on the local subnet.
     2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NAP Client is connected.
     3. All network devices between the local subnet and the NAP Client are configured to allow packet flow between the two entities.
     4. The routing tables in the NEP are configured to enable correct packet routing between the NAP Client and the NEP.
  3. The HCEP server service has been started.

- **Means of knowing requirement satisfied**:

  1. The NEP can successfully ping the NAP Client over the network.
  2. The HCEP server is shown as running within the list of services.

3. A sniffer trace performed during a Health Certificate Enrollment event, shows HTTP packets traveling between the NAP Client and the NEP. These HTTP packets must contain [MS-HCEP] fields.

4. No errors are logged by the HCEP server.

- **Consequences of not satisfying requirement**: The task is unable to receive NAP request data via the [MS-HCEP] protocol.

Requirement: The DHCPN server is running and has the ability to receive [MS-DHCPN] packets from the DHCP client on the NAP Client.

- **Reason for requirement**: The DHCPN server is used to receive NAP request data (including the SoH) encapsulated within [MS-DHCPN] packets from the NAP Client.

- **Means of satisfying the requirement**:

1. The DHCPN server is configured with the DHCP settings from the registry.

2. The DHCPN server has network access to the DHCP client:

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

    2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NAP Client is connected.

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

    4. The routing tables in the NEP are configured to enable correct packet routing between the NAP Client and the NEP.

3. The DHCPN server service has been started.

- **Means of knowing requirement satisfied**:

1. The NEP can successfully ping the NAP Client over the network.

2. The DHCPN client is shown as running within the list of services.

3. An attempt to renew the IP address lease on each NIC via the DHCP Server on the NEP completes successfully.

4. A sniffer trace performed during a DHCP lease renewal event, shows DHCP packets traveling between the NAP Client and the NEP. These DHCP packets must contain [MS-DHCPN] fields.

5. No errors are logged by the DHCPN server.

- **Consequences of not satisfying requirement**: The task is unable to receive NAP request data via the [MS-DHCPN] protocol.

Requirement: The TSGU server is running and has the ability to receive [MS-TSGU] packets from the TSGU client on the NAP Client.

- **Reason for requirement**: The TSGU client is used to receive NAP request data (including the SoH) encapsulated within [MS-TSGU] packets from the NAP Client.

- **Means of satisfying the requirement**:

1. The TSGU server is configured with the TSG settings from the registry.
2. The TSGU server has network access to the TSGU client:
    1. The network interface of the NEP is configured to operate on the local subnet.
    2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NAP Client is connected.
    3. All network devices between the local subnet and the NAP Client are configured to allow packet flow between the two entities.
    4. The routing tables in the NEP are configured to enable correct packet routing between the NAP Client and the NEP.
3. The TSGU server service has been started.

- **Means of knowing requirement satisfied**:
    1. The NEP can successfully ping the NAP Client over the network.
    2. The TSGU server is shown as running within the list of services.
    3. An attempt to connect to a remote desktop via the Terminal Services Gateway on the NEP completes successfully.
    4. A sniffer trace performed during a RDP connection event, shows TSGU packets traveling between the NAP Client and the NEP. These TSGU packets must contain [MS-TSGU] fields.
    5. No errors are logged by the TSGU server.
- **Consequences of not satisfying requirement**: The task is unable to receive NAP request data via the [MS-TSGU] protocol.

Requirement: The EAPE proxy is running and has the ability to receive [MS-EAPE] packets from the EAPE peer on the NAP Client.

- **Reason for requirement**: The EAPE proxy is used to receive NAP request data (including the SoH) encapsulated within [MS-EAPE] packets from the NAP Client.
- **Means of satisfying the requirement**:
    1. The EAPE proxy is configured with the EAP/PEAP settings from the registry.
    2. The EAPE proxy has network access to the EAPE client:
        1. The network interface of the NAP Client is configured to operate on the local subnet.
        2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NEP is connected.
        3. All network devices between the local subnet and the NEP are configured to allow packet flow between the two entities.
        4. The routing tables in the NAP Client are configured to enable correct packet routing between the NAP Client and the NEP.
    3. The EAPE peer service has been started.

- **Means of knowing requirement satisfied**:

  1. The NEP can successfully ping the NAP Client over the network.

  2. The EAPE proxy is shown as running within the list of services.

  3. An attempt to VPN to a computer on the corporate network via the NEP completes successfully.

  4. A sniffer trace performed during a VPN connection event, shows EAPE packets traveling between the NAP Client and the NEP. These EAPE packets must contain [MS-EAPE] fields.

  5. No errors are logged by the EAPE proxy.

- **Consequences of not satisfying requirement**: The task is unable to receive NAP request data via the [MS-EAPE] protocol.

Requirement: The RNAP Client is running and has the ability to send [MS-RNAP] packets to the RNAP server on the NAP Health Policy Server (NPS).

- **Reason for requirement**: The RNAP client is used to send NAP request data (including the SoH) encapsulated within [MS-RNAP] packets to the NPS.

- **Means of satisfying the requirement**:

  1. The RNAP client is configured with the RADIUS settings from the registry.

  2. The RNAP client has network access to the RNAP Server:

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

     2. The physical network path (network devices, Ethernet cables, etc.) 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 routing tables in the NEP are configured to enable correct packet routing between the NPS and the NEP.

  3. The RNAP client service has been started.

- **Means of knowing requirement satisfied**:

  1. The NEP can successfully ping the NPS over the network.

  2. The RNAP client is shown as running within the list of services.

  3. A sniffer trace performed during a RADIUS access-request event, shows RADIUS packets traveling between the NPS and the NEP. These must be "access-accept" RADIUS packets containing [MS-RNAP] fields.

  4. No errors are logged by the RNAP client.

- **Consequences of not satisfying requirement**: The task is unable to send NAP request data via the [MS-RNAP] protocol.

6.2.2 Task Relationships

6.2.2.1 Black-Box Relationship Diagrams

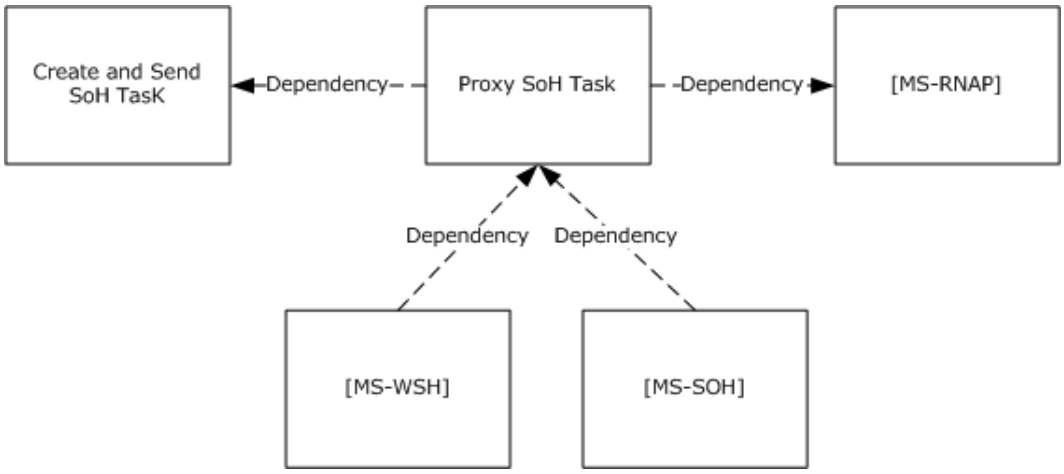

Figure 20: Proxy SoH Task black-box relationships

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

6.2.2.2 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 send NAP request data to the NAP transport protocol (MS-HCEP, MS-TSGU, MS-EAPE and MS-DHCPN) clients, for inclusion within their protocol messages. If the Create and Send SoH Task does not send the NAP request data, the NAP Proxy has no data to process.

The Proxy SoH Task depends on the MS-RNAP client to receive the NAP request data and to transport the data to the corresponding MS-RNAP server on the NAP Health Policy Server.

The MS-SoH protocol is dependent on the Proxy SoH Task as the communication between the MS-SoH client and the MS-SoH server is totally reliant on the proxy steps executed by the Proxy SoH Task.

The MS-WSH protocol is dependent on the Proxy SoH Task as the communication between the MS-WSH client and the MS-WSH server is totally reliant on the proxy steps executed by the Proxy SoH Task.

6.2.2.3 Task Influences

None.

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 of the NAP Enforcement Point computer are trustworthy.
- All NAP enforcement servers are available, correctly configured, and functioning correctly.

- Authentication information was transferred successfully by the underlying transport protocols.

6.2.4 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.

6.3 Task Architecture

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

6.3.1 Task Architectural Constraints

There can be more than one instance of the Proxy SoH Task if multiple NEP 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.

6.3.2 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.

Name	Type	Description
NasTypeTable	Table with two columns: 1. NasRef of type Caller Reference (thread ID, socket, callback pointer, etc.) 2. CorrId of type 24-byte GUID	A lookup table indexed by the correlation ID of the SoH message being proxied.

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.

Name	Type	Description
SoHRequest	[MS-SOH] section 2.2.5	A SoH message created by the SoH client.

Name	Type	Description
SoHReqLength	DWORD	Length, in bytes, of the SoH message.
CorrelationId	24-byte GUID	A correlation ID containing a unique transaction identifier shared across SoH and SoHR messages.
AuthData	binary BLOB	Binary blob containing user credentials from the authentication client.
ClientName	String	The Netbios name of the NAP Client generating the request.
MachineName	ANSI String	FQDN of the NAP Client.
ClientIPv4Address	DWORD	IPv4 address of the NAP Client.
ClientIpv6Address	16 Byte Array	IPv6 address of the NAP Client (if available).
DhcpClientLeaseOffer	DWORD	IP address the DHCP server will offer the NAP Client.
DhcpServiceClass	String	The DHCP scope corresponding to the DhcpClientLease.
NASIdentifier	String	Name of the DHCP server
SecurityIdentity	[MS-DTYP] section 2.4.2	The security-identifier (SID) of the user requesting access.
CallingStationID	String	Unique ID of NAP Client (usually IP Address or MAC).
eapBlob	binary BLOB	A binary BLOB that contains all the EAP protocol layers, from the outer EAP (MS-EAPE) to the inner EAP (MS-CEAP).
eapBlobLength	DWORD	Length, in bytes, of the eapBlob.
eapBlobSignature	Binary Data	Signature of EAPBlob
eapBlobSigLength	DWORD	Length, in bytes, of the eapBlobSignature.
NasServerType	String	"DHCPN" = DHCPN Server "EAPE" = EAPE Proxy "HCEP" = HCEP Server "TSGU" = TSGU Server

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 always run asynchronously and never returns values to the caller.

6.3.5 White-Box Relationships

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

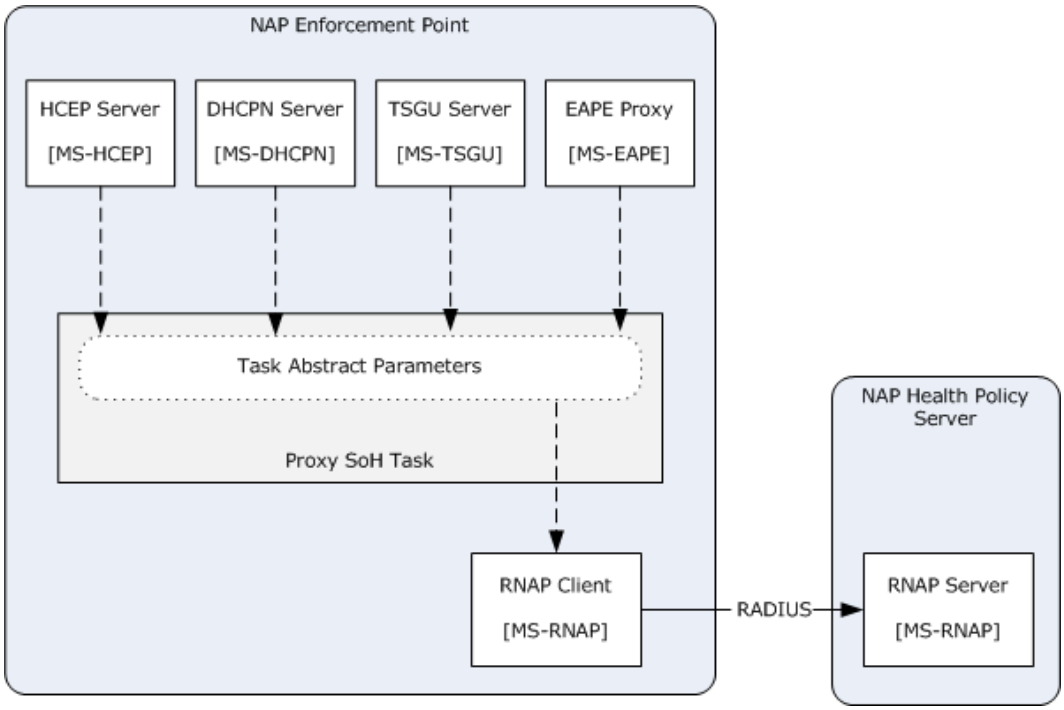

Figure 21: Proxy SoH Task white-box relationships

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.

6.3.6 Task Events

6.3.6.1 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.

6.3.6.2 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.

6.3.7 Task Architecture and Communication

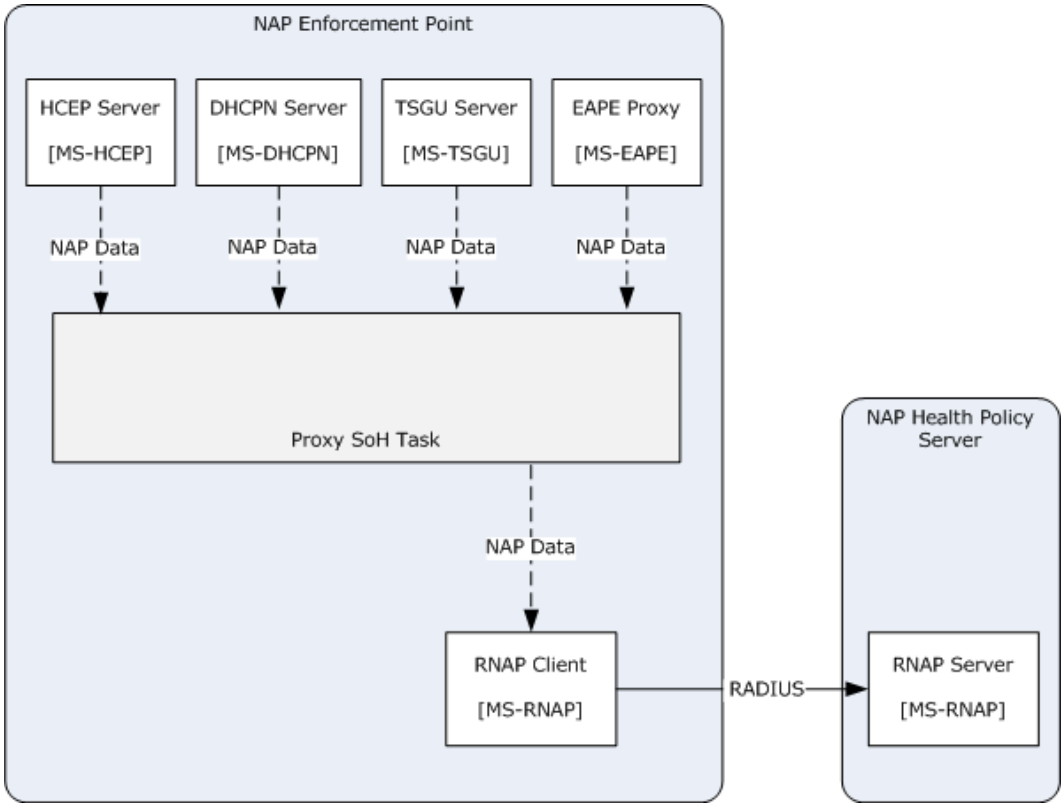

Figure 22: Proxy SoH Task architecture and communication

6.3.8 Task Processing Rules

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

1. The task is triggered by a thread associated with one of the NAP transport protocol servers invoking the task's abstract interface:
   - The DHCPN Server invokes the task with parameters specific to [MS-DHCPN].
   - The HCEP Server invokes the task with parameters specific to [MS-HCEP].
   - The TSGU Server invokes the task as with parameters specific to [MS-TSGU].
   - The EAPE Proxy invokes the task with parameters specific to [MS-EAPE].
2. If NasServerType equals "DHCPN", the NAP Proxy calls the SendDhcpAccessRequest abstract interface in [MS-RNAP].
3. If NasServerType equals "HCEP", the NAP Proxy calls the SendHcepAccessRequest abstract interface in [MS-RNAP].
4. If NasServerType equals "TSGU", the NAP Proxy calls the SendTsguAccessRequest abstract interface in [MS-RNAP].

5. If NasServerType equals "EAPE", the NAP Proxy calls the SendEapeAccessRequest abstract interface in [MS-RNAP].

6. The NAP Proxy creates a new entry in the NasTypeTable.

If an error is raised at any stage of the Proxy SoH Task, the NAP Proxy logs an error and exits.

6.3.9 Task Failure Scenarios

6.3.9.1 NAP Health Policy Server and NAP Enforcement Point Communication

These failures can be caused by:

- Misconfigurations on the NAP Enforcement Point.
- Network connectivity issues wherein the NAP health policy server cannot communicate with the NAP Enforcement Point.

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

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:

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

6.4.2 Task Initialization of External Entities

None.

6.4.3 Task Event Details

6.4.3.1 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].

6.4.3.2 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].

6.4.4 Task Architectural Details

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

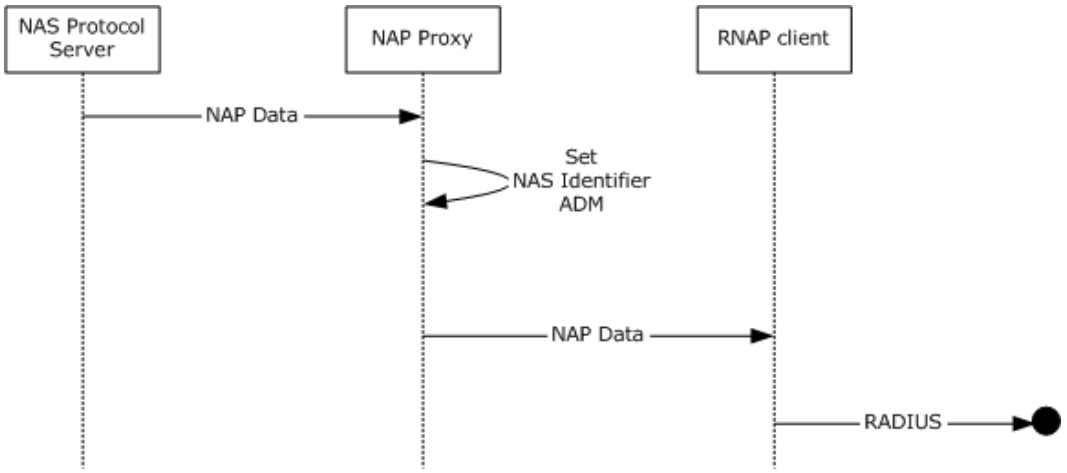

Figure 23: Sequence diagram for the main success scenario of the Proxy SoH Task

1. The NAP Enforcement Point receives the SoH message and a NAP Enforcement Point Channel enumerator.
2. The NAP Enforcement Point sets the **NetworkAccessServerType** ADM element (section 6.3.2) according to the transport protocol that was used to send the SoH message.
3. Based on the value of **NetworkAccessServerType**, the NAP Enforcement Point sends the SoH message to the RNAP client.

6.4.5 Task Processing Rule Details

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

1. One of the following occurs by a thread associated with a NAP transport protocol server to trigger the task:
   - The DHCPN Server invokes the task as described in section 3.2.7.1 DHCPServerProcessSoH in [MS-DHCPN].
   - The HCEP Server invokes the task as described in section 3.2.5.2 Processing an HCEP Request in [MS-HCEP].
   - The TSGU Server invokes the task as described in section 3.1.4.1.2 TsProxyAuthorizeTunnel (Opnum 2) in [MS-TSGU].
   - The EAPE Proxy invokes the task as described in section 6.5.2 Receiving an EAP-Response in [MS-EAPE].
2. If NasServerType equals "DHCPN", the NAP Proxy calls the SendDhcpAccessRequest abstract interface (section 3.3.4.1 Sending a DHCP SOH request) in [MS-RNAP] with the following parameters:
   - SoHRequest mapped to as SoH,
   - SoHReqLength mapped to SoHLength,
   - CorrelationId mapped to rasCorrelationId,

- MachineName mapped to netbios-name,
- ClientName mapped to client-unique id,
- NASIdentifier mapped to dhcp-server-name,
- DhcpClientLeaseOffer mapped to dhcp-lease-offer.

3. If NasServerType equals "HCEP", the NAP Proxy calls the SendHcepAccessRequest abstract interface (section 3.3.4.2 Sending a HCEP SOH request) in [MS-RNAP] with the following parameters:
   - SoHRequest mapped to as SoH,
   - SoHReqLength mapped to SoHLength,
   - CorrelationId mapped to rasCorrelationId,
   - ClientName mapped to clientName,
   - 0x00000005 as the value for networkAccessServerType,
   - ClientIPv4Address mapped to userIpv4Address.
4. If NasServerType equals "TSGU", the NAP Proxy calls the SendTsguAccessRequest abstract interface (section 3.3.4.3 Sending a TSGU SOH request) in [MS-RNAP] with the following parameters:
   - SoHRequest mapped to as SoH,
   - SoHReqLength mapped to SoHLength,
   - CorrelationId mapped to rasCorrelationId,
   - ClientName mapped to clientName,
   - MachineName mapped to machineName.
5. If NasServerType equals "EAPE", the NAP Proxy calls the SendEapeAccessRequest abstract interface (section 3.3.4.4 Sending a EAPE SOH request) in [MS-RNAP] with the following parameters:
   - SoHRequest mapped to SoH,
   - SoHReqLength mapped to SoHLength,
   - CorrelationId mapped to rasCorrelationId,
   - ClientName mapped to clientName,
   - MachineName mapped to machineName.
6. If NasServerType does not equal "DHCPN", "HCEP", "TSGU" or "EAPE", the NAP Proxy logs an error message and stops execution.
7. The NAP Proxy creates a new entry in the NasTypeTable, using the CorrelationId parameter as the CorrId and a reference to the calling thread as the NasRef parameter.

6.5 Task Security

The NAP Enforcement Point and the NAP health policy server must maintain a trust relationship. For additional information about security considerations, see section 12, as well as the Security sections of the referenced protocol Technical Documents.

7 Connect to NPS Task

This section describes the task of executing the connection policies on the NAP health policy server by the Policy Engine, which determine how to authenticate the connection request. The protocols that can be used to accomplish this task are specified in [MS-RNAP] and [MS-EAPE].

7.1 Task Overview

7.1.1 Task Purpose

The purpose of this task is to execute connection policies whenever messages are received from the RNAP Server [MS-RNAP] or the EAPE Server [MS-EAPE].

7.1.2 Task Applicability

This task is used when the Policy Engine receives NAP request data from either the EAPE Server [MS-EAPE] or the RNAP Server [MS-RNAP]. For EAPE messages, the RNAP server forwards the NAP request data to the EAPE Server for de-encapsulation of the SoH packet. This task is not applicable if the NAP System is not deployed.

7.1.3 Task Use Cases

7.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Receive SoH Task are as follows:

Policy Engine: Responsible for executing NAP Health Policy Server policies (Connection, Health and Network) based on the NAP data. In response to the execution of those policies, the Policy Engine generates NAP request data to be returned to the RNAP Server [MS-RNAP] or the EAPE Server [MS-EAPE], depending on the caller. The Policy Engine's interest in this particular task is to process the connection policies.

RNAP Server: This protocol server is used to process RNAP Protocol [MS-RNAP] messages received from a RNAP client on the NAP Enforcement Point. The interest of this actor in the task is that the NAP request data passed to the task is processed.

EAPE Server: This protocol server is used to process EAPE Protocol [MS-EAPE] messages received from the RNAP Server. The interest of this actor in the task is that the NAP request data passed to the task is processed.

Network Administrator: The Network Administrator wants to limit the computers connected to the corporate network to those verified as healthy. The Network Administrator also wants to control the network access of healthy computer. The Network Administrator creates and deploys policies to implement these restrictions. The Network Administrator's interests in this specific task are that only its deployed (enabled) connection policies are executed.

7.1.3.2 Supporting Actors and Task Interests Summary

Authentication Server: This actor is used to authenticate the NAP entities, such as the user on the NAP Client attempting to access network resources or the NAS client trying to connect to the NAP Health Policy Server. The Authentication Server can either perform authentication locally using the RADIUS Server implementation, or can send the information off-box to an Active Directory Server. The task employs this actor whenever authentication data needs to be processed.

Policy DB Manager: This actor maintains the persistent data store containing the policy configuration. The policy configuration contains the Connection Policy ADM elements documented in section 7.3.2 Task Abstract Data Model. The policy configuration also contains the Health Policy and Network Policy ADM elements documented in section 8.3.2 Task Abstract Data Model. The task contacts the Policy DB Manager whenever it needs policy configuration values from the data store.

7.1.3.3 Use Case Diagrams

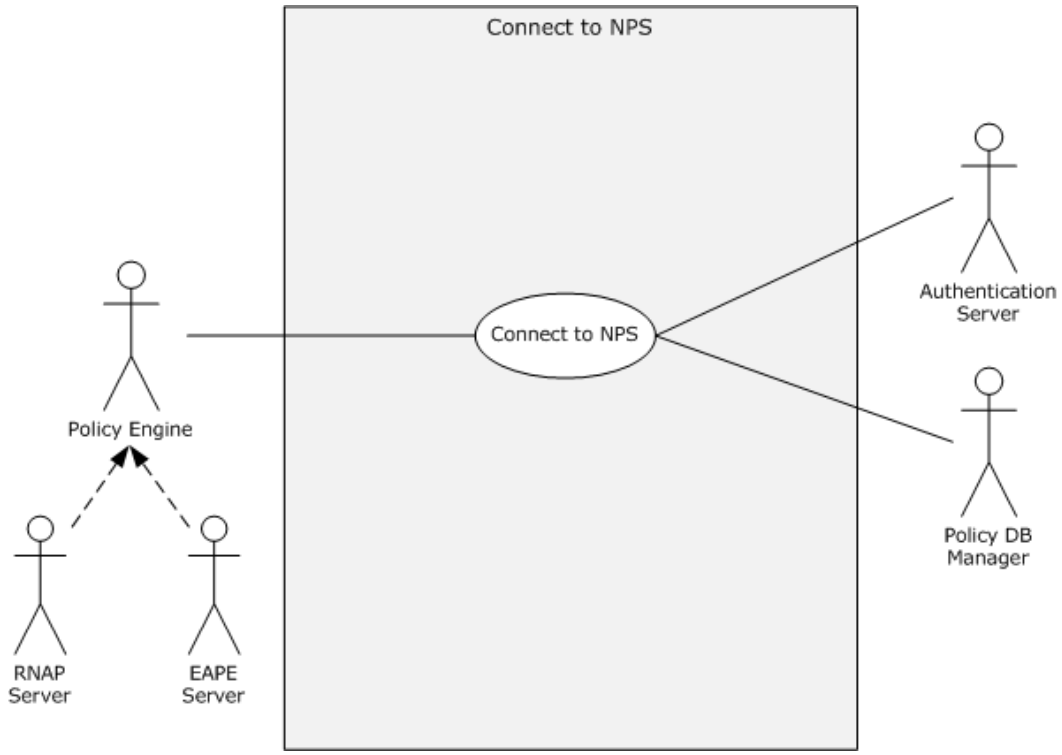

Figure 24: Connect to NPS use case diagram

7.1.3.4 Use Case: Connect to NPS -- Policy Engine

This use case is associated with the use case diagram in section 7.1.3.3.

Goal: To authenticate the connection request, using the Connection Policies in conjunction with the NAP data, and pass the NAP request data to the Process SoH Task.

Context of Use: This use case is used when NAP request data is passed to the Policy Engine by either the EAPE Server [MS-EAPE] or the RNAP Server [MS-RNAP]. The direct actor is internal to the task.

Direct Actor: This role is performed by the Policy Engine.

Primary Actor: This role is performed by two actors: the **RNAP** Server or the EAPE Server.

Supporting Actors:

Authentication Server: This actor is used to authenticate the NAP entities, such as the user on the NAP Client attempting to access network resources or the NAS client trying to connect to the NAP

Health Policy Server. The Authentication Server can either perform authentication locally using the RADIUS Server implementation, or can send the information off-box to an Active Directory Server. The use case employs this actor whenever authentication data needs to be processed.

Policy DB Manager: This actor maintains the persistent data store containing the policy configuration. The policy configuration contains the Connection Policy ADM elements documented in section 7.3.2 Task Abstract Data Model. The policy configuration also contains the Health Policy and Network Policy ADM elements documented in section 8.3.2 Task Abstract Data Model. The use case contacts the Policy DB Manager whenever it needs policy configuration values from the data store.

Stakeholders and Interests: The stakeholders are defined as follows:

Network Administrator: The Network Administrator wants to limit the computers connected to the corporate network to those verified as healthy. The Network Administrator also wants to restrict which NAS are able to connect to the NAP Health Policy Server. The Network Administrator creates and deploys policies to implement these restrictions. The Network Administrator's interests in this specific task are that the Network Administrator's connection policies are executed.

Preconditions:

- The NAP Health Policy Server components on the server are configured and working correctly.
- The integrity of the Policy DB is intact.
- The Network Administrator has deployed one or more connection policies.

Minimal Guarantees:

- The use case will always process the task abstract parameters passed to it.
- The connection policies will be processed.
- Only the connection policies deployed by the Network Administrator will be executed.
- The connection policies will always use the passed in NAP data.

Success Guarantee: The connection request is authenticated based on the connection policies and the NAP request data is passed to the Process SoH Task.

Trigger: The Task can be triggered by any of the following:

- A RNAP Server invokes the abstract interface of this task.
- An EAPE Server invokes the abstract interface of this task.

Main Success Scenario:

1. The task is triggered by one of the following direct actors invoking the task's abstract interface:
   - The RNAP Server.
   - The EAPE Server.
2. The Policy Engine receives the task abstract parameters.
3. The Policy Engine retrieves the connection policies from the Policy DB Manager.
4. The Policy Engine iterates through the each Connection Policy, doing the following:

- Check if the Connection Policy is enabled (deployed). If not, skip to next policy.
- Using the NAP request data as input, evaluate the Connection Policy conditions.
- If the Connection Policy conditions do not evaluate to true, skip to next policy.
- If the Connection Policy conditions evaluate to true, note the Connection Policy settings and stop processing any more connection policies.

5. The Policy Engine processes the Connection Policy settings to create a set of authentication settings describing how connection requests are processed (local or remote, which authentication method, etc).
6. The Policy Engine passes AuthData and the authentication settings to the Authentication Server, which in turn performs authentication.
7. The Authentication Server returns its authentication results to the Policy Engine.
8. The Policy Engine invokes the Process SoH Task using the NAP request data as the input parameters.

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 Policy Engine in an environment where the NAP request data forwarded by a NEP computer using RNAP [MS-RNAP] has arrived at the server. The environment should meet the following requirement to support this task.

Requirement: The RNAP server is running and has the ability to receive [MS-RNAP] packets from the RNAP client on the Network Enforcement Point (NEP).

- **Reason for requirement**: The RNAP server is used to receive NAP request data (including the SoH) encapsulated within [MS-RNAP] packets from the NEP.
- **Means of satisfying the requirement**:

  1. The RNAP server is configured with the RADIUS settings from the registry.
  2. The RNAP server has network access to the RNAP client:
     1. The network interface of the NPS is configured to operate on the local subnet.
     2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NEP is connected.
     3. All network devices between the local subnet and the NEP are configured to allow packet flow between the two entities.
     4. The routing tables in the NPS are configured to enable correct packet routing between the NPS and the NEP.
  3. The RNAP server service has been started.

- **Means of knowing requirement satisfied**:

  1. The NPS can successfully ping the NEP over the network.

  2. The RNAP server is shown as running within the list of services.

  3. A sniffer trace performed during a RADIUS access-request event, shows RADIUS packets traveling between the NPS and the NEP. These must be "access-accept" RADIUS packets containing [MS-RNAP] fields.

  4. No errors are logged by the RNAP server.

- **Consequences of not satisfying requirement**: The task is unable to receive NAP request data via the [MS-RNAP] protocol. As a result, the task will never be invoked.

Requirement: The EAPE server is running and has the ability to receive NAP request data from the RNAP server on the NAP Health Policy Server (NPS).

- **Reason for requirement**: The EAPE client is used to receive NAP request data (including the EAP blob), de-encapsulate the SoH message from the EAP blob and pass all the NAP request data to this task.

- **Means of satisfying the requirement**:

  1. The EAPE server is configured with settings from the registry.

  2. The RNAP server service has been started.

  3. The EAPE server service has been started.

- **Means of knowing requirement satisfied**:

  1. The RNAP server is shown as running within the list of services.

  2. The EAPE server is shown as running within the list of services.

  3. A sniffer trace performed during a VPN connection event, shows EAPE packets traveling between the NAP Client, the NEP and the NPS (over RADIUS). These EAPE packets must contain [MS-EAPE] fields.

  4. No errors are logged by the EAPE server.

- **Consequences of not satisfying requirement**: The task is unable to receive NAP request data via the [MS-EAPE] protocol.

Requirement: The NAP Policy Database is uncorrupted and accessible.

- **Reason for requirement**: The Policy DB Manager will need to read the NAP Policy Database.

- **Means of satisfying the requirement**:

  1. The Policy DB Manager is configured with the path and security settings to access the NAP Policy Database.

  2. The Policy DB Manager Service is started.

  3. The Policy DB Manager verifies the integrity of the NAP Policy Database, fixing any corrupted data as it is found.

- **Means of knowing requirement satisfied**:

  1. Every field in every policy within the NAP Policy Database can be accessed and the field value retrieved.

  2. Each field value in the NAP Policy Database is within the range specified in sections 7.3.2 Task Abstract Data Model and 8.3.2 Task Abstract Data Model.

- **Consequences of not satisfying requirement**: The task will not be able to read the values from the NAP Policy Database and will not be able to execute any NAP policies.

Requirement: The Authentication server is operational and has the ability to authenticate user credential.

- **Reason for requirement**: The Authentication server is used to authenticate user credentials against using its authentication algorithm. Part of this service may involve sending the credentials to an off-box authentication service, such as Active Directory. The Authentication server then returns the authentication result.

- **Means of satisfying the requirement**:

  1. The Authentication server is configured from the registry.

  2. The Authentication service is started.

  3. The Authentication service is connected to any off-box authentication services.

- **Means of knowing requirement satisfied**:

  1. The Authentication service is shown as running within the list of services.

  2. A user on the NAP client is able to authenticate and login via the Authentication services.

  3. A sniffer trace performed during a Health Certificate Enrollment event, shows HTTP packets traveling between the NAP Client and the NEP:

     1. The HTTP packets must contain an authentication blob in the Authorization field.

     2. The final server response is "200 OK".

  4. No errors are logged by the Authentication server.

- **Consequences of not satisfying requirement**: The task is unable to process authentication data on the NPS.

7.2.2 Task Relationships

7.2.2.1 Black-Box Relationship Diagrams

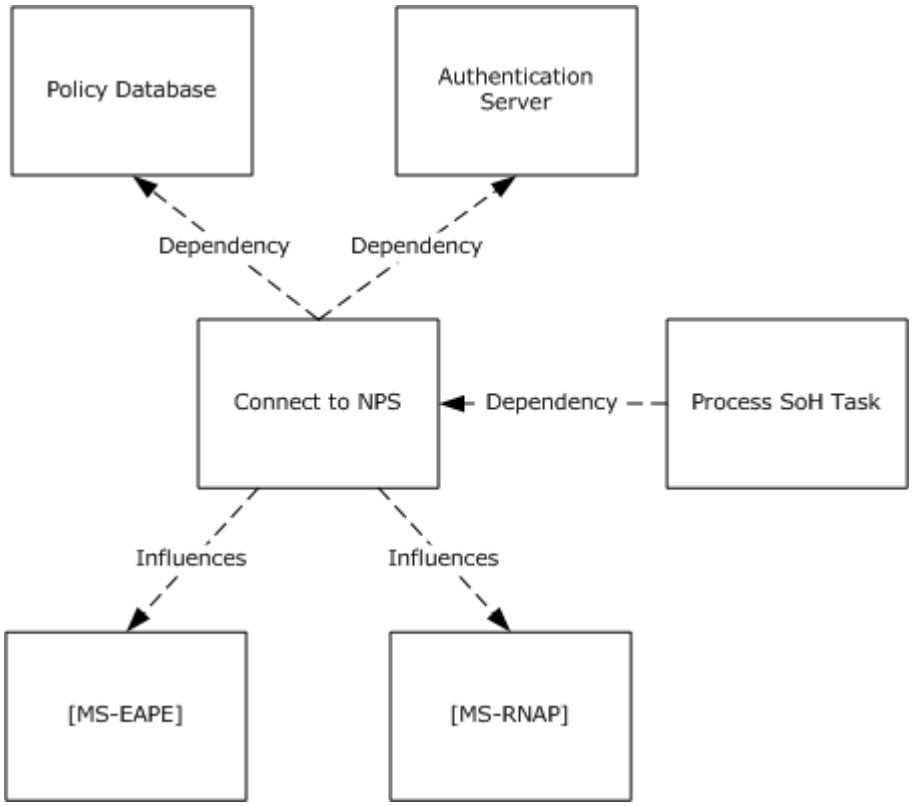

Figure 25: Connect to NPS Task black-box relationships

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

7.2.2.2 Task Dependencies

The Connect to NPS Task has a dependency on the Policy Database, which stores the Connection Policy that the task executes.

The Connect to NPS Task has a dependency on the Authentication Server, which processes the authentication data and renders an authentication decision for the user on the NAP Client.

The Process SoH Task has a dependency on the Connect to NPS Task. The Process SoH Task is triggered by the Connect to NPS Task and processes the data this task passes to it.

7.2.2.3 Task Influences

The Connect to NPS Task influences the MS-RNAP protocol, as the authentication decision determines whether the MS-RNAP server sends an Access-Reject or an Access-Accept message to the MS-RNAP client.

The Connect to NPS Task influences the MS-EAPE protocol, as the authentication decision determines what data the MS-EAPE server sends to the MS-EAPE client.

7.2.3 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.

7.2.4 Task Versioning and Capability Negotiation

The Connect to NPS 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.

7.3 Task Architecture

This section describes the structure of the Connect to NPS Task and the interrelationships among its parts.

7.3.1 Task Architectural Constraints

There can be more than one instance of the Connect to NPS 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.

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.

Authentication settings ADM:

Name	Type	Description
Authentication Method	DWORD	
Authentication Type	DWORD	0 = No Authentication

Name	Type	Description
		1 = Local Authentication 2 = Remote authentication
Remote Authentication Server	String	URL of the remote server to use.
Use Accounting	DWORD	0 = Don't use accounting. 1 = Use accounting.
Remote Accounting Server	String	URL of the remote server to use.
Called-Station-Id Rule	String	Find/Replace rule for RADIUS field.
Calling-Station-Id Rule	String	Find/Replace rule for RADIUS field.
User-Name Rule	String	Find/Replace rule for RADIUS field.
RADIUS Attribute	TLV list	RADIUS attribute to send with authentication request
AuthResponse	Blob	Binary blob containing response from the authentication server.
AuthResult	DWORD	0 = Authentication Rejected 1 = Authenticated 2 = Authentication Challenge

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].

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.

Name	Type	Description
SoHRequest	[MS-SOH] section 2.2.5	A SoH message created by the SoH client.
SoHReqLength	DWORD	Length, in bytes, of the SoH message.
AuthData	binary BLOB	Binary blob containing user credentials from the authentication client.
NotQuarantineCapable	DWORD	1 = The endpoint sent an SoH. 0 = The endpoint did not send an SoH.

Name	Type	Description
CorrelationId	24-byte GUID	A correlation ID containing a unique transaction identifier shared across SoH and SoHR messages.
ClientName	String	The Netbios name of the NAP Client generating the request.
MachineName	ANSI String	FQDN of the NAP Client.
ClientIPv4Address	DWORD	IPv4 address of the NAP Client.
ClientIpv6Address	16 Byte Array	IPv6 address of the NAP Client (if available).
DhcpServiceClass	String	The DHCP scope corresponding to the DhcpClientLease.
NASIdentifier	String	Name of the DHCP server
SecurityIdentity	[MS-DTYP] section 2.4.2	The security-identifier (SID) of the user requesting access.
CallingStationID	String	Unique ID of NAP Client (usually IP Address or MAC).
NetworkAccessServerType	DWORD	1 = Terminal Server Gateway 2 = Remote Access Service (RAS) server (VPN or dial-in) 3 = DHCP server 5 = Health Registration Authority (HRA) 6 = Host Credential Authorization Protocol (HCAP) server
TunnelType		Tunnel type (PPTP, L2TP, etc.)
NASPortType		Access Media type used by RNAP (ISDN, Ethernet, etc.)
NASIPv4Address	DWORD	IPv4 address of the RNAP Client.
NASIpv6Address	16 Byte Array	IPv6 address of the RNAP Client (if available).
HCAPLocationGroup	String	The location group name for the HCAP entity.
HCAPUserGroup	String	The group name to which an HCAP user belongs.
HCAPUserName	String	User identity information received over a HCAP interface [CM-HCAP].

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.

This task is always run asynchronously and never returns values to the caller.

7.3.5 White-Box Relationships

The following diagram shows the white-box relationships for the Connect to NPS Task.

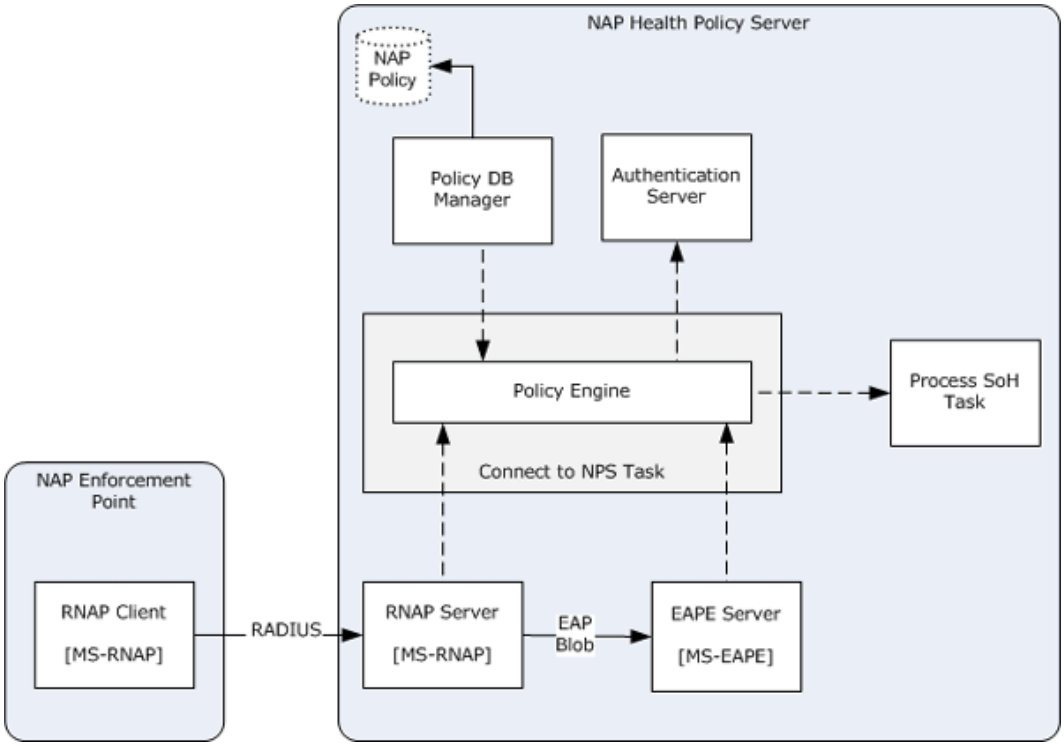

Figure 26: Connect to NPS Task white-box relationships

The Connect to NPS 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 Connect to NPS 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.

7.3.6 Task Events

7.3.6.1 Task Timers

The Connect to NPS 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

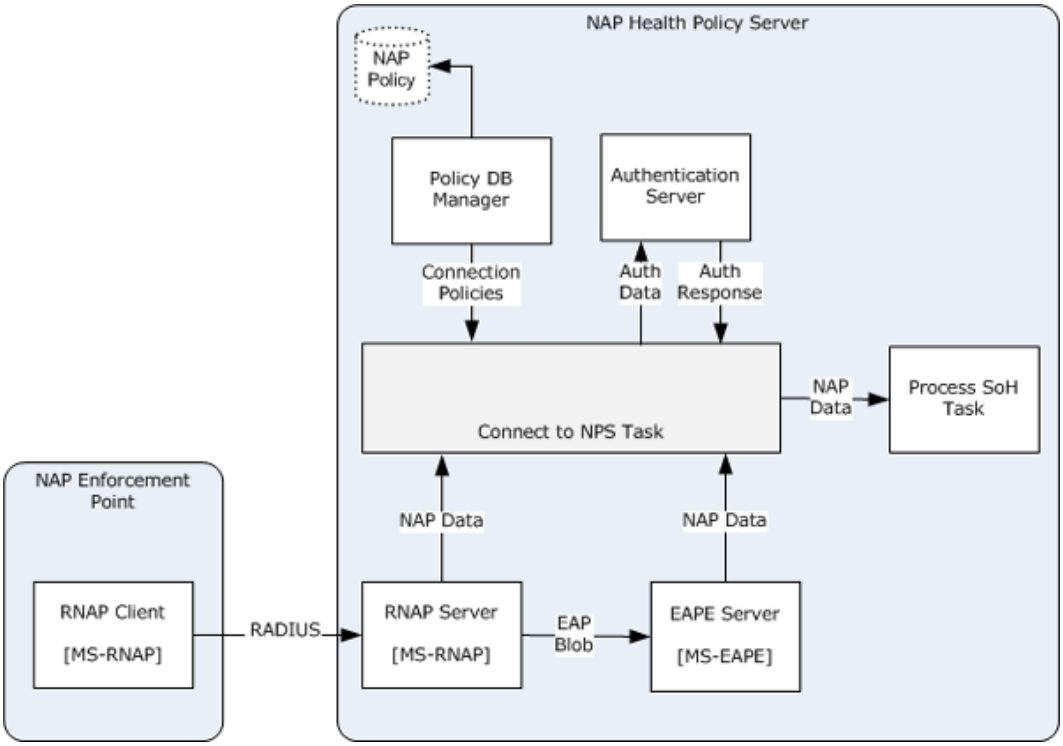

Figure 27: Connect to NPS Task architecture and communication

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.

7.3.8 Task Processing Rules

1. One of the following occurs to trigger the task:
   - The RNAP Server invokes the task with parameters specific to [MS-RNAP].
   - The CEAP Server invokes the task with parameters specific to [MS-CEAP].
2. The Policy Engine iterates through the connection policies using the following procedure:
   1. The Policy Engine requests the next connection policy object from the Policy DB Manager.
   2. The Policy DB Manager searches the Policy Database for the next connection policy in order.
   3. The Policy DB Manager passes the next connection policy object to the Policy Engine.
   4. The Policy Engine processes the connect policy as follows:
      1. The Policy Engine checks if the policy is enabled and is applicable to the NAS type.
      2. The Policy Engine iterates through the connection policy conditions and tries to match them against the input parameters.

3. If all the connection policy conditions evaluate to true, the Policy Engine iterates through the connection policy settings and sets the authentication settings ADM accordingly.

3. If all the connection policies have been processed without any connection policy having all its conditions evaluate to true, the authentication settings ADM is set to the configured default values.

4. Using the authentication settings, the Policy Engine passes the AuthData parameter to the corresponding Authentication Server, which in turn performs authentication.

5. The Authentication Server returns the results of the authentication to the Policy Engine.

6. The Policy Engine calls the Process SoH Task with all the abstract parameters it received, plus the AuthResult.

If an error is raised at any stage of the Connect to NPS Task, the Policy Engine logs an error and exits.

7.3.9 Task Failure Scenarios

7.3.9.1 NAP Health Policy Server and NEP Communication

These failures can be caused by:

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

If the NAP health policy server cannot communicate with the NEP, the server may not receive any encapsulated SoH messages from the NEP. 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:

- 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.

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 are defined in [MS-RNAP] and [MS-PEAP].

7.4.3.2 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].

7.4.4 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.

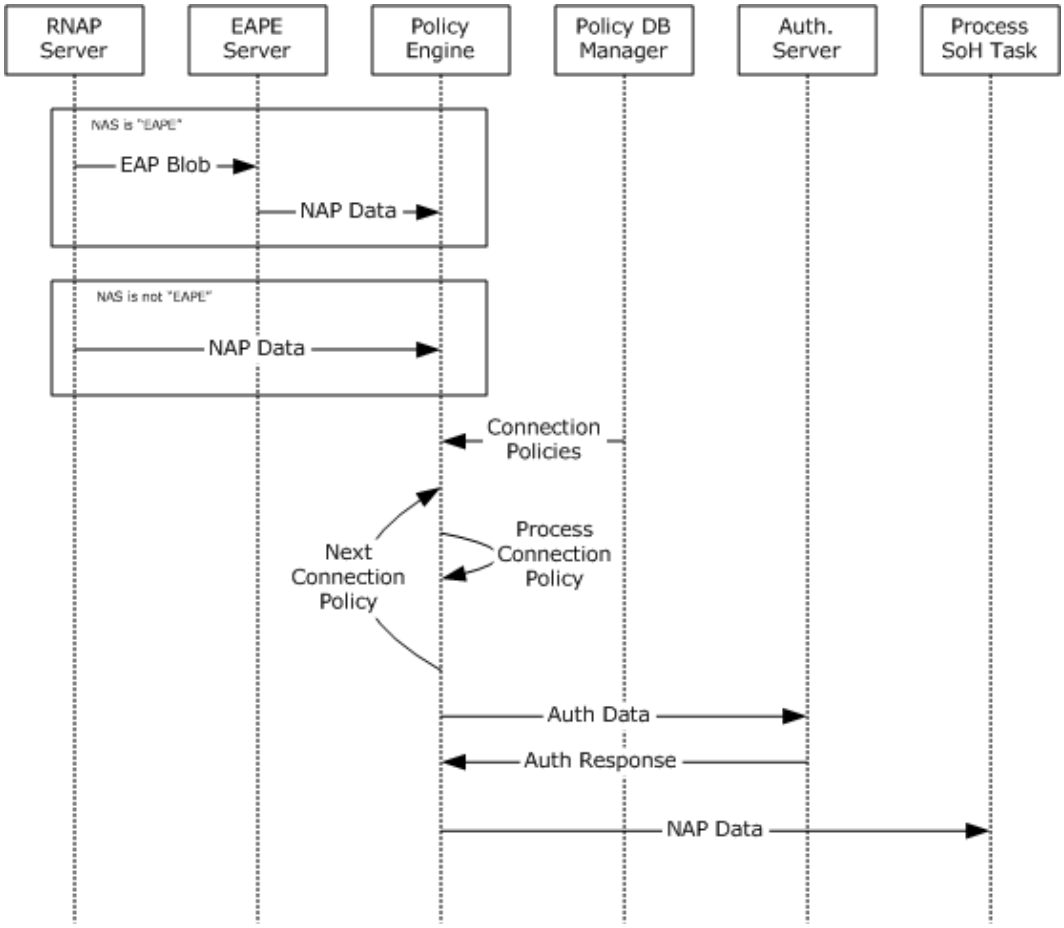

Figure 28: Sequence diagram for the main success scenario of the Connect to NPS Task

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.

7.4.5 Task Processing Rule Details

The following describes the operational details of the Connect to NPS Task:

1. One of the following occurs to trigger the task:
   - The RNAP Server invokes the task as described in section 3.2.5.1 Processing RADIUS Access-Request Messages in [MS-RNAP].
   - The CEAP Server invokes the task as described in section x.x.x Receiving an EAP Request in [MS-CEAP].
2. The Policy Engine iterates through the connection policies using the following procedure:
   1. The Policy Engine requests the next connection policy object from the Policy DB Manager.
   2. The Policy DB Manager searches the Policy Database for the next connection policy in order.
   3. The Policy DB Manager passes the next connection policy object to the Policy Engine.
   4. The Policy Engine processes the connect policy as follows:
      1. The connection policy's enabled flag is checked. If the connection policy is disabled, processing stops on the current policy and the Policy Engine jumps to the next iteration.
      2. The connection policy's NAS server type is compared to the NetworkAccessServerType parameter value. If they do not match, processing stops on the current policy and the Policy Engine jumps to the next connection policy iteration.
      3. The Policy Engine iterates through each of the connection policy conditions:
         - The condition name is looked up in the conditions table.
         - The configured value range is retrieved for that condition from the connection policy's conditions table.
         - The parameter value or system value corresponding to that condition, as specified in the conditions table, is tested against the configured value range.
         - If the compared value is not within the configured range, processing stops on the current policy and the Policy Engine jumps to the next connection policy iteration. Otherwise, the next connection policy condition is processed.
      4. If this point is reached, all the conditions in the current connection policy have evaluated to true. The Policy Engine next iterates through each of the connection policy settings:
         - The Policy Engine sets the type of authentication to use (Authentication Method).
         - The Policy Engine sets whether to perform authentication locally, remotely to a specific server or forego authentication.

- The Policy Engine sets whether to perform accounting and to what RNAP server.
- The Policy Engine sets any realm name modification rules.
- The Policy Engine sets any additional RADIUS attributes to use in authentication.

3. At this point, either 1) all the connection policies have been processed without any connection policy having all its conditions evaluate to true or 2) one connection policy had all its conditions evaluate to true.
4. If all the connection policies have been processed without any connection policy having all its conditions evaluate to true, the authentication settings ADM is set to the configured default values.
5. The Policy Engine performs authentication as follows:
    1. If the Authentication Type is set to 0, AuthResult is set to 1 (authenticated) and authentication processing is completed.
    2. If Called-Station-Id RADIUS attribute is present, a search and replace is performed using the Called-Station-Id Rule.
    3. If Calling-Station-Id RADIUS attribute is present, a search and replace is performed using the Calling-Station-Id Rule.
    4. If User-Name RADIUS attribute is present, a search and replace is performed using the Called-Station-Id Rule.
    5. If the Authentication Type is set to 1, the AuthData parameter is sent to the local Authentication Server that handles the Authentication Method for authentication.
    6. If the Authentication Type is set to 2, the AuthData parameter is sent to the remote Authentication Server located at Remote Authentication Server using the Authentication Method.
    7. The Policy Engine sets the AuthResult based on the Authentication Server's response (authenticated, rejected or challenged).
    8. The Policy Engine also receives authentication response data from the Authentication Server, which is used to set the AuthResponse ADM.
6. The Policy Engine calls the Process SoH Task with the abstract parameters from section 7.3.3, the AuthResult and the AuthResponse.

7.5 Task Security

The NEP and the NAP health policy server must maintain a trust relationship. For additional information about security considerations, see section 12, as well as the Security sections of the referenced protocol Technical Documents.

8 Process SoH Task

This section describes the task of executing the network policies on the NAP health policy server by the Policy Engine, which determine the client's network access.

8.1 Task Overview

8.1.1 Task Purpose

The purpose of this task is to pass the SoH message to the SoH Server for processing and to evaluate the NAP request data against the Health and Network policies.

8.1.2 Task Applicability

This task is used when the Policy Engine receives NAP request data from the Connect to NPS Task. 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 Process SoH Task are as follows:

Policy Engine: Responsible for executing NAP Health Policy Server policies (Connection, Health and Network) based on the NAP request data. In response to the execution of those policies, the Policy Engine generates NAP response data to be returned to the RNAP Server [MS-RNAP] or the EAPE Server [MS-EAPE], depending on the caller. The Policy Engine's interest in this particular task is to process the health and network policies.

Connect to NPS SoHR Task: The Connect to NPS Task executes the connection policies to determine how to authenticate the connection request, then forwards the NAP request data to this task. The interest of this actor in the Process SoH task is that the NAP request data passed in is processed.

Network Administrator: The Network Administrator wants to limit the computers connected to the corporate network to those verified as healthy. The Network Administrator also wants to control the network access of healthy computer. The Network Administrator creates and deploys policies to implement these restrictions. The Network Administrator's interests in this specific task are that only its deployed (enabled) health and network policies are executed.

SHAs: The purpose of these actors is to create SoHEntry fields, which are encapsulated within the SoH message. Those SoHEntry fields are evaluated by the corresponding SHVs on the NAP Health Policy Server, which in turn create SoHREntry fields for use by the SHAs on the NAP Client. The SHAs' interest in the task is that any SoH message created on the NAP Client, along with the SoHEntry fields, is passed on to the SoH Server for processing.

8.1.3.2 Supporting Actors and Task Interests Summary

Policy DB Manager: This actor maintains the persistent data store containing the policy configuration. The policy configuration contains the Connection Policy ADM elements documented in section 7.3.2 Task Abstract Data Model. The policy configuration also contains the Health Policy and Network Policy ADM elements documented in section 8.3.2 Task Abstract Data Model. The task contacts the Policy DB Manager whenever it needs policy configuration values from the data store.

SoH Server: The purpose of this actor is to utilize the processing rules defined in [MS-SOH] to evaluate the SoH packets for health and create SoHR response packets. This is accomplished by the actor then calling the abstract interface of each registered and enabled SHV in turn with the SoH packet. Each SHV evaluates the SoH packet and returns a SoHRReportEntry, which is appended together to create a SoHR packet. A SSoHR header is then prepended to the SoHR packet. The task employs this actor whenever a SoH packet needs evaluation.

8.1.3.3 Use Case Diagrams

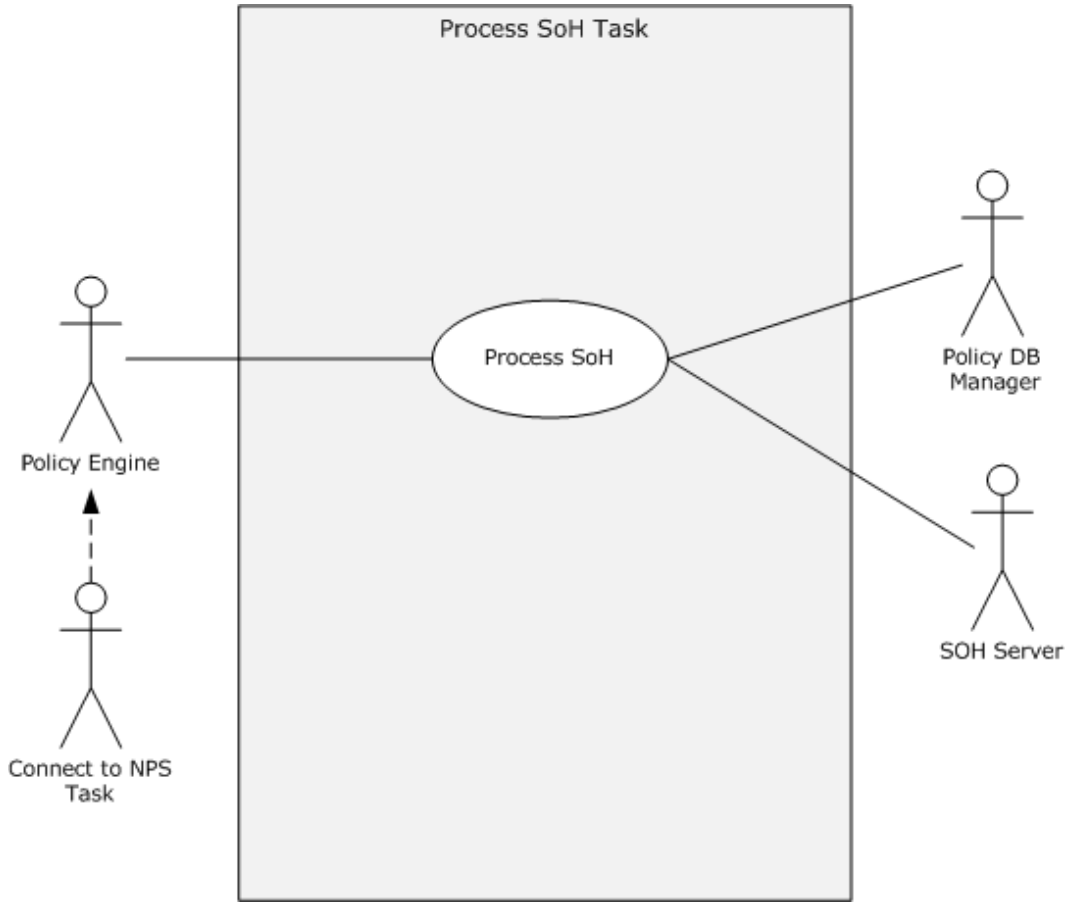

Figure 29: Process SoH Task use case diagram

8.1.3.4 Use Case: Process SoH -- Policy Engine

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

Goal: To process the NAP request data and pass the resulting NAP response data to the Send SoHR Task.

Context of Use: This use case is used when NAP request data is passed to the Policy Engine by the Connect to NPS Task. The direct actor is internal to the task.

Direct Actor: This role is performed by the Policy Engine.

Primary Actor: This role is performed by the Connect to NPS Task.

Supporting Actors:

Policy DB Manager: This actor maintains the persistent data store containing the policy configuration. The policy configuration contains the Connection Policy ADM elements documented in section 7.3.2 Task Abstract Data Model. The policy configuration also contains the Health Policy and Network Policy ADM elements documented in section 8.3.2 Task Abstract Data Model. The use case contacts the Policy DB Manager whenever it needs policy configuration values from the data store.

SoH Server: The purpose of this actor is to utilize the processing rules defined in [MS-SOH] to evaluate the SoH packets for health and create SoHR response packets. This is accomplished by the actor then calling the abstract interface of each registered and enabled SHV in turn with the SoH packet. Each SHV evaluates the SoH packet and returns a SoHRReportEntry, which is appended together to create a SoHR packet. A SSoHR header is then prepended to the SoHR packet. The use case employs this actor whenever a SoH packet needs evaluation.

Stakeholders and Interests: The stakeholders are defined as follows:

Network Administrator: The Network Administrator wants to limit the computers connected to the corporate network to those verified as healthy. The Network Administrator also wants to restrict which NAS are able to connect to the NAP Health Policy Server. The Network Administrator creates and deploys policies to implement these restrictions. The Network Administrator's interests in the use case are that only its deployed (enabled) health and network policies are executed.

SHAs: The purpose of these actors is to create SoHEntry fields, which are encapsulated within the SoH message. Those SoHEntry fields are evaluated by the corresponding SHVs on the NAP Health Policy Server, which in turn create SoHREntry fields for use by the SHAs on the NAP Client. The SHAs' interest in the use case is that any SoH message created on the NAP Client, along with the SoHEntry fields, is passed on to the SoH Server for processing.

Precondition:

- The NAP Health Policy Server components on the server are configured and working correctly.
- The integrity of the Policy DB is intact.
- The Network Administrator has deployed one or more Health policies.
- The Network Administrator has deployed one or more Network policies.

Minimal Guarantees:

- The use case will always process the task abstract parameters passed to it.
- The health and network policies will be processed.
- Only the health and network policies deployed by the Network Administrator will be executed.
- The SoH message, if present, will be sent to the SoH Server for processing, along with the encapsulated SoHEntry fields.

Success Guarantee: The network policies successfully create the NAP response data, which is passed to the Send SoHR Task.

Trigger: The trigger is the invoking of this task by the Connect to NPS Task.

Main Success Scenario:

1. The task is triggered when the Connect to NPS Task invoke the task's abstract interface.

2. The Policy Engine receives the task abstract parameters.

3. If NotQuarantineCapable is set to 1, the Policy Engine processes a SoH packet in the following manner:

   - The Policy Engine calls the EvaluateMachineHealth abstract interface ([MS-SOH] section 3.3.7.1), on the SoH Server, passing the SoHRequest as an input.
   - The SoH Server processes the SoH Packet using the process described in [MS-SOH] section 3.3.5.2.
   - The SoH Server returns a SoHR Packet to the Policy Engine using the SoHResponse output parameter of the EvaluateMachineHealth abstract interface.

4. The Policy Engine retrieves the network policies from the Policy DB Manager.

5. The Policy Engine iterates through the each Network Policy, doing the following:

   - Check if the Network Policy is enabled (deployed). If not, skip to next policy.
   - Using the NAP request data and SoHR as input, evaluate the Network Policy conditions.
   - If a Network Policy condition refers to a Health Policy, fetch that Health Policy from the Policy DB Manager and evaluate that Health Policy conditions.
   - If the Network Policy conditions (or the contained Health Policy conditions) do not evaluate to true, skip to next policy.
   - If the Network Policy conditions evaluate to true, note the Network Policy constraints and settings. Stop processing any more network policies.

6. The Policy Engine processes the Network Policy constraints. If the Network Policy constraints evaluate to true, the configured Network Access Permission is used. If the Network Policy constraints are not matched with the connection request, the network access decision is set to denied.

7. The Policy Engine processes the Network Policy settings to create the NAP response data, which will be used by the NAP Enforcement Point and NAP Client processes.

8. The Policy Engine invokes the Send SoHR Task using the NAP response data as the input parameters.

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 Policy Engine in an environment where the NAP request data has been forwarded to this task by the Connect to NPS Task. The environment should meet the following requirement to support this task.

Requirement: The SoH server is operational and has the ability to process [MS-SOH] messages.

- **Reason for requirement**: The SoH server is used to process the NAP Client health information contained in the SoH message and create the SoHR message that specifies the health of the NAP Client.
- **Means of satisfying the requirement**:

  1. The SoH server is configured with the registry settings.
  2. The SoH server service has been started.
  3. All required SHV plug-ins are installed and configured.
  4. All required SHV plug-ins have initialized and reported to the SoH server.
  5. All resources that each SHV depends on (such as AD server or requirements servers) are running and network accessible to the SHVs.

- **Means of knowing requirement satisfied**:

  1. The SoH server is shown as running within the list of services.
  2. A sniffer trace performed during a DHCP lease renewal event, shows DHCPN packets traveling between the NAP Client and the NEP containing a SoHR message.
  3. No errors are logged by the SoH server or the SHVs.

- **Consequences of not satisfying requirement**: The task is unable to process SoH messages and the NAP Client's health cannot be determined.

Requirement: The NAP Policy Database is uncorrupted and accessible.

- **Reason for requirement**: The Policy DB Manager will need to read the NAP Policy Database.
- **Means of satisfying the requirement**:

  1. The Policy DB Manager is configured with the path and security settings to access the NAP Policy Database.
  2. The Policy DB Manager Service is started.
  3. The Policy DB Manager verifies the integrity of the NAP Policy Database, fixing any corrupted data as it is found.

- **Means of knowing requirement satisfied**:

  1. Every field in every policy within the NAP Policy Database can be accessed and the field value retrieved.
  2. Each field value in the NAP Policy Database is within the range specified in sections 7.3.2 Task Abstract Data Model and 8.3.2 Task Abstract Data Model.

- **Consequences of not satisfying requirement**: The task will not be able to read the values from the NAP Policy Database and will not be able to execute any NAP policies.

8.2.2 Task Relationships

8.2.2.1 Black-Box Relationship Diagrams

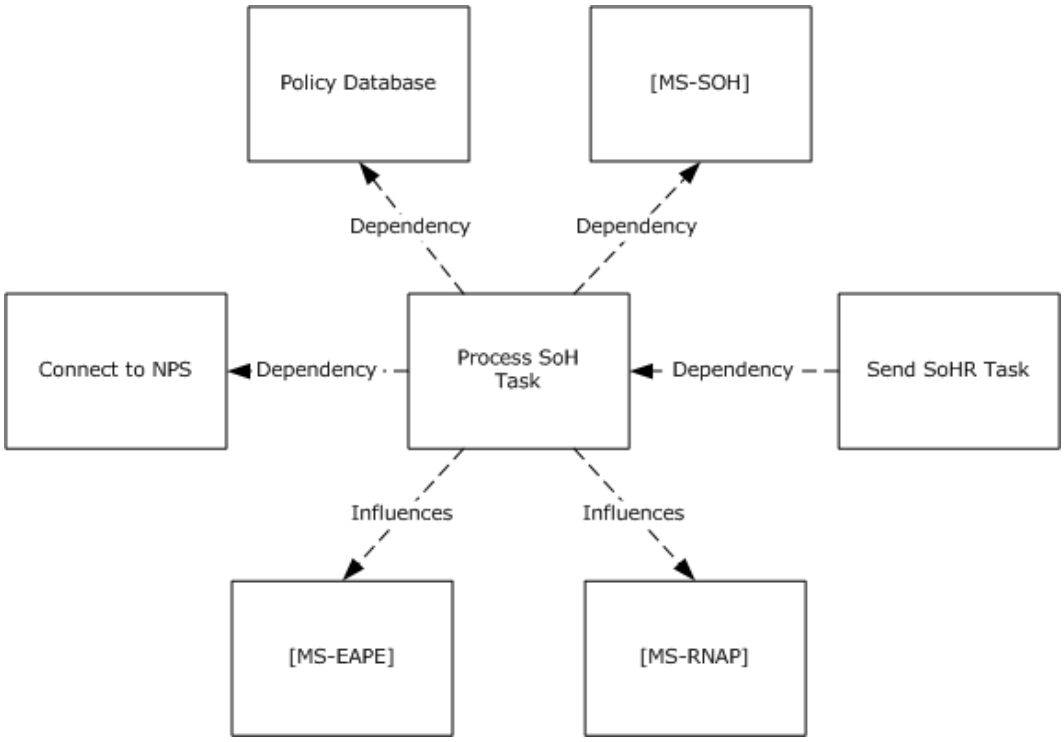

Figure 30: Process SoH Task black-box relationships

The NAP health policy server (PDP) is responsible to perform the health evaluation internally. This task has no relationship to external entities.

8.2.2.2 Task Dependencies

As shown in the previous figure, the Process SoH Task depends on the Connect to NPS Task, as it processes the NAP request data the Connect to NPS Task passes to it.

The Send SoHR Task has a dependency on the Process SoH Task. The Send SoHR is triggered by the Process SoH Task and processes the data this task passes to it.

The Process SoH Task has a dependency on the Policy Database, which stores the Health and Network Policies that the task executes.

The Process SoH Task depends on the MS-SOH server, and the MS-SOH processing rules, to process the SoH message and create a corresponding SoHR message.

8.2.2.3 Task Influences

The Process SoH Task influences the MS-RNAP protocol, as this task determines what values will be sent from the MS-RNAP server to the MS-RNAP client, including whether the MS-RNAP server sends an Access-Reject or an Access-Accept message.

The Process SoH Task influences the MS-EAPE protocol, as this task determines what data the MS-EAPE server sends to the MS-EAPE client.

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 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 executable 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.

8.2.4 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.

8.3 Task Architecture

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

8.3.1 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.

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.

Network settings ADM:

Name	Type	Description
Network Access	DWORD	O = Deny 1 = Grant

Name	Type	Description
SoHResponse	[MS-SOH] section 2.2.6	A SoHR message created by the SoH server.
SoHRespLength	DWORD	Length, in bytes, of the SoHR message.
IPFilter	[MS-RNAP] section 2.2.1.3	Set of IP filters to be set on NAP Client.
DhcpUserClass	String	DHCP user class to assign the NAP client to.
QuarantineState	Integer	0 = Full Access 1 = Limited Access (remediation) 2 = On probation until grace time
GraceTime	Integer	Deadline for NAP Client to conform to NAP policy (become healthy), expressed as number of seconds since 1/1/1970 UTC.
AfwZone	Integer	1 = A boundary policy (requires encryption) should be used by IPsec. 2 = An unprotected policy (does not require encryption) should be used by IPsec. 3 = A protected policy (requires encryption) should be used by IPsec.
AfwProtectionLevel	Integer	1 = HCEP certificate can be used only for signing. 2 = HCEP certificate can be used for signing and encrypting.
IPv4RemediationServers	[MS-RNAP] Section 2.2.1.16	A list of IPv4 servers to be used by the NAP Client for remediation.
IPv6RemediationServers	[MS-RNAP] Section 2.2.1.17	A list of IPv6 servers to be used by the NAP Client for remediation.
HcapExtendedState	DWORD	0 = No data 1 = Transition 2 = Infected 3 = Unknown
TsguRedirection	[MS-RNAP] Section 2.2.1.27	Redirection specification for Remote Desktop.

Policy Database:

Name	Type	Description
Connection Policies	Ordered list of type Connection Policy	See Connection Policy below.
Network Policies	Ordered list of type Network Policy	See Network Policy below.

Name	Type	Description
Health Policies	Ordered list of type Health Policy	See Health Policy below.

Connection Policy (see [MSFT-ConnReqPolicies]):

Name	Type	Comparator or Setting	Description
Policy Name	String		A name to describe the policy.
Policy State	Boolean		A flag to indicate whether the policy is enabled.
Network Connection Method	String	MS-Network-Access-Server-Type ([MS-RNAP], section 2.2.1.11)	Type of network access server that sends the connection request to NPS.
Settings:			*Policy Settings. These values are set by the policy if the conditions evaluate to TRUE and are returned to the NAS.*
Authentication Methods	Enum	Local ADM	The authentication method (MS-CHAP-V2, MS-CHAP, PAP, SPAP, PEAP, etc.) the client will authenticate with. Needs to be enabled or the Network policy value will be used.
Authentication Type	Enum	Local ADM	Type of authentication to perform: - Local - Remote - No Authentication
Authentication Server	String	Local ADM	IP address or DNS name of remote server.
Accounting	Boolean	Local ADM	Enable/disable use of RADIUS accounting.
Accounting server	String	Local ADM	IP address or DNS name of remote server.
Called-Station-Id Rules	List of rules	Local ADM	Rules that specify text to look for and text to replace the found text with.
Calling-Station-Id Rules	List of rules	Local ADM	Rules that specify text to look for and text to replace the found text with.
User-Name Rules	List of rules	Local ADM	Rules that specify text to look for and text to replace the found text with.
Standard RADIUS Attribute	List of TLVs	Local ADM	Add the given standard RADIUS attributes to the RADIUS authentication message.
Vendor Specific RADIUS Attribute	List of TLVs	Local ADM	Add the given standard RADIUS attributes to the RADIUS authentication message.

Network Policy (see [MSFT-NetworkPolicies]):

Name	Type	Comparator or Setting	Description
General:			
Policy Name	String	N/A	A name to describe the policy.
Policy State	Boolean	N/A	A flag to indicate whether the policy is enabled.
Network Connection Method	String	MS-Network-Access-Server-Type ([MS-RNAP], section 2.2.1.11)	Type of network access server that sends the connection request to NPS.
Access Permission	DWORD	TBD	Grany or Deny.
Constraints:			*Policy constraints. Each item MUST be separately enabled to take effect. If any constraint evaluates to FALSE, the Access Permission is set to DENY.*
Authentication Methods	Enum	TBD	The authentication method (MS-CHAP-V2, MS-CHAP, PAP, SPAP, PEAP, etc.) the client authenticated with.
Idle Timeout	DWORD	Current Time	Maximum time in minutes server can remain idle.
Session Timeout	DWORD	Current Time	Maximum time in minutes user can stay connected.
Called Station ID	Pattern-matching String	Called-Station-ID (RADIUS Attribute 30)	The Called Station ID condition specifies a character string that is the telephone number of the network access server (NAS).
Day and Time Restrictions	DateTime	Current Time	Day and Time Restrictions specify the days and times when connection attempts are and are not allowed. These restrictions are based on the time zone where the NPS server is located.
NAS Port Type	Enum	NAS-Port-Type (RADIUS Attribute 61)	The NAS Port Type condition specifies the type of media used by the access client. Such as analog phone lines. ISDN, tunnels or virtual private networks, IEEE802.11 wireless, and Ethernet switches.
Settings:			*Policy Settings. These values are set by the policy if the conditions evaluate to TRUE and are returned to the NAS.*
Standard RADIUS Attribute	List of TLVs	Standard RADIUS Attributes	Add the given standard RADIUS attributes to the RADIUS response message.
Vendor Specific RADIUS Attribute	List of TLVs	Vendor Specific RADIUS Attributes	Add the given standard RADIUS attributes to the RADIUS response message.
NAP Enforcement	DWORD	MS-Quarantine-	The target restrictive state of the

Name	Type	Comparator or Setting	Description
		State ([MS-RNAP], section 2.2.1.9)	endpoint.
NAP Enforcement – Time Limit	DWORD	MS-Quarantine-Grace-Time ([MS-RNAP], section 2.2.1.10)	The amount of time a host has to conform to network policy. Number of seconds since January 1, 1970
Extended State	Enum	MS-Extended-Quarantine-State ([MS-RNAP], section 2.2.1.21)	Used to specify additional information about a restricted access decision for HCAP.
Multilink	DWORD		How to handle multiple connections to the network
Bandwidth Allocation Protocol Capacity	DWORD	TBD	Percentage of capacity before reducing the multilink connection.
Bandwidth Allocation Protocol Period of Time	DWORD	TBD	Period of time (minutes) before reducing the multilink connection.
IPv4 Input Filters	IP Filter	MS-Quarantine-IPFilter ([MS-RNAP], section 2.2.1.3)	Traffic filters to a NAS for restricting access for a specific network access connection.
IPv4 Output Filters	IP Filter	MS-Quarantine-IPFilter ([MS-RNAP], section 2.2.1.3)	Traffic filters to a NAS for restricting access for a specific network access connection.
IPv6 Input Filters	IP Filter	MS-IPv6-Filter ([MS-RNAP], section 2.2.1.15)	Traffic filters to a NAS for restricting access for a specific network access connection.
IPv6 Output Filters	IP Filter	MS-IPv6-Filter ([MS-RNAP], section 2.2.1.15)	Traffic filters to a NAS for restricting access for a specific network access connection.
Encryption	Boolean List	TBD	List of encryption methods enabled for RRAS
IP Settings	Enum/IP Address	TBD	Client IP address assignment rules.

Health Policy (see [MSFT-HealthPolicies]):

Name	Type	Comparator	Description
Policy Name	String		A name to describe the policy.
SHV Evaluation	Enum	GetLastEvaluation ([MS-SOH], section 3.3.7.2)	The results of the SoH evaluation:

Name	Type	Comparator	Description
			1. Client passes all SHVs. 2. Client fails all SHVs. 3. Client passes 1+ SHVs. 4. Client fails 1+ SHVs. 5. Client reported transitional by 1+ SHVs. 6. Client reported infected by 1+ SHVs. 7. Client reported unknown by 1+ SHVs.
SHVs Used	Table with two columns: 1. SHV ID 2. Boolean – include this SHV?	GetLastEvaluation ([MS-SOH], section 3.3.7.2)	Table of SHV identifiers with flag to tell whether the SHV should be included in the policy evaluation.

Conditions:

Name	Type	Comparator	Description
Windows Groups	Group Object(s)	Active Directory return value	The Windows Group the connecting user or computer must belong to.
Machine Groups	Group Object(s)	MS-Machine-Name ([MS-RNAP], section 2.2.1.14)	The Machine Groups the connecting computer must belong to.
User Groups	Group Object(s)	Active Directory return value	The User Groups the connecting user must belong to.
Location Groups	String	HCAP-Location-Group-Name ([MS-RNAP], section 2.2.1.23)	The Host Credential Authorization Protocol (HCAP) location groups required to match this policy.
HCAP User Groups	Pattern-matching String	HCAP-User-Groups ([MS-RNAP], section 2.2.1.22)	The Host Credential Authorization Protocol (HCAP) location user groups required to match this policy.
User Name	Pattern-matching String	User-Name (RADIUS Attribute 1) or HCAP-User-Name ([MS-RNAP], section 2.2.1.24)	The user name that is used by the access client in the RADIUS message. This attribute typically contains a realm name and a user account name.

Name	Type	Comparator	Description
Access Client IPv4 Address	IPv4 Address	MS-User-IPv4-Address ([MS-RNAP], section 2.2.1.25)	The IPv4 address of the Access Client that is requesting access from the RADIUS client.
Access Client IPv6 Address	IPv6 Address	MS-User-IPv6-Address ([MS-RNAP], section 2.2.1.26)	The IPv6 address of the Access Client that is requesting access from the RADIUS client.
Framed Protocol	Enum	Framed-Protocol (RADIUS Attribute 7)	The Framed Protocol condition restricts the policy to only clients specifying a certain framing protocol for incoming packets. Such as PPP or SLIP.
Service Type	Enum	Service-Type (RADIUS Attribute 6)	The Service Type condition restricts the policy to only clients specifying a certain type of service. Such as Telnet or Point to Point Protocol connections.
Tunnel Type	Enum	Tunnel-Type ([MS-RNAP], section 2.2.2.1)	The Tunnel Type condition restricts the policy to only clients that create a specific type of tunnel, such as PPTP or L2TP.
Day and Time Restrictions	DateTime	Current Time	Day and Time Restrictions specify the days and times when connection attempts are and are not allowed. These restrictions are based on the time zone where the NPS server is located.
Identity Type	Enum	MS-Identity-Type ([MS-RNAP], section 2.2.1.6)	The Identity Type condition restricts the policy to only clients that can be identified through the specified mechanism. Such as NAP statement of health (SoH).
MS-Service-Class	String	MS-Service-Class ([MS-RNAP], section 2.2.1.7)	MS-Service-Class condition specifies that the connecting computer must have an IP address lease from a DHCP scope that matches the selected profile name.
Health Policies	Health Policy	See Health Policy	The Health Policies condition restricts the policy to only clients that meet the health criteria specified in the health policy.
NAP-Capable Computers	Boolean	Not-Quarantine-Capable ([MS-RNAP], section 2.2.1.18)	The NAP-Capable Computers condition specifies that connecting computers either are or are not capable of participating in NAP. This capability is determined by whether the client computer sends a statement of health to NPS.
Operating System	OS Version	MS-Machine-Inventory ([MS-SOH], section 2.2.4.1)	The Operating System condition specifies the operating system, role, and architecture required for client computer configuration to match this policy.
Policy Expiration	DateTime	Current Time	The Policy Expiration condition specifies when the network policy expires and is no longer evaluated by NPS. This condition is used with

Name	Type	Comparator	Description
			the NPS Enforcement setting that allows clients full network access for a limited time. If used for this policy. Configure another NAP network policy for after expiration time.
Authentication Type	enum	TBD	The Authentication Type condition specifies the authentication methods required to match this policy.
Allowed EAP Types	Integer	TBD	The Allowed EAP Types condition specifies the EAP types required for client computer authentication method configuration to match this policy. Use of this condition requires that EAP is also configured in connection request policy.
Calling Station ID	Pattern-matching String	Calling-Station-ID (RADIUS Attribute 31)	The Calling Station ID condition specifies the network access server telephone number dialed by the access client.
Client Friendly Name	Pattern-matching String	TBD	The Client Friendly Name condition specifies the name of the RADIUS client that forwarded the connection request to NPS.
Client IPv4 Address	Pattern-matching IPv4 Address	System Call	The Client IPv4 Address condition specifies the IPv4 address of the RADIUS client that forwarded the connection request to NPS.
Client IPv6 Address	Pattern-matching IPv6 Address	System Call	The Client IPv6 Address condition specifies the IPv6 address of the RADIUS client that forwarded the connection request to NPS
Client Vendor	Enum	TBD	The Client Vendor condition specifies the manufacturer of the RADIUS client that sends connection requests to NPS.
MS-RAS Vendor	SMI Code	MS-RAS-Vendor (RFC2548 section 2.7.1)	The MS-RAS Vendor condition specifies the vendor identification number of the network access server (NAS) that is requesting authentication.
Called Station ID	Pattern-matching String	Called-Station-ID (RADIUS Attribute 30)	The Called Station ID condition specifies a character string that is the telephone number of the network access server (NAS).
NAS Identifier	String	TBD	The NAS Identifier condition specifies a character string that is the name of the network access server (NAS).
NAS IPv4 Address	Pattern-matching IPv4 Address	NAS-IP-Address (RADIUS Attribute 4)	The NAS IPv4 Address condition specifies a character string that is the IPv4 address of the NAS.
NAS IPv6 Address	Pattern-matching	NAS-IP-Address (RADIUS Attribute	The NAS IPv6 Address condition specifies a character string that is the IPv6 address of

Name	Type	Comparator	Description
	IPv4 Address	95)	the NAS.
NAS Port Type	Enum	NAS-Port-Type (RADIUS Attribute 61)	The NAS Port Type condition specifies the type of media used by the access client. Such as analog phone lines. ISDN, tunnels or virtual private networks, IEEE802.11 wireless, and Ethernet switches.

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.

Name	Type	Description
AuthResponse	Blob	Binary blob containing response from the authentication server.
AuthResult	DWORD	0 = Authentication Rejected 1 = Authenticated 2 = Authentication Challenge
SoHRequest	[MS-SOH] section 2.2.5	A SoH message created by the SoH client.
SoHReqLength	DWORD	Length, in bytes, of the SoH message.
NotQuarantineCapable	DWORD	1 = The endpoint sent a SoH. 0 = The endpoint did not send a SoH.
CorrelationId	24-byte GUID	A correlation ID containing a unique transaction identifier shared across SoH and SoHR messages.
ClientName	String	The Netbios name of the NAP Client generating the request.
MachineName	ANSI String	FQDN of the NAP Client.
ClientIPv4Address	DWORD	IPv4 address of the NAP Client.
ClientIpv6Address	16 Byte Array	IPv6 address of the NAP Client (if available).
DhcpServiceClass	String	The DHCP scope corresponding to the DhcpClientLease.
NASIdentifier	String	Name of the DHCP server
SecurityIdentity	[MS-DTYP] section 2.4.2	The security-identifier (SID) of the user requesting access.

Name	Type	Description
CallingStationID	String	Unique ID of NAP Client (usually IP Address or MAC).
NetworkAccessServerType	DWORD	1 = Terminal Server Gateway 2 = Remote Access Service (RAS) server (VPN or dial-in) 3 = DHCP server 5 = Health Registration Authority (HRA) 6 = Host Credential Authorization Protocol (HCAP) server
TunnelType		Tunnel type (PPTP, L2TP, etc.)
NASPortType		Access Media type used by RNAP (ISDN, Ethernet, etc.)
NASIPv4Address	DWORD	IPv4 address of the RNAP Client.
NASIpv6Address	16 Byte Array	IPv6 address of the RNAP Client (if available).
HCAPLocationGroup	String	The location group name for the HCAP entity.
HCAPUserGroup	String	The group name to which an HCAP user belongs.

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.

This task is always run asynchronously and never returns values to the caller.

8.3.5 White-Box Relationships

The following diagram shows the white-box relationships between the Process SoH Task and other tasks.

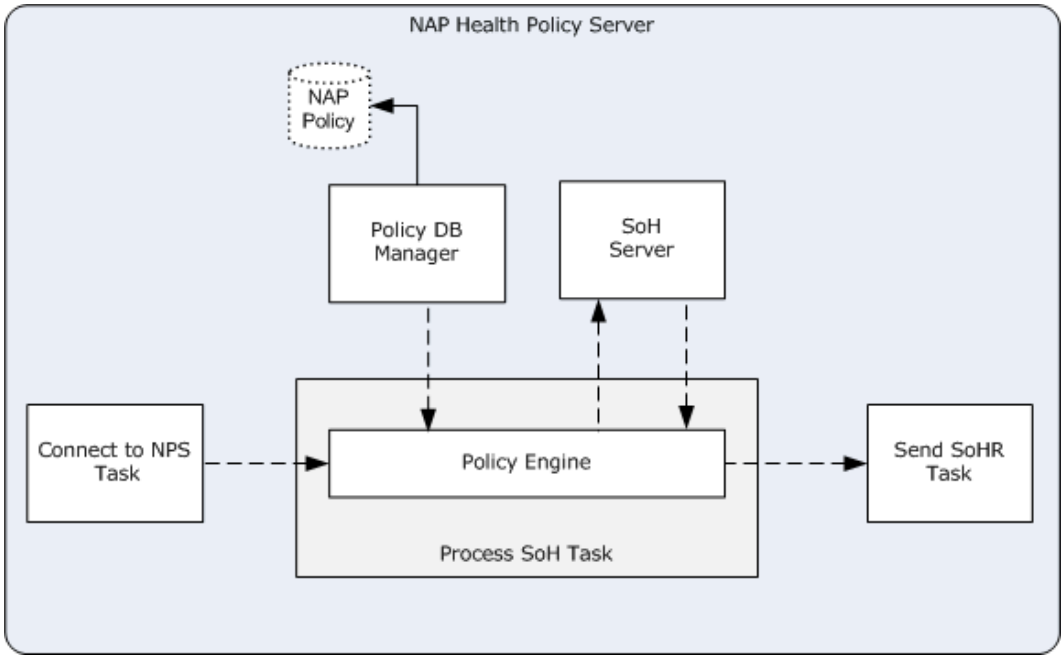

Figure 31: Process SoH Task white-box relationships

The Process SoH Task uses three major NAP health policy server components: SoH Server, Policy Engine, and Policy Configuration Manager.

From the 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 [MS-SOH] 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.

8.3.6 Task Events

8.3.6.1 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 Microsoft Windows® registry. Further details can be found in section 8.4.3.1.

8.3.6.2 Task Non-Timer Events

This task does not use or respond to any additional non-timer events.

8.3.7 Task Architecture and Communication

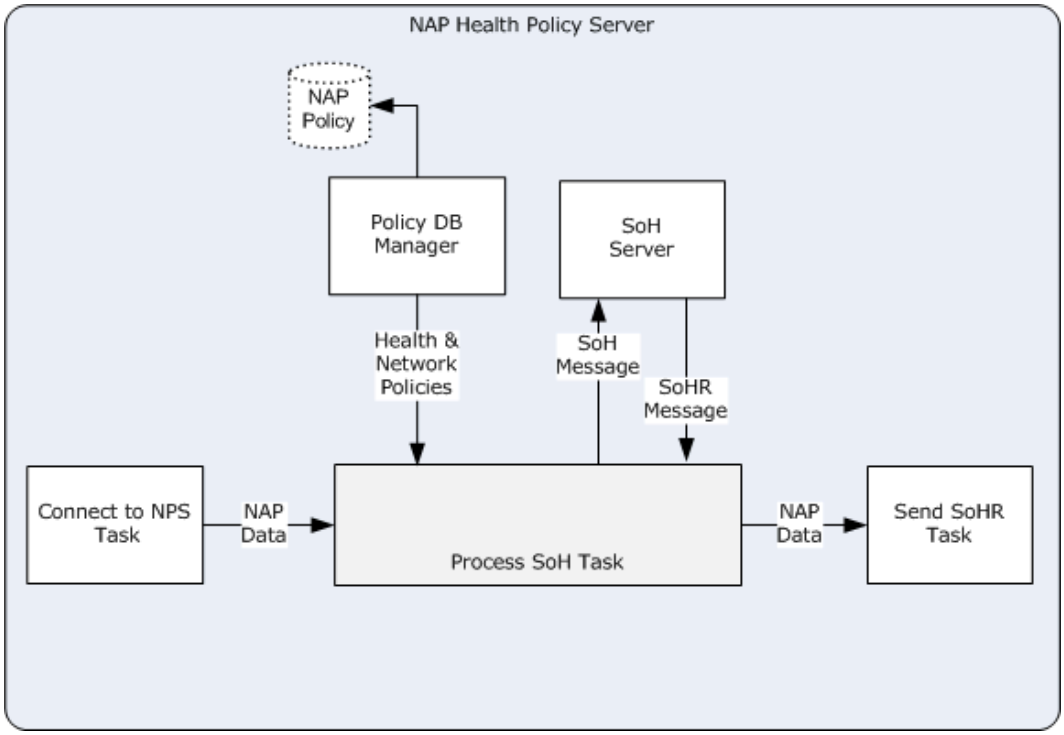

Figure 32: Process SoH Task architecture and communication overview

8.3.8 Task Processing Rules

1. The Connect to NPS Task invokes this task.
2. If NotQuarantineCapable is set to 1, the Policy Engine calls the EvaluateMachineHealth abstract interface in [MS-SOH], passing the SoHRequest as an input and receiving the SoHResponse as an output.
3. The Policy Engine iterates through the network policies using the following procedure:
    1. The Policy Engine requests the next network policy object from the Policy DB Manager.
    2. The Policy DB Manager searches the Policy Database for the next connection policy in order.
    3. The Policy DB Manager passes the next network policy object to the Policy Engine.
    4. The Policy Engine processes the network policy as follows:
        1. The Policy Engine checks if the policy is enabled and is applicable to the NAS type.
        2. The Policy Engine iterates through the network policy conditions and tries to match them against the input parameters.
        3. If condition refers to a health policy:
            - The Policy Engine requests the health policy object from the Policy DB Manager.

- The Policy DB Manager fetches the health policy from the Policy Database and pass it to the Policy Engine.
- The health policy conditions are matched against the SoHResponse from the SoH Server.

4. If all the conditions in the current network policy have evaluated to true. The Policy Engine next iterates through each of the network policy constraints. If any network constraint does not evaluate to true, Network Access is set to Deny.
5. If all of the constraints have evaluated to true, the Policy Engine sets the Network Access to the policy's configured value (Grant or Deny).
6. If all of the constraints have evaluated to true, the Policy Engine iterates through the network policy settings and sets the network settings ADM accordingly.

4. If all the network policies have been processed without any network policy having all its conditions evaluate to true, the network settings ADM is set to the configured default values.
5. The Policy Engine calls the Send SoHR Task with the network settings ADM elements.

If an error is raised at any stage of the Process SoH Task, the Policy Engine logs an error and exits.

8.3.9 Task Failure Scenarios

8.3.9.1 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.

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 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.

8.4.2 Task Initialization of External Entities

None.

8.4.3 Task Event Details

8.4.3.1 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 8.3.2.

8.4.3.2 Task Non-Timer Event Details

None.

8.4.4 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]).

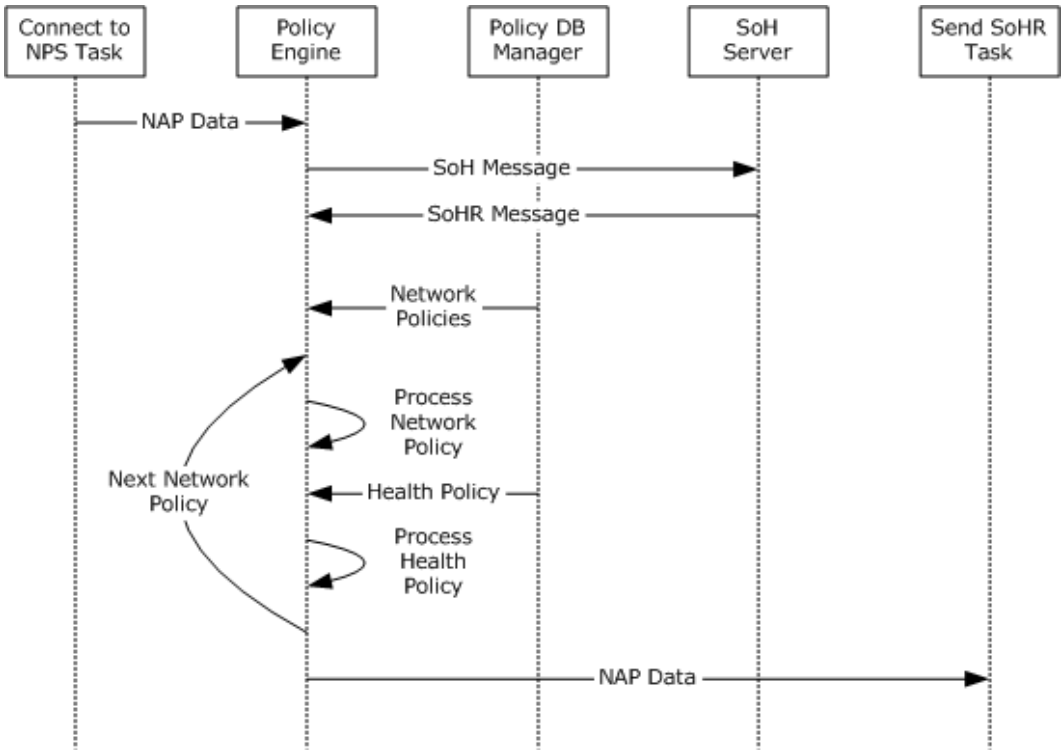

Figure 33: Sequence diagram for the main success scenario of the Process SoH Task

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.

8.4.5 Task Processing Rule Details

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

1. The Connect to NPS Task invokes this task in the last processing rule in section 7.4.5.
2. If NotQuarantineCapable is set to 1, the Policy Engine does the following:
    1. The Policy Engine calls the EvaluateMachineHealth abstract interface ([MS-SOH] section 3.3.7.1), on the SoH Server, passing the SoHRequest as an input.
    2. The SoH Server returns a SoHR Packet to the Policy Engine using the SoHResponse output parameter of the EvaluateMachineHealth abstract interface.
3. The Policy Engine iterates through the network policies using the following procedure:
    1. The Policy Engine requests the next network policy object from the Policy DB Manager.
    2. The Policy DB Manager searches the Policy Database for the next network policy in order.
    3. The Policy DB Manager passes the next network policy object to the Policy Engine.
    4. The Policy Engine processes the network policy as follows:
        1. The network policy's enabled flag is checked. If the network policy is disabled, processing stops on the current policy and the Policy Engine jumps to the next network policy iteration.
        2. The network policy's NAS server type is compared to the NetworkAccessServerType parameter value. If they do not match, processing stops on the current policy and the Policy Engine jumps to the next network policy iteration.
        3. The Policy Engine iterates through each of the network policy conditions:
            - If the condition name is 'Health Policies', the following is performed:
                1. The health policy name is retrieved from the network policy's conditions table.
                2. The Policy Engine requests the health policy object corresponding to the health policy name from the Policy DB Manager.
                3. The Policy Engine iterates through the SHVs Used table and calls GetLastEvaluation (see section 3.3.7.2 GetLastEvaluation in [MS-SOH]) with the SHV ID value to get the evaluationResult, if the boolean value is set to true (SHV evaluation enabled).
                4. While treating through the SHVs Used table, the Policy engine maintains a count of SHVs enabled, the SHVs reporting failure and the SHVs reporting no failure.
                5. Using the SHV Evaluation value from the Health policy, the Policy engine evaluates the SHVs reporting failure, the SHVs reporting no failure and the count of SHVs enabled to determine if the health policy evaluates to true or false.
                6. If the health policy evaluates to false, processing stops on the current policy and the Policy Engine jumps to the next network policy iteration. Otherwise, the next network policy condition is processed.
            - If the condition name is NOT 'Health Policies', the following is performed:

1. The condition name is looked up in the conditions table.
2. The configured value range is retrieved for that condition from the network policy's conditions table.
3. The parameter value or system value corresponding to that condition, as specified in the conditions table, is tested against the configured value range.
4. If the compared value is not within the configured range, processing stops on the current policy and the Policy Engine jumps to the next network policy iteration. Otherwise, the next network policy condition is processed.

4. If this point is reached, all the conditions in the current network policy have evaluated to true. The Policy Engine next iterates through each of the network policy constraints:
   - The Policy Engine checks if the Authentication Method used matches the Authentication Method in the network policy. If not, Network Access is set to 0 (Deny).
   - If enabled, the Policy Engine checks if maximum idle timeout has been reached. If so, Network Access is set to 0 (Deny).
   - If enabled, the Policy Engine checks if maximum session timeout has been reached. If so, Network Access is set to 0 (Deny).
   - If enabled, the Policy Engine checks if the Called-Station-Id value in the network policy constraints equals the CalledStationId parameter. If not, Network Access is set to 0 (Deny).
   - If enabled, the Policy Engine checks if current time is within the time range specified in the network policy constraints. If not, Network Access is set to 0 (Deny).
   - The Policy Engine checks if any configured media type constraints in the network policy match the NASPortType. If configured and the NASPortType doesn't match the constraint, Network Access is set to 0 (Deny).
5. If none of the constraints have resulted in the Network Access being set to 0 (Deny), the Policy Engine sets the Network Access to the value configured in the network policy object (Grant or Deny).
6. If none of the constraints have resulted in the Network Access being set to 0 (Deny), the Policy Engine next iterates through each of the network policy settings:
   - The Policy Engine sets any additional RADIUS attributes to use in the RNAP response.
   - The Policy Engine sets the QuarantineState (full, limited or probation) based on the NAP enforcement settings.
   - The Policy Engine sets GraceTime, if the QuarantineState is set to probation.
   - The Policy Engine sets the IPv4RemediationServers and IPv6RemediationServers values.
   - The Policy Engine sets HcapExtendedState per the extended state setting.
   - The Policy Engine creates a set of input and output IPv4 network interface filters for use by the NAS server. The Policy Engine adds a set of input and output IPv6 filters.

4. At this point, either 1) all the network policies have been processed without any network policy having all its conditions evaluate to true or 2) one network policy had all its conditions evaluate to true.

5. If all the network policies have been processed without any network policy having all its conditions evaluate to true, the network settings ADM is set to the configured default values.

6. The Policy Engine calls the Send SoHR Task with AuthResult, Network Access, SoHResponse, SoHRespLength, CorrelationId, NetworkAccessServerType, IPFilter, DhcpUserClass, NotQuarantineCapable, QuarantineState, GraceTime, AfwZone, AfwProtectionLevel, IPv4RemediationServers, IPv6RemediationServers, TsguRedirection and the AuthResponse.

8.5 Task Security

There are no task-specific security considerations. For additional information about security considerations, see section 12, as well as the Security sections of the referenced protocol Technical Documents.

9 Send SoHR Task

This section describes the task of sending NAP data, including the SoHR messages, received from the Process SoH Task, to the RNAP Server [MS-RNAP] or the EAPE Server [MS-EAPE].

9.1 Task Overview

9.1.1 Task Purpose

The purpose of this task is to send the results of the health evaluation back to the NEP. It does this by invoking abstract interfaces on either the RNAP Server [MS-RNAP] or the EAPE Server [MS-EAPE], passing in NAP response data it received from the Process SoH Task. NAP response data passed to the EAPE Server [MS-EAPE] is later sent to the RNAP Server [MS-RNAP] for transport to the RNAP client.

9.1.2 Task Applicability

This task is used when the Policy Engine receives NAP response data from the Process SoH Task. This task is not applicable if the NAP System is not deployed.

9.1.3 Task Use Cases

9.1.3.1 Stakeholders and Interests Summary

Policy Engine: Responsible for executing NAP Health Policy Server policies (Connection, Health and Network) based on the NAP data. In response to the execution of those policies, the Policy Engine generates NAP response data to be returned to the RNAP Server [MS-RNAP] or the EAPE Server [MS-EAPE], depending on the caller. The Policy Engine's interest in this particular task is that the task parameters are passed on to a [MS-RNAP] or [MS-EAPE] abstract interface for valid NAS server types.

Process SoH Task: The purpose of the Process SoH task is to pass the SoH message to the SoH Server for processing and to evaluate the NAP request data against the Health and Network policies. It then generates NAP response data based on those policies. The interest of this actor in the Send SoHR task is that the NAP response data passed in is processed.

Proxy SoHR Task: The purpose of this stakeholder is to proxy the NAP specific data from the RNAP client to the DHCPN, HCEP, TSGU, or EAPE servers. As such, the primary interest of this stakeholder is to ensure that the Send SoHR Task only sends protocol messages that contain NAP data.

9.1.3.2 Supporting Actors and Task Interests Summary

RNAP Server: This protocol server is used to process RNAP Protocol [MS-RNAP] messages received from a RNAP client on the NAP Enforcement Point. The task employs this actor whenever NAP response data needs to be sent back to the RNAP client.

EAPE Server: This protocol server is used to process EAPE Protocol [MS-EAPE] messages received from the RNAP Server. The task employs this actor whenever NAP response data needs to be sent back to the EAPE proxy.

9.1.3.3 Use Case Diagrams

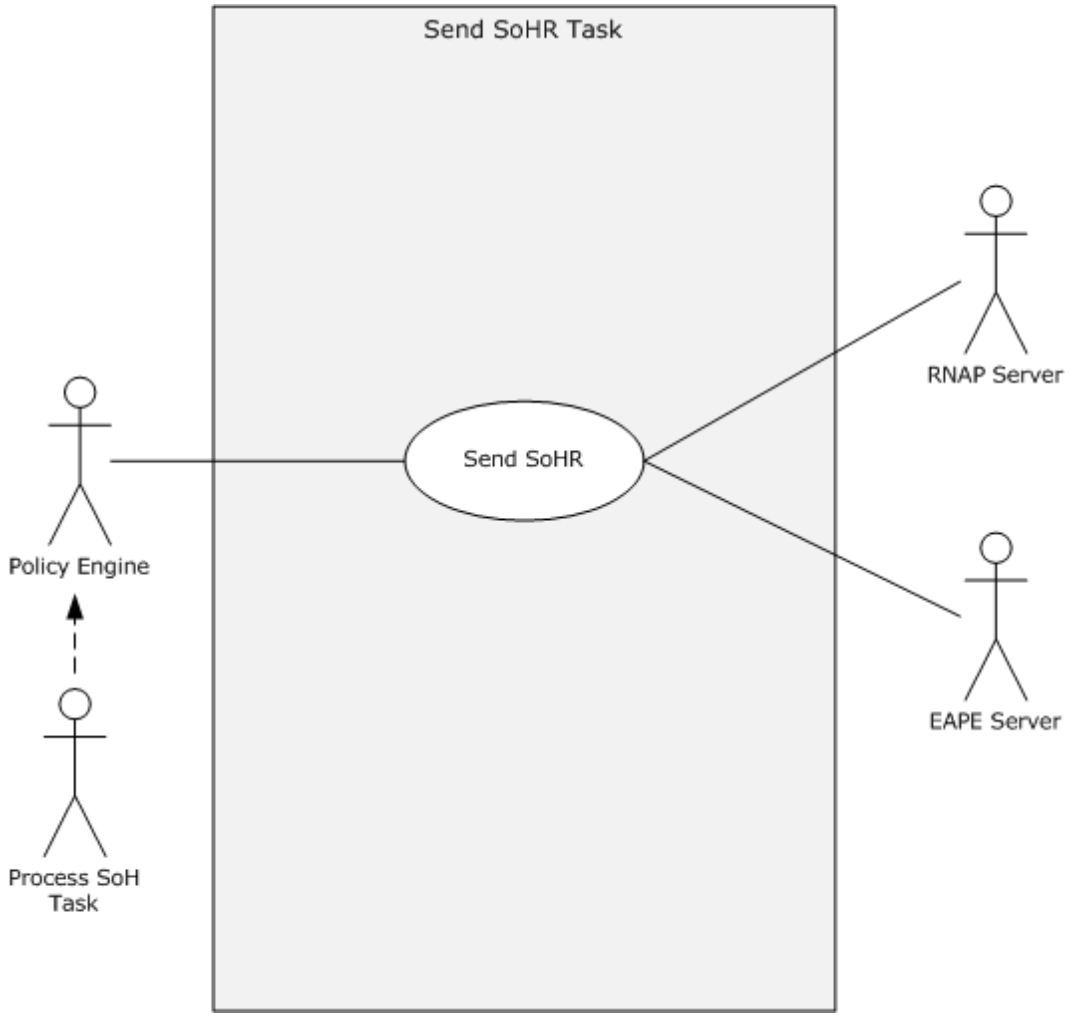

Figure 34: Send SoHR Task use case diagram

9.1.3.4 Use Case: Send SoHR – Policy Engine

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

Goal: To create an SoHR message [MS-SOH] containing health evaluation results for the NAP Client and send it to the NEP.

Context of Use: This use case is used when NAP response data is passed by the Process SoH Task to the Policy Engine. The direct actor is internal to the task.

Direct Actor: This role is performed by the Policy Engine.

Primary Actor: This role is performed by the Process SoH Task.

Supporting Actors:

RNAP Server: This protocol server is used to process RNAP Protocol [MS-RNAP] messages received from a RNAP client on the NAP Enforcement Point. The use case employs this actor whenever NAP response data needs to be sent back to the RNAP client.

EAPE Server: This protocol server is used to process EAPE Protocol [MS-EAPE] messages received from the RNAP Server. The use case employs this actor whenever NAP response data needs to be sent back to the EAPE proxy.

Stakeholders and Interests: The stakeholders are defined as follows:

Proxy SoHR Task: The purpose of this stakeholder is to proxy the NAP specific data from the RNAP client to the DHCPN, HCEP, TSGU, or EAPE servers. As such, the primary interest of this stakeholder is to ensure that the Send SoHR Task only sends protocol messages that contain NAP data.

Preconditions: The NAP Health Policy Server components on the server are configured and working correctly.

Minimal Guarantees:

- The use case will always process the task abstract parameters passed to it.
- A [MS-RNAP] or [MS-EAPE] abstract interface will be invoked, with properly mapped parameters, for valid NAS server types.
- No protocol messages are sent by the task without NAP response data.

Success Guarantee: The NAP response data is successfully sent to either the RNAP Server or the EAPE Server.

Trigger: The trigger is the invoking of this task by the Connect to NPS Task.

Main Success Scenario:

1. The task is triggered by RNAP Client invoking the task's abstract interface.
2. The Policy Engine receives the task abstract parameters.
3. If the NetworkAccessServerType equals 2:
   - The Policy Engine invokes an [MS-EAPE] abstract interface on the EAPE Server.
   - The Policy Engine passes values to the invoked [MS-EAPE] abstract interface by mapping the task abstract parameters to the parameter listing of the [MS-EAPE] abstract interface.
4. If the NetworkAccessServerType is in the set of [1, 3, 5, 6]:
   - The Policy Engine invokes an [MS-RNAP] abstract interface on RNAP Server.
   - The NAP Proxy passes values to the invoked [MS-RNAP] abstract interface by mapping the task abstract parameters to the parameter listing of the [MS-RNAP] abstract interface.

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 Policy Engine in an environment where the NAP response data has been forwarded to this task by the Process SoH Task. The environment should meet the following requirement to support this task.

Requirement: The RNAP server is running and has the ability to send [MS-RNAP] packets to the RNAP client on the Network Enforcement Point (NEP).

- **Reason for requirement**: The RNAP server is used to send NAP response data (including the SoHR) encapsulated within [MS-RNAP] packets to the NEP.
- **Means of satisfying the requirement**:

  1. The RNAP server is configured with the RADIUS settings from the registry.
  2. The RNAP server has network access to the RNAP client:
     1. The network interface of the NPS is configured to operate on the local subnet.
     2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NEP is connected.
     3. All network devices between the local subnet and the NEP are configured to allow packet flow between the two entities.
     4. The routing tables in the NPS are configured to enable correct packet routing between the NPS and the NEP.
  3. The RNAP server service has been started.

- **Means of knowing requirement satisfied**:

  1. The NPS can successfully ping the NEP over the network.
  2. The RNAP server is shown as running within the list of services.
  3. A sniffer trace performed during a RADIUS access-request event, shows RADIUS packets traveling between the NPS and the NEP. These must be "access-accept" RADIUS packets containing [MS-RNAP] fields.
  4. No errors are logged by the RNAP server.

- **Consequences of not satisfying requirement**: The task is unable to send NAP response data via the [MS-RNAP] protocol.

Requirement: The EAPE server is running and has the ability to send NAP response data to the RNAP server on the NAP Health Policy Server (NPS).

- **Reason for requirement**: The EAPE client is used to encapsulate the SoHR message into an EAP blob and pass the EAP blob (along with NAP response data) to the RNAP server.
- **Means of satisfying the requirement**:

  1. The EAPE server is configured with settings from the registry.
  2. The RNAP server service has been started.
  3. The EAPE server service has been started.

- **Means of knowing requirement satisfied**:

  1. The RNAP server is shown as running within the list of services.

  2. The EAPE server is shown as running within the list of services.

  3. A sniffer trace performed during a VPN connection event, shows EAPE packets traveling between the NAP Client, the NEP and the NPS (over RADIUS). These EAPE packets must contain [MS-EAPE] fields.

  4. No errors are logged by the EAPE server.

- **Consequences of not satisfying requirement**: The task is unable to send NAP response data via the [MS-EAPE] protocol.

9.2.2 Task Relationships

9.2.2.1 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.

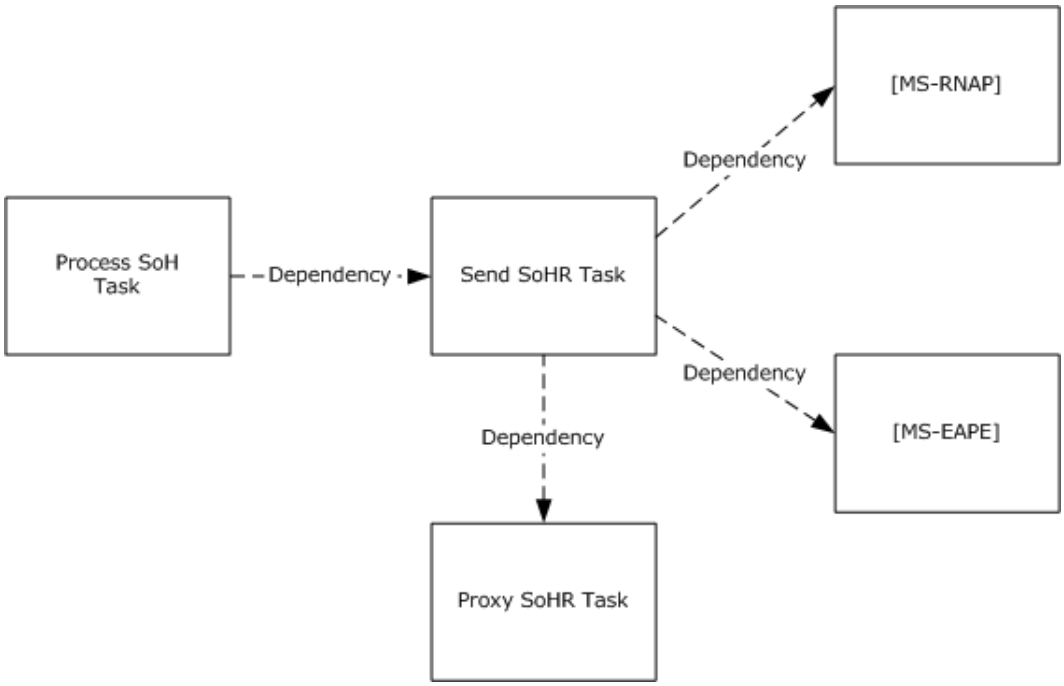

Figure 35: Send SoHR Task black-box relationships

9.2.2.2 Task Dependencies

As shown in the previous figure, the Send SoHR Task depends on the Process SoH Task, as it processes the NAP request data the Process SoH Task passes to it.

The Send SoHR Task depends on the MS-RNAP server to receive the NAP response data and to transport the data to the corresponding MS-RNAP client on the NAP Enforcement Point.

The Send SoHR Task depends on the MS-CEAP server to receive the NAP response data, encapsulate the SoHR message and to transport the data to the corresponding MS-CEAP client on the NAP client.

The Proxy SoHR Task depends on the Send SoHR Task. This is because the Proxy SoHR Task must rely on the Send SoHR Task to send NAP response data to the MS-RNAP client, for inclusion within its protocol messages. If the Send SoHR Task does not send the NAP response data, the NAP Proxy has no data to process.

9.2.2.3 Task Influences

None.

9.2.3 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.

9.2.4 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.

9.3 Task Architecture

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

9.3.1 Task Architectural Constraints

There can be more than one instance of the 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.

9.3.2 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.

None.

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.

Name	Type	Description
AuthResponse	Blob	Binary blob containing response from the authentication server.
AuthResult	DWORD	0 = Authentication Rejected 1 = Authenticated 2 = Authentication Challenge
SoHResponse	[MS-SOH] section 2.2.6	A SoHR message created by the SoH server.
SoHRespLength	DWORD	Length, in bytes, of the SoHR message.
CorrelationId	24-byte GUID	A correlation ID containing a unique transaction identifier shared across SoH and SoHR messages.
NetworkAccessServerType	DWORD	1 = Terminal Server Gateway 2 = Remote Access Service (RAS) server (VPN or dial-in) 3 = DHCP server 5 = Health Registration Authority (HRA) 6 = Host Credential Authorization Protocol (HCAP) server
IPFilter	[MS-RNAP] section 2.2.1.3	Set of IP filters to be set on NAP Client.
DhcpUserClass	String	DHCP user class to assign the NAP client to.
NotQuarantineCapable	Boolean	TRUE if the NAP Client is NAP capable (SoH message sent in RNAP AccessRequest).
QuarantineState	Integer	0 = Full Access 1 = Limited Access (remediation) 2 = On probation until grace time
GraceTime	Integer	Deadline for NAP Client to conform to NAP policy (become healthy), expressed as number of seconds since 1/1/1970 UTC.
AfwZone	Integer	1 = A boundary policy (requires encryption) should be used by IPsec. 2 = An unprotected policy (does not require encryption) should be used by IPsec. 3 = A protected policy (requires encryption) should be

Name	Type	Description
		used by IPsec.
AfwProtectionLevel	Integer	1 = HCEP certificate can be used only for signing. 2 = HCEP certificate can be used for signing and encrypting.
IPv4RemediationServers	[MS-RNAP] Section 2.2.1.16	A list of IPv4 servers to be used by the NAP Client for remediation.
IPv6RemediationServers	[MS-RNAP] Section 2.2.1.17	A list of IPv6 servers to be used by the NAP Client for remediation.
TsguRedirection	[MS-RNAP] Section 2.2.1.27	Redirection specification for Remote Desktop.

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.

This task is always run asynchronously and never returns values to the caller.

9.3.5 White-Box Relationships

The following diagram shows the white-box relationships for the Send SoHR Task.

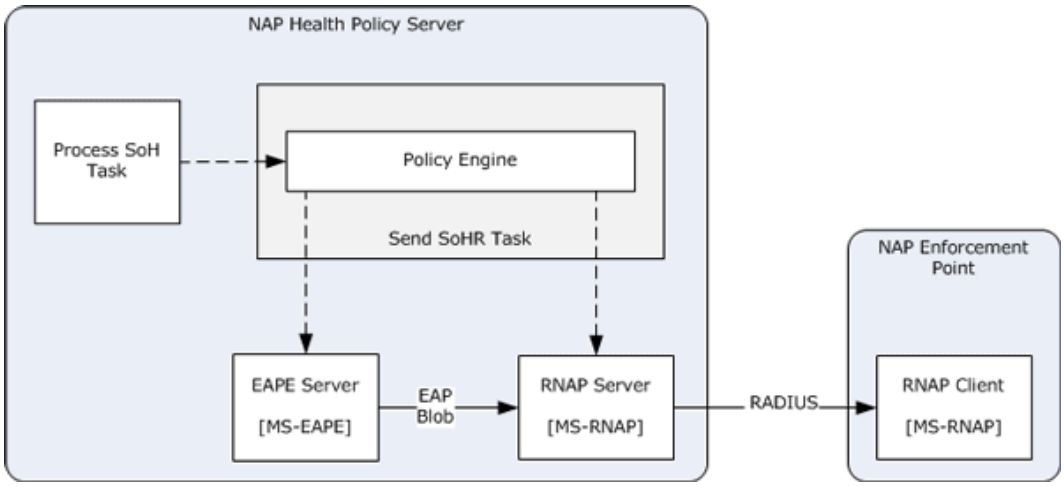

Figure 36: Send SoHR Task white-box relationships

The 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 [MS-SOH] 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 NEP.

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 NEP.

9.3.6 Task Events

9.3.6.1 Task Timers

The system does not define any task timers beyond those protocols supported by the system and defined in [MS-RNAP].

9.3.6.2 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].

9.3.7 Task Architecture and Communication

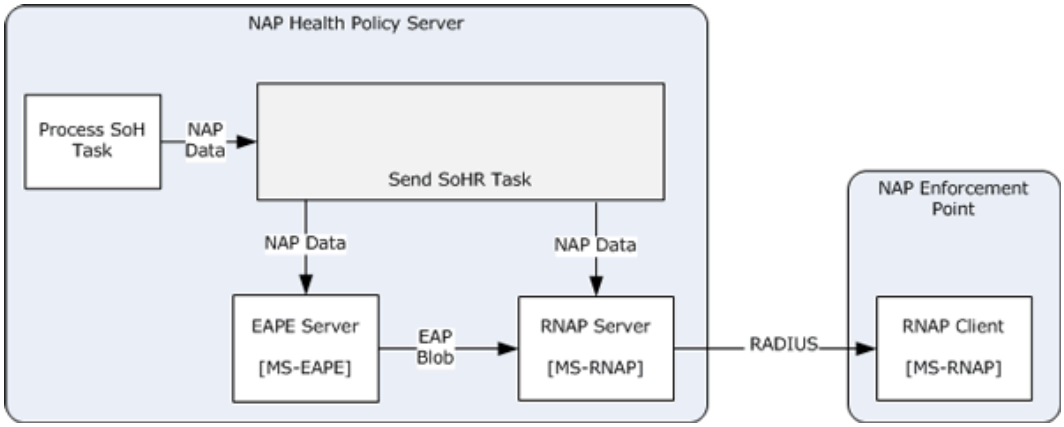

Figure 37: Send SoHR Task architecture and communication

The diagram shows the architectural details and the interworking between the SoH Server, RNAP Server, and PEAP Server components to accomplish the Send SoHR Task, and its supporting (dependent and influenced) tasks.

The 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 [MS-SOH], and sends it directly to NEP.

The Send SoHR Task is triggered on the PDP after the receipt of an evaluated SoH results.

9.3.8 Task Processing Rules

The following describes the operational flow of the Send SoHR Task:

1. The Process SoH Task invokes this task in its last processing rule.

2. If NetworkAccessServerType equals 2, the Policy Engine calls the SendRadiusReply abstract interface in [MS-CEAP] with all the task's abstract parameters.

3. If NetworkAccessServerType equals 1, 3, 5 or 6, the Policy Engine calls the SendRadiusReply abstract interface in [MS-RNAP] with all the task's abstract parameters.

If an error is raised at any stage of the Send SoHR Task, the Policy Engine logs an error and exits.

9.3.9 Task Failure Scenarios

9.3.9.1 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 NEP, 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.

9.3.9.2 NAP Health Policy Server and NEP communication

These failures can be caused by:

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

If the NAP health policy server cannot communicate with the NEP, the NAP health policy server may not send any RADIUS messages to the NEP. The system cannot recover from this failure. This failure cannot be detected by the NAP Server because RADIUS uses UDP.

9.3.9.3 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

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 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.

9.4.2 Task Initialization of External Entities

None.

9.4.3 Task Event Details

9.4.3.1 Task Timer Details

There are no task timer events for this task.

9.4.3.2 Task Non-Timer Event Details

None.

9.4.4 Task Architectural Details

This section illustrates an example of a PDP (NAP health policy server) creating an SoHR and sending it directly to the NEP. 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 as specified in [MS-PEAP] section 3.3, which in turn sends it using the PEAP channel.

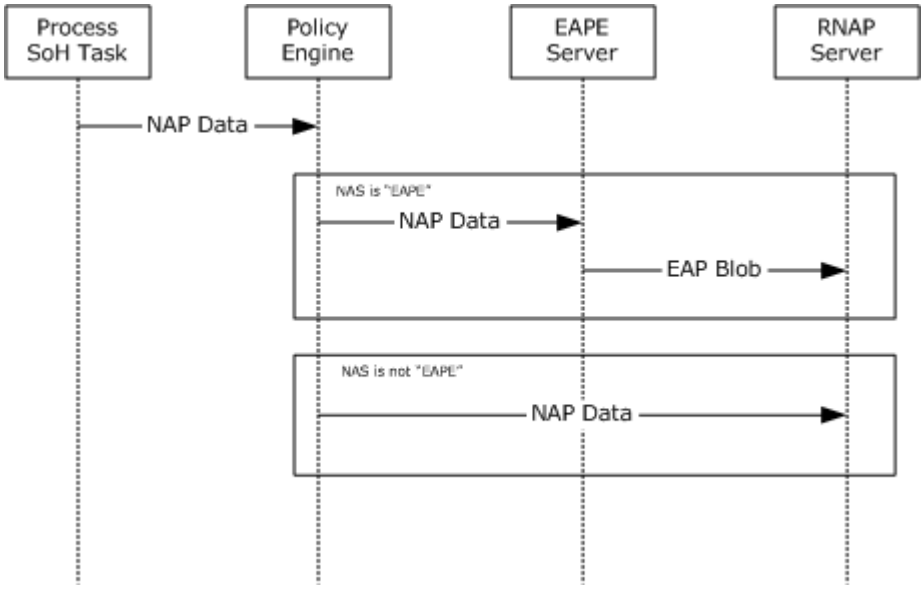

Figure 38: Sequence diagram for the main success scenario of the Send SoHR Task

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-CEAP] 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-CEAP] 2.2.4.2.3.
    2. The resulting EAP message is encapsulated using the EAP-Message attribute as specified in [RFC3579] and sent to the NEP 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 NEP via the RNAP Channel as defined in [MS-RNAP].

9.4.5 Task Processing Rule Details

The following describes the operational details of the Send SoHR Task:

1. The Process SoH Task invokes this task in the last processing rule in section 8.4.5.
2. If NetworkAccessServerType equals 2, the Policy Engine calls the SendRadiusReply abstract interface (section 3.1.7 Other Local Events) in [MS-CEAP] with all the task's abstract parameters (see section 9.3.3).
3. If NetworkAccessServerType equals 1, 3, 5 or 6, the Policy Engine calls the SendRadiusReply abstract interface (section 3.2.4.1 Sending a non-EAP Reply) in [MS-RNAP] with all the task's abstract parameters (see section 9.3.3).
4. If NetworkAccessServerType does not equal 1, 2, 3, 5 or 6, and error is logged and the task exits.

9.5 Task Security

The security consideration for this task is that the NEP and the NAP health policy server must maintain a trust relationship.

For additional information about security considerations, see section 12, as well as the Security sections of the referenced protocol Technical Documents.

10 Proxy SoHR Task

This section describes the task of proxying the NAP specific data from the RNAP client to the NAP transport protocol servers. The NAP response data is received from the incoming RNAP client and passed to the NAP Proxy. The NAP response data is then passed by the NAP Proxy to one of the different transport protocol servers by calling an abstract interfaces provided by the transport protocol.

10.1 Task Overview

10.1.1 Task Purpose

The purpose of this task is to proxy NAP response data from the RNAP client to the NAP transport protocol ([MS-DHCPN], [MS-HCEP], [MS-TSGU] and [MS-EAPE]) servers.

10.1.2 Task Applicability

This task is used when NAP specific data is received by the NAP Proxy. This task is not applicable if a NAP System is not deployed.

10.1.3 Task Use Cases

10.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Proxy SoHR Task are as follows:

RNAP client: This protocol client is used to send Vendor-Specific RADIUS Attributes for NAP [MS-RNAP] messages to the NAP health policy server. The interest of this actor in the task is that the NAP response data passed to the task is processed.

NAP Proxy: The NAP Proxy is used to proxy NAP response data (including the SoH) passed into the task abstract parameters by the RNAP Client to the NAP transport protocol ([MS-DHCPN], [MS-HCEP] and [MS-TSGU]) servers. The NAP Proxy's interest in the task is that the task parameters are passed on to the proper NAP transport protocol abstract interface based on valid correlation IDs.

SoH Server: The purpose of this actor is to utilize the processing rules defined in [MS-SOH] to create SoHR packets, which will be consumed by the SoH Client to remediate the health of the NAP Client. The SoH Server's interest in the task is that any proxied NAP response data includes the [MS-SOH] message created in the Process SoH Task.

WSH Server: The purpose of this actor is to utilize the processing rules defined in [MS-WSH] to create SoHREntry packets, which will be consumed by the WSH Client. The WSH Server's interest in the task is that any proxied NAP response data includes the [MS-WSH] message created in the Process SoH Task.

10.1.3.2 Supporting Actors and Task Interests Summary

HCEP Server: This protocol server is used to send HCEP Protocol [MS-HCEP] messages to an HCEP client on the NAP Client. It also acts as a policy enforcement point, using protocol specific behaviors in its enforcement. The task employs this actor whenever NAP response data needs to be sent back to the HCEP client.

DHCPN Server: This protocol server is used to send (DHCP) Extensions for NAP [MS-DHCPN] messages to a DHCPN client on the NAP Client. It also acts as a policy enforcement point, using

protocol specific behaviors in its enforcement. The task employs this actor whenever NAP response data needs to be sent back to the DHCPN client.

TSGU Server: This protocol server is used to send TSGU Protocol [MS-TSGU] messages to a TSGU client on the NAP Client. It also acts as a policy enforcement point, using protocol specific behaviors in its enforcement. The task employs this actor whenever NAP response data needs to be sent back to the TSGU client.

EAPE Proxy: This protocol server is used to send EAPE Protocol [MS-EAPE] messages to an EAPE peer on the NAP Client. It also acts as a policy enforcement point, using protocol specific behaviors in its enforcement. The task employs this actor whenever NAP response data needs to be sent back to the EAPE client.

10.1.3.3 Use Case Diagrams

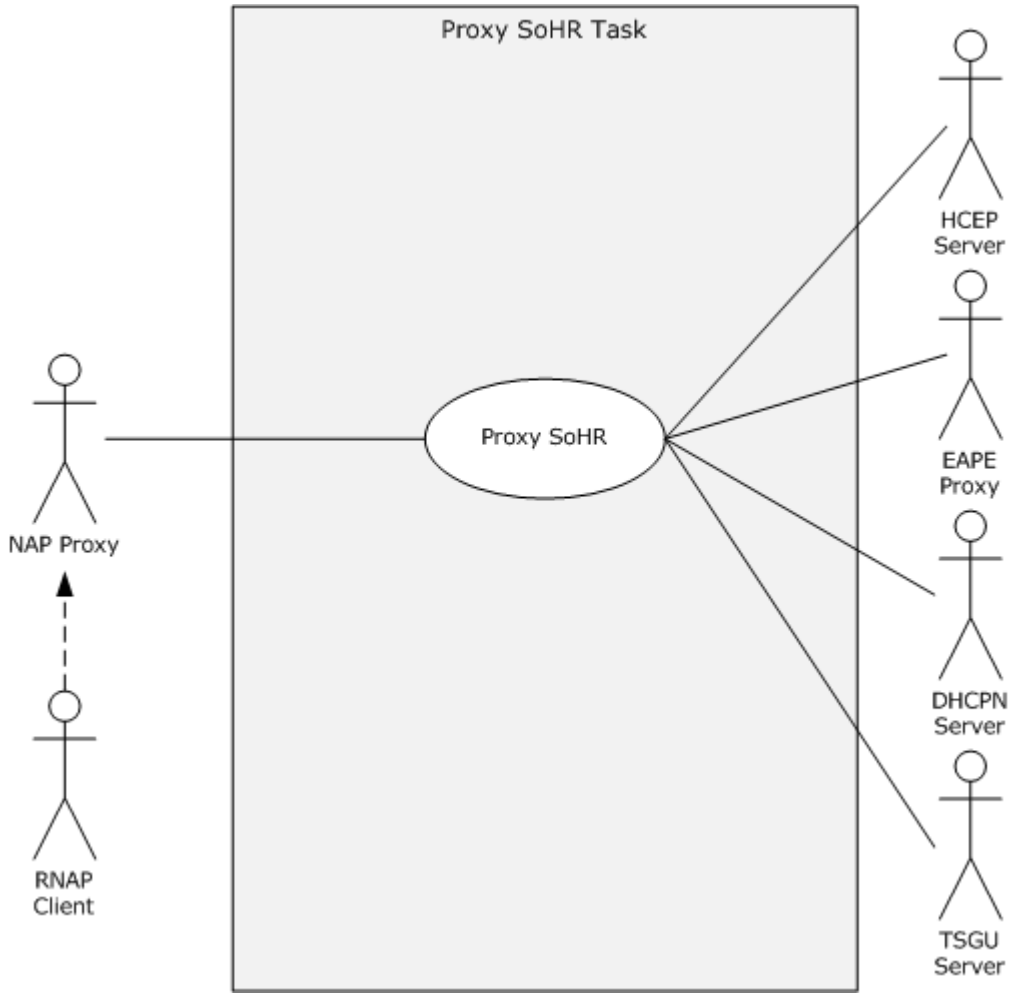

Figure 39: Proxy SoHR Task use case diagram

10.1.3.4 Use Case: Proxy SoHR -- NAP Enforcement Point

Goal: To proxy the NAP response data from the RNAP Client to the DHCPN Server, the EAPE Proxy, the HCEP Server or the TSGU Server.

Context of Use: This use case is used when NAP response data is passed by a protocol server to the NAP Proxy. The direct actor is internal to the task.

Direct Actor: This role is performed by the NAP Proxy.

Primary Actor: This role is performed by the RNAP client.

Supporting Actors:

HCEP Server: This protocol server is used to process HCEP Protocol [MS-HCEP] messages from an HCEP client on the NAP Enforcement Point. It also acts as a policy enforcement point, using protocol specific behaviors in its enforcement. The use case employs this actor whenever NAP response data needs to be sent back to the HCEP client.

DHCPN Server: This protocol server is used to Process (DHCP) Extensions for NAP [MS-DHCPN] messages from a DHCPN client on the NAP Enforcement Point. It also acts as a policy enforcement point, using protocol specific behaviors in its enforcement. The use case employs this actor whenever NAP response data needs to be sent back to the DHCPN client.

TSGU Server: This protocol server is used to process TSGU Protocol [MS-TSGU] messages from a TSGU client on the NAP Enforcement Point. It also acts as a policy enforcement point, using protocol specific behaviors in its enforcement. The use case employs this actor whenever NAP response data needs to be sent back to the TSGU client.

EAPE Proxy: This protocol server is used to process EAPE Protocol [MS-EAPE] messages received from an EAPE peer on the NAP Enforcement Point. It also acts as a policy enforcement point, using protocol specific behaviors in its enforcement. The task employs this actor whenever NAP response data needs to be sent back to the EAPE client.

Stakeholders and Interests: The stakeholders are defined as follows:

SoH Server: The purpose of this actor is to utilize the processing rules defined in [MS-SOH] to create SoHR packets, which will be consumed by the SoH Client to remediate the health of the NAP Client. The SoH Server's interest in the use case is that any proxied NAP response data includes the [MS-SOH] message created in the Process SoH Task.

WSH Server: The purpose of this actor is to utilize the processing rules defined in [MS-WSH] to create SoHREntry packets, which will be consumed by the WSH Client. The WSH Server's interest in the use case is that any proxied NAP response data includes the [MS-WSH] message created in the Process SoH Task.

Preconditions: The RNAP server successfully received the RADIUS Access-accept message and extracted the NAP response data from it.

Minimal Guarantees:

- The use case will always process the task abstract parameters passed to it.
- The proper NAS server abstract interface will be invoked, with properly mapped parameters, for valid correlation IDs.
- The [MS-SOH] message, if present, will be proxied with the rest of the NAP data.

- The [MS-WSH] message, if present, will be proxied with the rest of the NAP data.

Success Guarantee: The NAP response data is successfully sent to the NAP transport protocol ([MS-DHCPN], [MS-HCEP] and [MS-TSGU]) server.

Trigger: The task is triggered by RNAP Client invoking the task's abstract interface.

Main Success Scenario:

1. The task is triggered by RNAP Client invoking the task's abstract interface.
2. The NAP Proxy receives the task abstract parameters.
3. The NAP Proxy finds the correct entry in the NasTypeTable based on the passed in correlation ID, and retrieves the caller reference, NasRef.
4. If the Network Access Server type equals 1:
   - The NAP Proxy invokes an [MS-TSGU] abstract interface on the on the thread referenced by NasRef.
   - The NAP Proxy passes values to the invoked [MS-TSGU] abstract interface by mapping the task abstract parameters to the parameter listing of the [MS-TSGU] abstract interface.
5. If the Network Access Server type equals 2:
   - The NAP Proxy invokes an [MS-EAPE] abstract interface on the thread referenced by NasRef.
   - The NAP Proxy passes values to the invoked [MS-EAPE] abstract interface by mapping the task abstract parameters to the parameter listing of the [MS-EAPE] abstract interface.
6. If the Network Access Server type equals 3:
   - The NAP Proxy invokes an [MS-DHCPN] abstract interface on the thread referenced by NasRef.
   - The NAP Proxy passes values to the invoked [MS-DHCPN] abstract interface by mapping the task abstract parameters to the parameter listing of the [MS-DHCPN] abstract interface.
7. If the Network Access Server type equals 5:
   - The NAP Proxy invokes an [MS-HCEP] abstract interface on the on the thread referenced by NasRef.
   - The NAP Proxy passes values to the invoked [MS-HCEP] abstract interface by mapping the task abstract parameters to the parameter listing of the [MS-HCEP] abstract interface.
8. If the Network Access Server type equals 6:
   - The NAP Proxy invokes an HCAP abstract interface on the on the thread referenced by NasRef.
   - The NAP Proxy passes values to the invoked HCAP abstract interface by mapping the task abstract parameters to the parameter listing of the HCAP abstract interface.
9. The NAP Proxy removes the entry from the NasTypeTable corresponding to the passed in correlation ID.

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 the NAP Proxy in an environment where the NAP response data is transferred from the RNAP client to the NAP transport protocol ([MS-DHCPN], [MS-HCEP], [MS-TSGU] and [MS-EAPE]) servers. The environment should meet the following requirement to support this task.

Requirement: The RNAP Client is running and has the ability to receive [MS-RNAP] packets from the RNAP server on the NAP Health Policy Server (NPS).

- **Reason for requirement**: The RNAP client is used to receive NAP response data (including the SoHR) encapsulated within [MS-RNAP] packets from the NPS.
- **Means of satisfying the requirement**:

  1. The RNAP client is configured with the RADIUS settings from the registry.
  2. The RNAP client has network access to the RNAP Server:
     1. The network interface of the NEP is configured to operate on the local subnet.
     2. The physical network path (network devices, Ethernet cables, etc.) 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 routing tables in the NEP are configured to enable correct packet routing between the NPS and the NEP.
  3. The RNAP client service has been started.

- **Means of knowing requirement satisfied**:

  1. The NEP can successfully ping the NPS over the network.
  2. The RNAP client is shown as running within the list of services.
  3. A sniffer trace performed during a RADIUS access-request event, shows RADIUS packets traveling between the NPS and the NEP. These must be "access-accept" RADIUS packets containing [MS-RNAP] fields.
  4. No errors are logged by the RNAP client.

- **Consequences of not satisfying requirement**: The task is unable to receive NAP response data via the [MS-RNAP] protocol. As a result, the task will never be invoked.

Requirement: The HCEP server is running and has the ability to send [MS-HCEP] packets to the HCEP client on the NAP Client.

- **Reason for requirement**: The HCEP server is used to send NAP response data (including the SoHR) encapsulated within [MS-HCEP] packets to the NAP Client.
- **Means of satisfying the requirement**:

1. The HCEP server is configured with the HCEP server settings from the registry.
2. The HCEP server has network access to the HCEP client:
    1. The network interface of the NEP is configured to operate on the local subnet.
    2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NAP Client is connected.
    3. All network devices between the local subnet and the NAP Client are configured to allow packet flow between the two entities.
    4. The routing tables in the NEP are configured to enable correct packet routing between the NAP Client and the NEP.
3. The HCEP server service has been started.

- **Means of knowing requirement satisfied**:

1. The NEP can successfully ping the NAP Client over the network.
2. The HCEP server is shown as running within the list of services.
3. A sniffer trace performed during a Health Certificate Enrollment event, shows HTTP packets traveling between the NAP Client and the NEP. These HTTP packets must contain [MS-HCEP] fields.
4. No errors are logged by the HCEP server.

- **Consequences of not satisfying requirement**: The task is unable to send NAP response data via the [MS-HCEP] protocol.

Requirement: The HCEP server is able to access the Certificate Authority and utilize its services.

- **Reason for requirement**: The HCEP server uses the CA to fetch IPSec certificates, if the client is deemed healthy.
- **Means of satisfying the requirement**:

1. The HCEP server is configured with the CA server settings from the registry.
2. The HCEP server has network access to the CA:
    1. The network interface of the NEP is configured to operate on the local subnet.
    2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the CA is connected.
    3. All network devices between the local subnet and the CA are configured to allow packet flow between the two entities.
    4. The routing tables in the NEP are configured to enable correct packet routing between the CA and the NEP.
3. The HCEP server service has been started.
4. The certificate services are started on the CA.

- **Means of knowing requirement satisfied**:

1. The NEP can successfully ping the CA over the network.
2. The HCEP server is shown as running within the list of services on the NEP.
3. The certificate services are shown as running within the list of services on the CA.
4. A sniffer trace performed during a Health Certificate Enrollment event, shows WCCE packets traveling between the CA and the NEP. These WCCE packets must contain fields as described in section 2.2.1.4 Health Certificate Request and section 2.2.2.4 Health Certificate Response of [MS-HCEP].
5. No errors are logged by the HCEP server.

- **Consequences of not satisfying requirement**: The HCEP server is unable to include IPSec certificates in the [MS-HCEP] protocol response.

Requirement: The DHCPN server is running and has the ability to send [MS-DHCPN] packets to the DHCP client on the NAP Client.

- **Reason for requirement**: The DHCPN server is used to send NAP response data (including the SoHR) encapsulated within [MS-DHCPN] packets to the NAP Client.
- **Means of satisfying the requirement**:

1. The DHCPN server is configured with the DHCP settings from the registry.
2. The DHCPN server has network access to the DHCP client:
    1. The network interface of the NEP is configured to operate on the local subnet.
    2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NAP Client is connected.
    3. All network devices between the local subnet and the NAP Client are configured to allow packet flow between the two entities.
    4. The routing tables in the NEP are configured to enable correct packet routing between the NAP Client and the NEP.
3. The DHCPN server service has been started.

- **Means of knowing requirement satisfied**:

1. The NEP can successfully ping the NAP Client over the network.
2. The DHCPN client is shown as running within the list of services.
3. An attempt to renew the IP address lease on each NIC via the DHCP Server on the NEP completes successfully.
4. A sniffer trace performed during a DHCP lease renewal event, shows DHCP packets traveling between the NAP Client and the NEP. These DHCP packets must contain [MS-DHCPN] fields.
5. No errors are logged by the DHCPN server.

- **Consequences of not satisfying requirement**: The task is unable to send NAP response data via the [MS-DHCPN] protocol.

Requirement: The TSGU server is running and has the ability to send [MS-TSGU] packets to the TSGU client on the NAP Client.

- **Reason for requirement**: The TSGU client is used to send NAP response data (including the SoHR) encapsulated within [MS-TSGU] packets to the NAP Client.
- **Means of satisfying the requirement**:

  1. The TSGU server is configured with the TSG settings from the registry.
  2. The TSGU server has network access to the TSGU client:
     1. The network interface of the NEP is configured to operate on the local subnet.
     2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NAP Client is connected.
     3. All network devices between the local subnet and the NAP Client are configured to allow packet flow between the two entities.
     4. The routing tables in the NEP are configured to enable correct packet routing between the NAP Client and the NEP.
  3. The TSGU server service has been started.

- **Means of knowing requirement satisfied**:

  1. The NEP can successfully ping the NAP Client over the network.
  2. The TSGU server is shown as running within the list of services.
  3. An attempt to connect to a remote desktop via the Terminal Services Gateway on the NEP completes successfully.
  4. A sniffer trace performed during a RDP connection event, shows TSGU packets traveling between the NAP Client and the NEP. These TSGU packets must contain [MS-TSGU] fields.
  5. No errors are logged by the TSGU server.

- **Consequences of not satisfying requirement**: The task is unable to send NAP data via the [MS-TSGU] protocol.

Requirement: The EAPE proxy is running and has the ability to send [MS-EAPE] packets to the EAPE peer on the NAP Client

- **Reason for requirement**: The EAPE proxy is used to send NAP response data (including the SoHR) encapsulated within [MS-EAPE] packets to the NAP Client.
- **Means of satisfying the requirement**:

  1. The EAPE proxy is configured with the EAP/PEAP settings from the registry.
  2. The EAPE proxy has network access to the EAPE client:
     1. The network interface of the NAP Client is configured to operate on the local subnet.
     2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NEP is connected.

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

4. The routing tables in the NAP Client are configured to enable correct packet routing between the NAP Client and the NEP.

3. The EAPE peer service has been started.

- **Means of knowing requirement satisfied**:

1. The NEP can successfully ping the NAP Client over the network.

2. The EAPE proxy is shown as running within the list of services.

3. An attempt to VPN to a computer on the corporate network via the NEP completes successfully.

4. A sniffer trace performed during a VPN connection event, shows EAPE packets traveling between the NAP Client and the NEP. These EAPE packets must contain [MS-EAPE] fields.

5. No errors are logged by the EAPE proxy.

- **Consequences of not satisfying requirement**: The task is unable to send NAP response data via the [MS-EAPE] protocol.

10.2.2 Task Relationships

10.2.2.1 Black-Box Relationship Diagrams

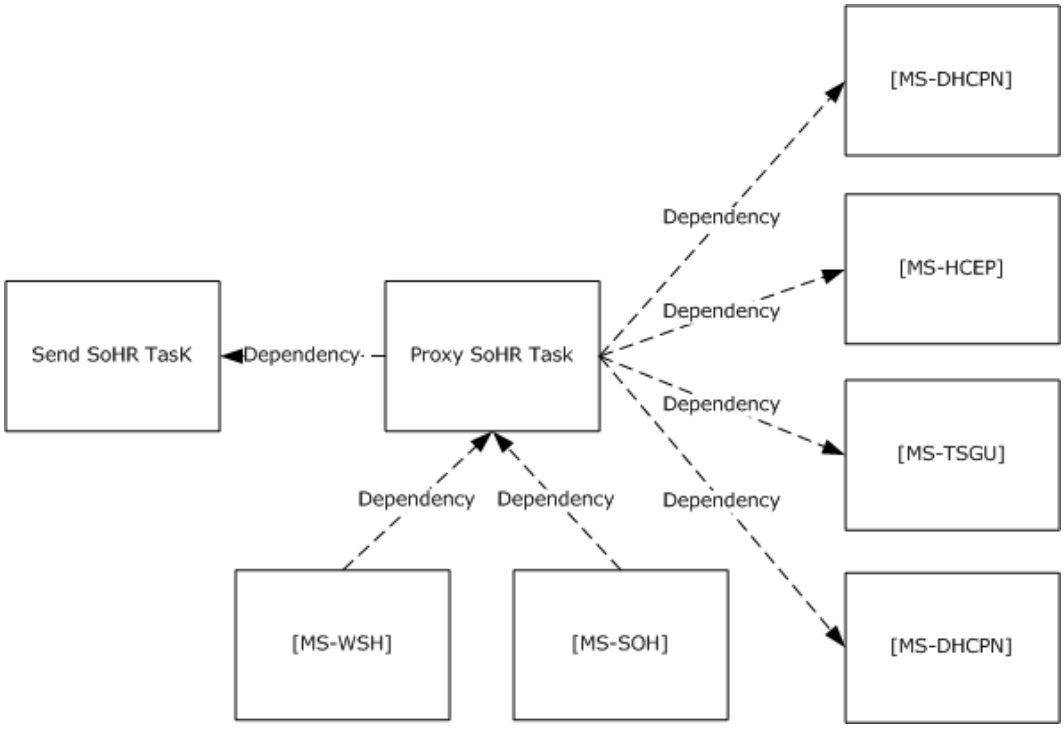

Figure 40: Proxy SoHR Task black-box relationships

In this task, the NAP health policy server sends encapsulated SoHR messages and enforcement decisions to the NAP Enforcement Point via the RADIUS channel.

10.2.2.2 Task Dependencies

The Proxy SoHR Task depends on the Send SoHR Task. This is because the Proxy SoHR Task must rely on the Send SoHR Task to send NAP response data to the MS-RNAP client, for inclusion within its protocol messages. If the Send SoHR Task does not send the NAP response data, the NAP Proxy has no data to process.

The Proxy SoHR Task depends on the servers of the NAP transport protocols (MS-HCEP, MS-TSGU, MS-EAPE and MS-DHCPN) to accept the NAP response data and to transport the data to its corresponding client on the NAP Client.

The MS-SoH protocol is dependent on the Proxy SoHR Task as the communication between the MS-SoH client and the MS-SoH server is totally reliant on the proxy steps executed by the Proxy SoHR Task.

The MS-WSH protocol is dependent on the Proxy SoHR Task as the communication between the MS-WSH client and the MS-WSH server is totally reliant on the proxy steps executed by the Proxy SoHR Task.

10.2.2.3 Task Influences

None.

10.2.3 Task Assumptions and Preconditions

To accomplish this task, the NAP Enforcement Point 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.

10.2.4 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.

10.3 Task Architecture

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

10.3.1 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.

10.3.2 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.

Following ADM elements are declared in (and shared by) the Proxy SoH Task (section 6.3.2).

Name	Type	Description
NasTypeTable	Table with two columns: 1. NasRef of type Caller Reference (thread ID, socket, callback pointer, etc.) 2. CorrId of type 24-byte GUID	A lookup table indexed by the correlation ID of the SoH message being proxied.

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.

Name	Type	Description
AuthResult	DWORD	0 = Authentication Rejected 1 = Authenticated 2 = Authentication Challenge
AuthResponse	Blob	Binary blob containing response from the authentication server.
SoHResponse	[MS-SOH] section 2.2.6	A SoHR message created by the SoH server.
SoHRespLength	DWORD	Length, in bytes, of the SoHR message.
CorrelationId	24-byte GUID	A correlation ID containing a unique transaction identifier shared across SoH and SoHR messages.
NetworkAccessServerType	DWORD	1 = Terminal Server Gateway 2 = Remote Access Service (RAS) server (VPN or dial-

Name	Type	Description
		in) 3 = DHCP server 5 = Health Registration Authority (HRA) 6 = Host Credential Authorization Protocol (HCAP) server
IPFilter	[MS-RNAP] section 2.2.1.3	Set of IP filters to be set on NAP Client.
DhcpUserClass	String	DHCP user class to assign the NAP client to.
NotQuarantineCapable	Boolean	TRUE if the NAP Client is NAP capable (SoH message sent in RNAP AccessRequest).
QuarantineState	Integer	0 = Full Access 1 = Limited Access (remediation) 2 = On probation until grace time
GraceTime	Integer	Deadline for NAP Client to conform to NAP policy (become healthy), expressed as number of seconds since 1/1/1970 UTC.
AfwZone	Integer	1 = A boundary policy (requires encryption) should be used by IPsec. 2 = An unprotected policy (does not require encryption) should be used by IPsec. 3 = A protected policy (requires encryption) should be used by IPsec.
AfwProtectionLevel	Integer	1 = HCEP certificate can be used only for signing. 2 = HCEP certificate can be used for signing and encrypting.
IPv4RemediationServers	[MS-RNAP] Section 2.2.1.16	A list of IPv4 servers to be used by the NAP Client for remediation.
IPv6RemediationServers	[MS-RNAP] Section 2.2.1.17	A list of IPv6 servers to be used by the NAP Client for remediation.
eapBlob	binary BLOB	A binary BLOB that contains all the EAP protocol layers, from the outer EAP (MS-EAPE) to the inner EAP (MS-CEAP).
eapBlobLength	DWORD	Length, in bytes, of the eapBlob.
eapBlobSignature	Binary Data	Signature of EAPBlob
eapBlobSigLength	DWORD	Length, in bytes, of the eapBlobSignature.
TsguRedirection	[MS-RNAP] Section 2.2.1.27	Redirection specification for Remote Desktop.

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.

This task is always run asynchronously and never returns values to the caller.

10.3.5 White-Box Relationships

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

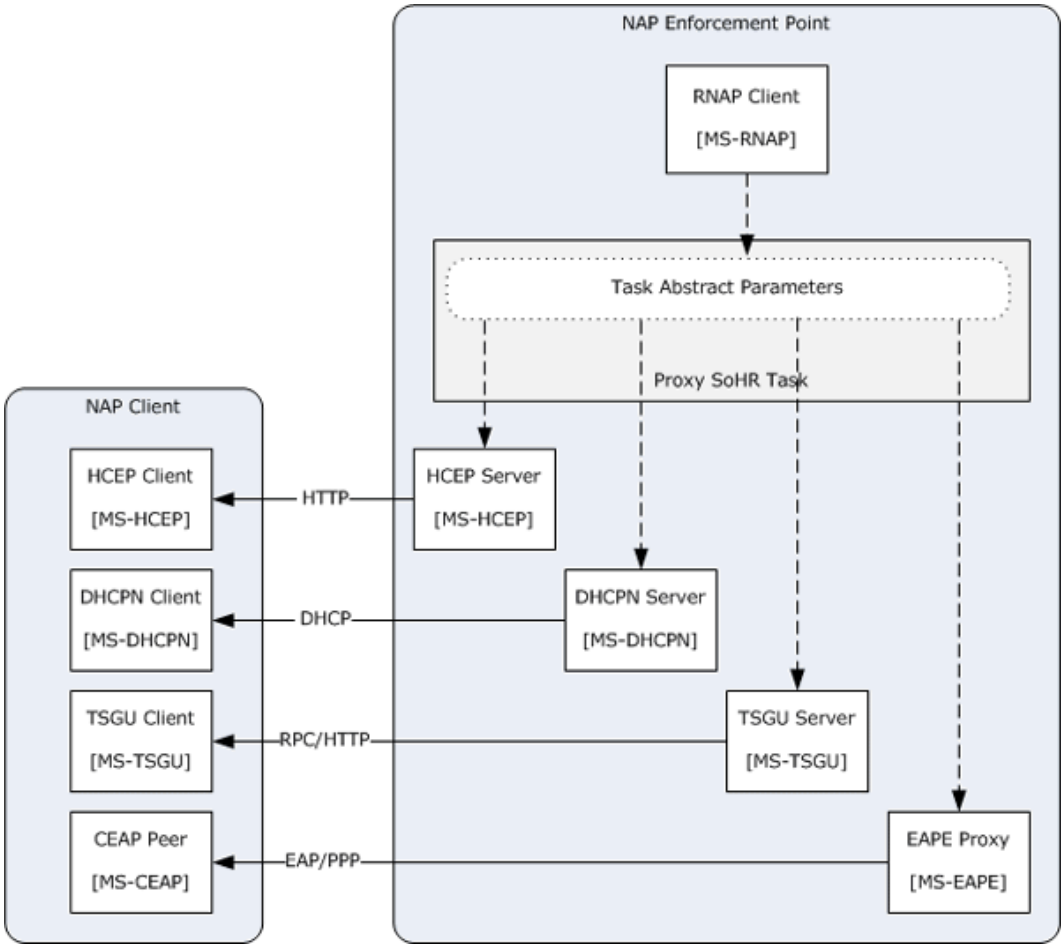

Figure 41: Proxy SoHR Task white-box relationships

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 Send SoHR Task (section 9) or the NEP, the Proxy SoHR Task provides SoHR encapsulation and transportation services.

10.3.6 Task Events

10.3.6.1 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.

10.3.6.2 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.

10.3.7 Task Architecture and Communication

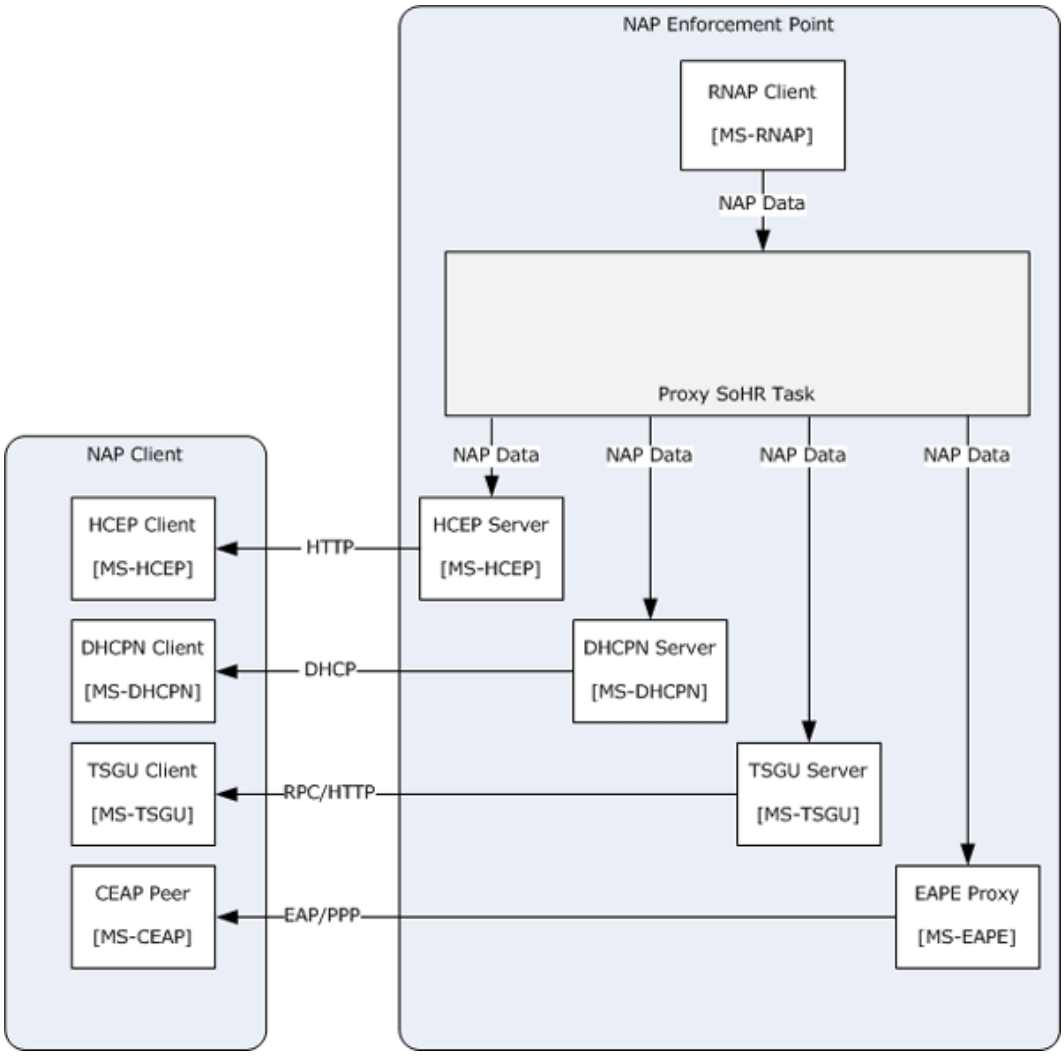

Figure 42: Proxy SoHR Task architecture and communication

10.3.8 Task Processing Rules

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

1. The RNAP Client invokes this task in one of three ways:
   - If the authentication was successful, the RNAP Client invokes the task with NAP response data and the AuthResult parameter set to 1.
   - If the authentication failed, the RNAP Client invokes the task without NAP response data and the AuthResult parameter set to 2.
   - If authentication specified a challenge response is needed, the RNAP Client invokes the task without NAP response data and the AuthResult parameter set to 3.
2. The NAP Proxy iterates through the NasTypeTable looking for an entry matching the CorrelationId parameter and retrieves the corresponding BasRef value.
3. If NetworkAccessServerType equals 1, the NAP Proxy calls the SoHRAsyncCallback abstract interface in [MS-TSGU], on the thread corresponding to NasRef.
4. If NetworkAccessServerType equals 2, the NAP Proxy calls the SendEAPReply abstract interface in [MS-EAPE], on the thread corresponding to NasRef.
5. If NetworkAccessServerType equals 3, the NAP Proxy calls the DHCPServerSetSoHR abstract interface in [MS-DHCPN], on the thread corresponding to NasRef.
6. If NetworkAccessServerType equals 5, the NAP Proxy calls the HCEPSetSoHR abstract interface in [MS-HCEP], on the thread corresponding to NasRef.
7. If NetworkAccessServerType equals 6, the NAP Proxy passes all the input parameters to the HCAP server.
8. The NAP Proxy iterates through the NasTypeTable removing all entries matching the CorrelationId parameter.

If an error is raised at any stage of the Proxy SoHR Task, the NAP Proxy logs an error and exits.

10.3.9 Task Failure Scenarios

10.3.9.1 NAP Health Policy Server and NEP 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 NEP.

If the NAP health policy server cannot communicate with the NEP, the NAP health policy server may not send any RADIUS messages to the NEP. The system cannot recover from this failure. This failure cannot be detected by the NAP server because RADIUS uses UDP.

10.3.9.2 NAP Client and NEP communication

These failures can be caused by:

- Mis-configurations on the NAP client and/or NEP.
- Network connectivity issues wherein the NAP client cannot communicate with the NEP.

If the NAP client cannot communicate with the NEP, 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.

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:

- Depending on the specific configuration, any of the required NEP channels are functioning correctly, including the HTTP/S channel, the TSGU channel, or the DHCP channel.
- The RADIUS channel between the NAP health policy server and the NEP is functioning correctly, and the NEP recognizes the RADIUS protocol.

10.4.2 Task Initialization of External Entities

None.

10.4.3 Task Event Details

10.4.3.1 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].

10.4.3.2 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].

10.4.4 Task Architectural Details

This section illustrates an example of a NAP health policy server sending an SoHR.

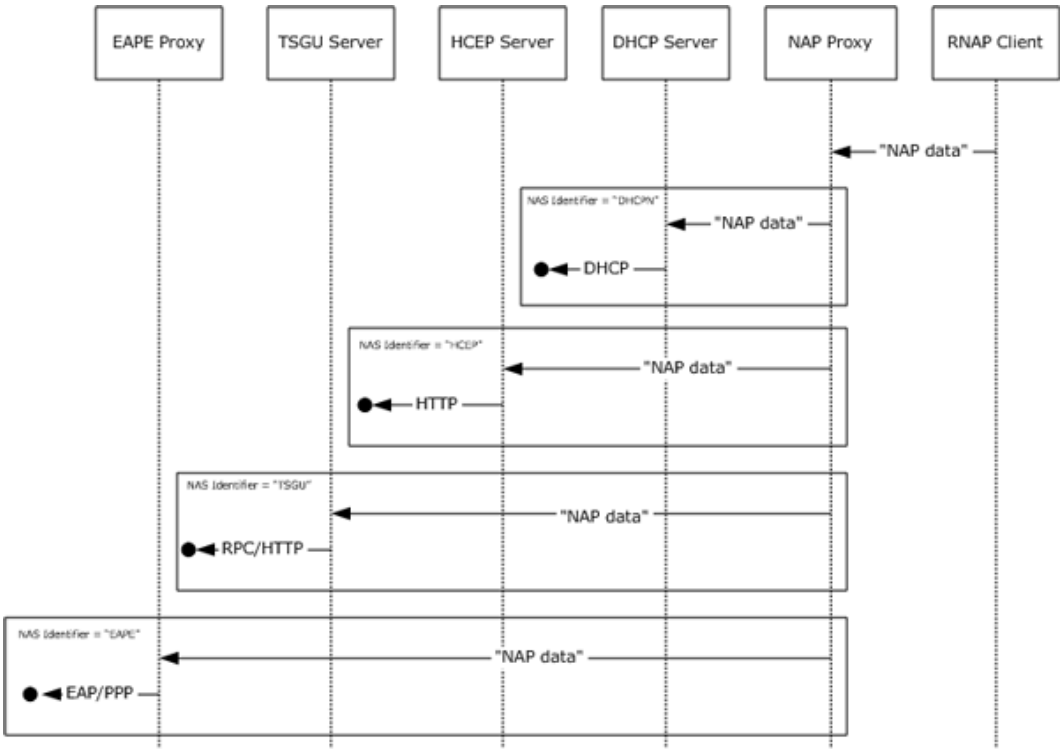

Figure 43: Sequence diagram for the main success scenario of the Proxy SoHR Task

1. An SoHR buffer is passed to the NAP Enforcement Point.
2. The NAP Enforcement Point passes the SoHR buffer to one of the NAP ESs (HCEP HRA, DHCPN Server, TSGU Server).

10.4.5 Task Processing Rule Details

The following describes the operational details of the Proxy SoHR Task:

1. The RNAP Client invokes this task in one of three ways:
   - If authentication was successful, the RNAP Client invokes the task with NAP response data as described in section 3.3.5.2 Processing RADIUS Access-Accept Messages in [MS-RNAP].
   - If authentication was unsuccessful, the RNAP Client invokes the task without NAP response data as described in section 3.3.5.3 Processing RADIUS Access-Reject Messages in [MS-RNAP].
   - If authentication specified a challenge response is needed, the RNAP Client invokes the task without NAP response data as described in section 3.3.5.4 Processing RADIUS Access-Challenge Messages in [MS-RNAP].
2. The NAP Proxy iterates through the NasTypeTable looking for an entry whose CorrId value matches the CorrelationId parameter it received from the [MS-RNAP] invocation:
   - If an entry is found, the corresponding NasRef value is saved locally.

- If multiple entries are found, the last entry is selected and its corresponding NasRef value is saved locally.
- If no entry, the NAP Proxy logs an error message and stops execution.

3. If NetworkAccessServerType equals 1, the NAP Proxy calls the SoHRAsyncCallback abstract interface (section 3.1.6 Other Local Events) in [MS-TSGU], on the thread corresponding to NasRef, with the following parameters:
   - X mapped to as X,
   - X mapped to as X, X mapped to X.
4. If NetworkAccessServerType equals 2, the NAP Proxy calls the SendEAPReply abstract interface (section 6.1.7.1.2 Forward a Message to EAPE Server) in [MS-EAPE], on the thread corresponding to NasRef, with the following parameters:
   - X mapped to as X,
   - X mapped to as X,
   - X mapped to X.
5. If NetworkAccessServerType equals 3, the NAP Proxy calls the DHCPServerSetSoHR abstract interface (section 3.2.7.3 DHCPServerSetSoHR) in [MS-DHCPN], on the thread corresponding to NasRef, with the following parameters:
   - X mapped to as X,
   - X mapped to as X,
   - X mapped to X.
6. If NetworkAccessServerType equals 5, the NAP Proxy calls the HCEPSetSoHR abstract interface (section 3.2.4 Higher-Layer Triggered Events) in [MS-HCEP], on the thread corresponding to NasRef, with the following parameters:
   - X mapped to as X,
   - X mapped to as X,
   - X mapped to X.
7. If NetworkAccessServerType equals 6, the NAP Proxy passes all the input parameters to the HCAP server.
8. If NetworkAccessServerType does not equal 1, 2, 3, 5 or 6, the NAP Proxy logs an error message and stops execution.
9. The NAP Proxy iterates through the NasTypeTable removing all entries whose CorrId value matches the CorrelationId parameter.

10.5 Task Security

The NEP and the NAP EC must maintain a trust relationship. For additional information about security considerations, see section 12, as well as the Security sections of the referenced protocol Technical Documents.

11 Process SoHR Task

This section describes the task of processing SoHR messages on a NAP Client. This task is performed by the SoH client and the NAP agent.

Note

11.1 Task Overview

11.1.1 Task Purpose

The purpose of this task is to pass the SoHR to the SoH client for evaluation and remediation. The remediation process is documented in [MS-SOH] and [MS-WSH].

11.1.2 Task Applicability

This task is used when NAP specific data is received by the NAP Agent. This task is not applicable if the NAP System is not deployed.

11.1.3 Task Use Cases

11.1.3.1 Stakeholders and Interests Summary

The stakeholders for the Process SoHR Task (section 11) are as follows:

NAP agent: The main software component on the NAP Client. 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 ability to perform its services is where the NAP agent's interests in this task are.

HCEP Client: This protocol client is used to receive [MS-HCEP] messages from an HCEP server on the NEP computer. In this use case, when HCEP is used, the HCEP Client acts in the role of an enforcement client (EC). The interest of this actor in the task is that the NAP response data passed to the task is processed.

DHCP Client: This protocol client is used to receive [MS-DHCPN] messages to a DHCPN server on the NEP computer. In this use case, when DHCPN is used, the DHCP Client acts in the role of an enforcement client (EC). The interest of this actor in the task is that the NAP response data passed to the task is processed.

TSGU Client: This protocol client is used to receive [MS-TSGU] messages from a TSGU server on the NEP computer. In this use case, when TSGU is used, the TSGU Client acts in the role of an enforcement client (EC). The interest of this actor in the task is that the NAP response data passed to the task is processed.

CEAP Peer: This protocol client is used to receive [MS-CEAP] messages from a CEAP server on the NEP computer. In this use case, when CEAP is used, the CEAP Peer acts in the role of an enforcement client (EC). The interest of this actor in the task is that the NAP response data passed to the task is processed.

SHVs: The purpose of these actors is to create SoHREntry fields, which are encapsulated within the SoHR message. The SoHREntry fields are based on an evaluation of the SoH message and information provided by the Health Requirement Servers. The SHVs' interest in the task is that the SoHR message, along with the SoHREntry fields, is passed on to the SoH Client for processing.

11.1.3.2 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 [MS-SOH] to evaluate SoHR packets. This is accomplished by extracting the correlation ID, the SSoH header prepended to the SoHR packet, and the SoHRReportEntry as specified in [MS-SOH] section 2.2.6.3. This task uses the SoH client to process the SoHR message and perform any required remediation.

Authentication Client: This actor is used to process the authentication response, including updating Credential Cache, notifying the user of authentication failures and responding to authentication challenges. The task employs this actor whenever authentication data needs to be processed.

11.1.3.3 Use Case Diagrams

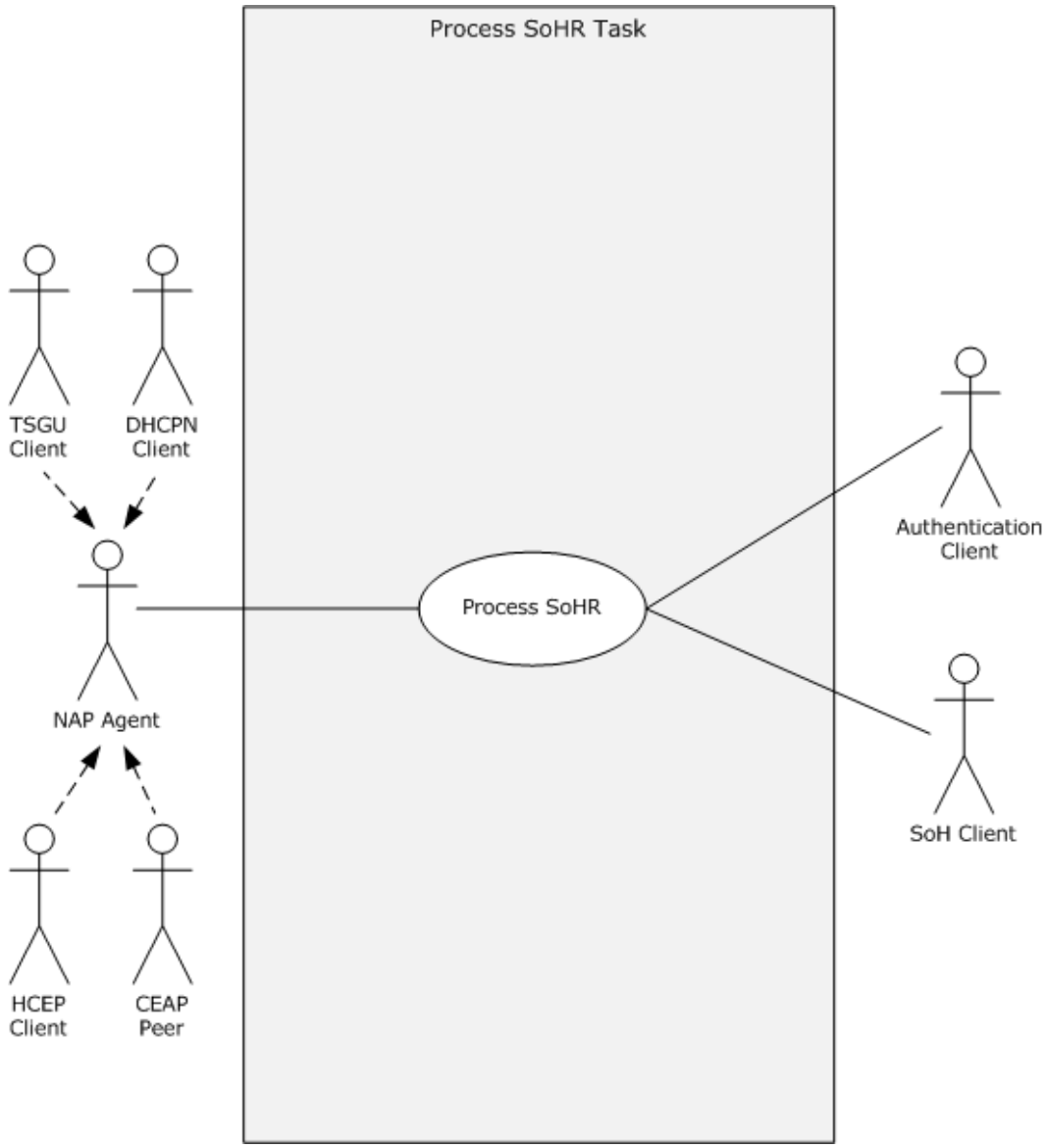

Figure 44: Process SoHR Task use case diagram

11.1.3.4 Use Case: Process SoHR - NAP Agent

This use case is associated with the use case diagram in section 11.1.3.3.

Goal: To process the validity of SoHR messages and trigger if the NAP Client is noncompliant.

Context of Use: This use case is initiated when an SoHR has been received by the SoH client.

Direct Actor: This role is performed by the NAP agent.

Primary Actor: This role is performed by four actors: the DHCPN Client, the HCEP Client, the TSGU Client and the CEAP Peer.

Supporting Actors:

SOH client: This actor uses the Statement of Health for Network Access Protection (NAP) Protocol processing rules specified in [MS-SOH] to evaluate SoHR packets. This is accomplished by extracting the correlation ID, the SSoH header prepended to the SoHR packet, and the SoHRReportEntry as specified in [MS-SOH] section 2.2.6.3. This use case employs the SoH client to process the SoHR message and perform any required remediation.

Authentication Client: This actor is used to process the authentication response, including updating Credential Cache, notifying the user of authentication failures and responding to authentication challenges. The use case employs this actor whenever authentication data needs to be processed.

Stakeholders and Interests:

SHVs: The purpose of these actors is to create SoHEntry fields, which are encapsulated within the SoH message. Those SoHEntry fields are evaluated by the corresponding SHVs on the NAP Health Policy Server, which in turn create SoHREntry fields for use by the SHAs on the NAP Client. The SHAs' interest in the task is that any SoH message created on the NAP Client, along with the SoHEntry fields, is passed on to the SoH Server for processing.

Preconditions: The NAP client components on the NAP Client are deployed and configured correctly by the client administrator.

Minimal Guarantees:

- NAP related operations are performed, without guaranteed success.
- The NAP response data will always be processed.
- The SoHR message, if present, will be sent to the SoH Client for processing, along with the encapsulated SoHREntry fields.

Success Guarantee: The SoH client successfully completes processing of the SoHR message and the authentication client successfully processes the authentication data.

Trigger: The Task can be triggered by any of the following:

- A DHCPN Client thread invokes the abstract interface of this task.
- A HCEP Client thread invokes the abstract interface of this task.
- A TSGU Client thread invokes the abstract interface of this task.

- A CEAP Peer thread invokes the abstract interface of this task.

Main Success Scenario:

1. The task is triggered by a thread associated with one of the following direct actors invoking the task's abstract interface:
   - The DHCPN Client.
   - The HCEP Client.
   - The TSGU Client.
   - The CEAP Peer.
2. The NAP agent calls the ProcessSoHResponse abstract interface in [MS-SOH] with the SoHResponse parameter to process the SoHR message.
3. The NAP agent processes the authentication response by sending the AuthResponse parameter to the Authentication Client.
4. If the Authentication Client responds that a new round of authentication is required (due to a challenge response or authentication failure), the NAP Agent sends an NAP re-authentication system event.

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 the NAP Agent in an environment where NAP Client request access to network resources under the control of devices or servers acting as Network Enforcement Points (NEP). The environment should meet the following requirement to support this task.

Requirement: The HCEP Client is running and has the ability to receive [MS-HCEP] packets from the HCEP server on the Network Enforcement Point (NEP).

- **Reason for requirement**: The HCEP client is used to receive NAP response data (including the SoHR) encapsulated within [MS-HCEP] packets from the NEP.
- **Means of satisfying the requirement**:
  1. The HCEP client is configured with the HCEP server settings described in section 2.4 Health Registration Authority (HRA) Settings in [MS-GPNAP].
  2. The HCEP client has network access to the HCEP Server:
     1. The network interface of the NAP Client is configured to operate on the local subnet.
     2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NEP is connected.
     3. All network devices between the local subnet and the NEP are configured to allow packet flow between the two entities.
     4. The routing tables in the NAP Client are configured to enable correct packet routing between the NAP Client and the NEP.

3. The HCEP client service has been started.

- **Means of knowing requirement satisfied**:

  1. The NAP Client can successfully ping the NEP over the network.
  2. The HCEP client is shown as running within the list of services.
  3. A sniffer trace performed during a Health Certificate Enrollment event, shows HTTP packets traveling between the NAP Client and the NEP. These HTTP packets must contain [MS-HCEP] fields and reflect the HRA settings from [MS-GPNAP].
  4. No errors are logged by the HCEP client.

- **Consequences of not satisfying requirement**: The task is unable to receive NAP response data via the [MS-HCEP] protocol.

Requirement: The DHCPN Client is running and has the ability to receive [MS-DHCPN] packets from the DHCP server on the Network Enforcement Point (NEP).

- **Reason for requirement**: The DHCPN client is used to receive response NAP data (including the SoHR) encapsulated within [MS-DHCPN] packets from the NEP.

- **Means of satisfying the requirement**:

  1. The DHCPN client is configured with the DHCP settings from the registry.
  2. The DHCPN client has network access to the DHCP Server:
     1. The network interface of the NAP Client is configured to operate on the local subnet.
     2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NEP is connected.
     3. All network devices between the local subnet and the NEP are configured to allow packet flow between the two entities.
     4. The routing tables in the NAP Client are configured to enable correct packet routing between the NAP Client and the NEP.
  3. The DHCPN client service has been started.

- **Means of knowing requirement satisfied**:

  1. The NAP Client can successfully ping the NEP over the network.
  2. The DHCPN client is shown as running within the list of services.
  3. An attempt to renew the IP address lease on each NIC completes successfully.
  4. A sniffer trace performed during a DHCP lease renewal event, shows DHCP packets traveling between the NAP Client and the NEP. These DHCP packets must contain [MS-DHCPN] fields.
  5. No errors are logged by the DHCPN client.

- **Consequences of not satisfying requirement**: The task is unable to receive NAP response data via the [MS-DHCPN] protocol.

Requirement: The TSGU Client is running and has the ability to receive [MS-TSGU] packets from the TSGU server on the Network Enforcement Point (NEP).

- **Reason for requirement**: The TSGU client is used to receive NAP response data (including the SoHR) encapsulated within [MS-TSGU] packets from the NEP.
- **Means of satisfying the requirement**:

  1. The TSGU client is configured with the RDP settings from the registry.
  2. The TSGU client has network access to the TSGU Server:
     1. The network interface of the NAP Client is configured to operate on the local subnet.
     2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NEP is connected.
     3. All network devices between the local subnet and the NEP are configured to allow packet flow between the two entities.
     4. The routing tables in the NAP Client are configured to enable correct packet routing between the NAP Client and the NEP.
  3. The TSGU client service has been started.

- **Means of knowing requirement satisfied**:

  1. The NAP Client can successfully ping the NEP over the network.
  2. The TSGU client is shown as running within the list of services.
  3. An attempt to connect to a remote desktop via the Terminal Services Gateway completes successfully.
  4. A sniffer trace performed during a RDP connection event, shows TSGU packets traveling between the NAP Client and the NEP. These TSGU packets must contain [MS-TSGU] fields.
  5. No errors are logged by the TSGU client.

- **Consequences of not satisfying requirement**: The task is unable to receive NAP response data via the [MS-TSGU] protocol.

Requirement: The CEAP peer is running and has the ability to receive [MS-CEAP] packets from the EAPE server on the Network Enforcement Point (NEP).

- **Reason for requirement**: The CEAP peer is used to receive NAP response data (including the SoHR) encapsulated within [MS-CEAP] packets from the NEP.
- **Means of satisfying the requirement**:

  1. The CEAP peer is configured with the EAP/PEAP settings from the registry.
  2. The CEAP peer has network access to the EAPE Proxy:
     1. The network interface of the NAP Client is configured to operate on the local subnet.
     2. The physical network path (network devices, Ethernet cables, etc.) between the local subnet and the NEP is connected.

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

4. The routing tables in the NAP Client are configured to enable correct packet routing between the NAP Client and the NEP.

3. The CEAP peer service has been started.

- **Means of knowing requirement satisfied**:

1. The NAP Client can successfully ping the NEP over the network.

2. The EAPE peer is shown as running within the list of services.

3. An attempt to VPN to a computer on the corporate network completes successfully.

4. A sniffer trace performed during a VPN connection event, shows EAPE packets traveling between the NAP Client and the NEP. These EAPE packets must contain [MS-EAPE] fields.

5. No errors are logged by the EAPE peer.

- **Consequences of not satisfying requirement**: The task is unable to receive NAP response data via the [MS-EAPE] protocol.

Requirement: The SoH client is operational and has the ability to process [MS-SOH] packets.

- **Reason for requirement**: The SoH client is used to process the SoHR message and perform remediation, as needed.

- **Means of satisfying the requirement**:

1. The SoH client is configured with the settings described in section 2.5 SoH Settings in [MS-GPNAP].

2. The SoH client service has been started.

3. All required SHA plug-ins are installed and configured.

4. All required SHA plug-ins have initialized and reported to the SoH client.

- **Means of knowing requirement satisfied**:

1. The SoH client is shown as running within the list of services.

2. A sniffer trace performed during a DHCP lease renewal event, shows DHCPN packets traveling between the NAP Client and the NEP containing a [MS-SOH] message.

3. No errors are logged by the SoH client.

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

Requirement: The Authentication client is operational and has the ability to process user credential response.

- **Reason for requirement**: The Authentication client is used to process the authentication response.

- **Means of satisfying the requirement**:

1. The Authentication client is configured from the registry.
2. The Authentication service is started.
3. The credential cache is initialized and is operational.
4. The Authentication service is connected to the credential cache.

- **Means of knowing requirement satisfied**:

1. The Authentication service is shown as running within the list of services.
2. A user on the NAP client is able to login.
3. A sniffer trace performed during a Health Certificate Enrollment event, shows HTTP packets traveling between the NAP Client and the NEP:
    1. The HTTP packets must contain an authentication blob in the Authorization field.
    2. The final server response is "200 OK".
4. No errors are logged by the Authentication client.

- **Consequences of not satisfying requirement**: The task is unable to process the authentication response.

11.2.2 Task Relationships

11.2.2.1 Black-Box Relationship Diagrams

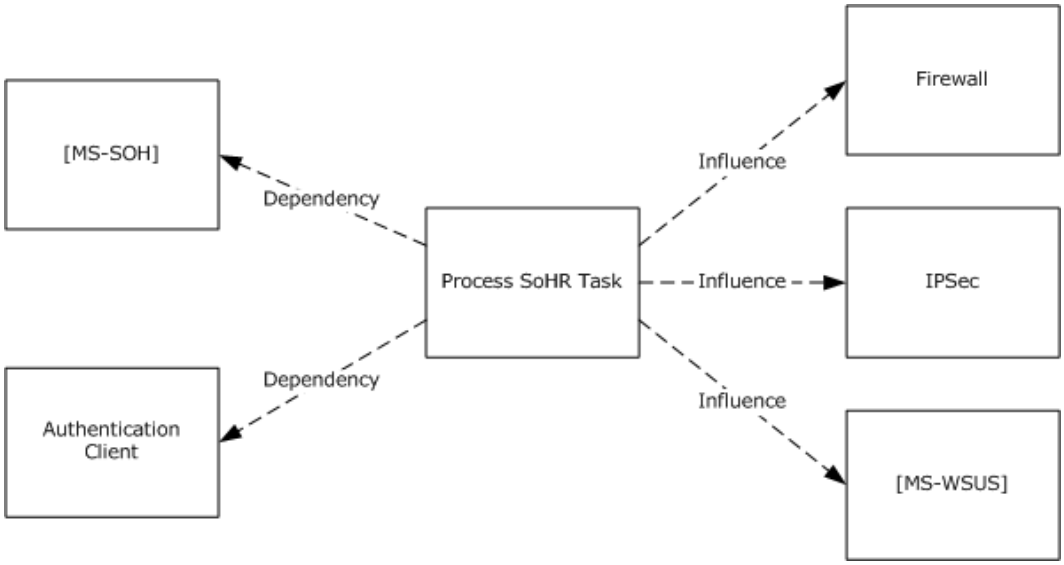

Figure 45: Process SoHR Task black-box relationships

11.2.2.2 Task Dependencies

The Process SoHR Task depends on the MS-SOH client, and the MS-SOH processing rules, to process the SoHR message and perform remediation.The Process SoHR Task has a dependency on

the Authentication Client, which processes the authentication response and determines if another round of authentication is required.

11.2.2.3 Task Influences

The Process SoHR Task influences the firewall, as the firewall rules will be modified if the NAP Client transitions from compliant to non-compliant, or vice versa. The Process SoHR Task influences IPSec, as the IPSec connection will only be completed if the [MS-HCEP] client returns a certificate due to the NAP Client being deemed healthy. The Process SoHR Task influences the [MS-WSUS] client, as a NAP Client that is deemed non-compliant due to missing Windows Updates, will require the [MS-WSUS] client to fetch new updates.

11.2.3 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 NAP Client 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.

11.2.4 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.

11.3 Task Architecture

This section describes the structure of the Process SoHR Task and the interrelationships among its parts.

11.3.1 Task Architectural Constraints

There should be only one instance of the Process SoHR Task on each NAP Client and this instance initializes itself each time it starts. Different instances of this task on different NAP Clients can run independently. There are no constraints among these instances.

11.3.2 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.

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.

Name	Type	Description
SoHResponse	[MS-SOH] section 2.2.6	A SoHR message created by the SoH server.
SoHRespLength	DWORD	Length, in bytes, of the SoHR message.
AuthResponse	Blob	Binary blob containing response from the authentication server.

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 is always run asynchronously and never returns values to the caller.

11.3.5 White-Box Relationships

The white box relationships for the Process SoHR Task are shown in the following figure.

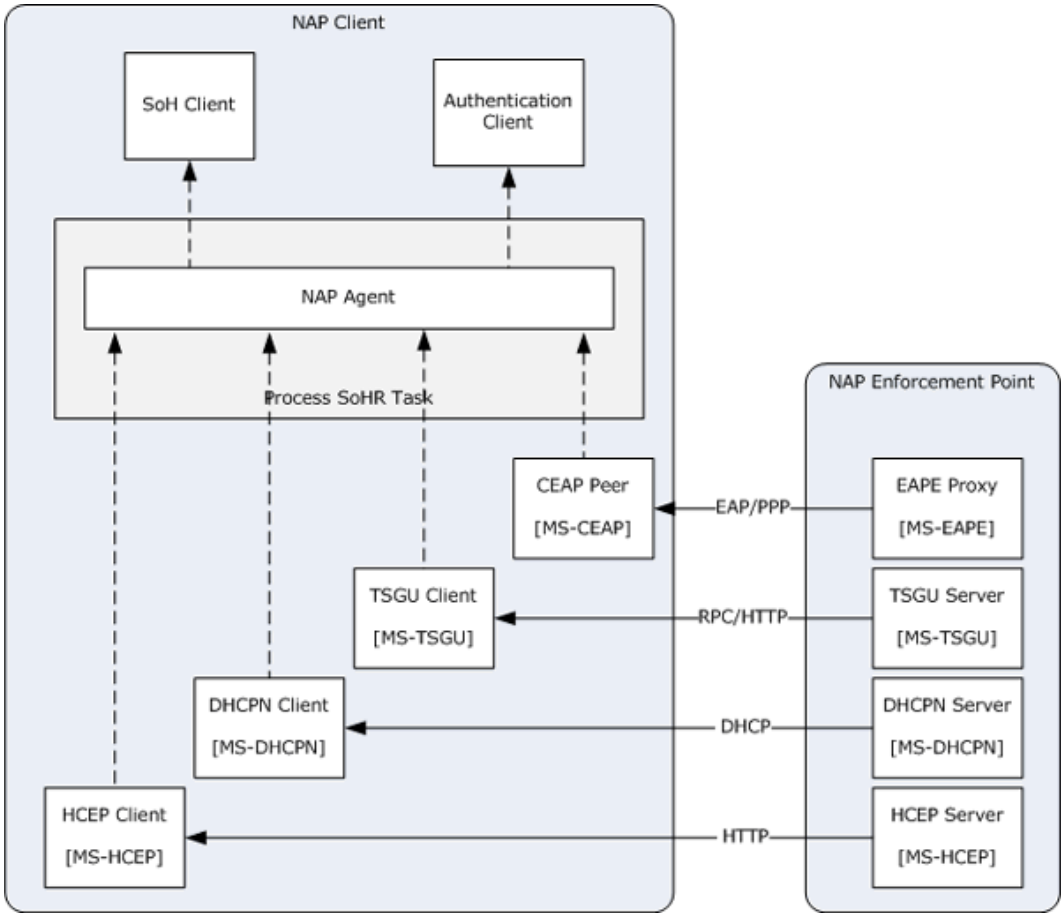

Figure 46: Process SoHR Task white-box relationships

After receiving the SoHR message, the NAP agent and SoH client process the SoHR following the format defined in the Statement of Health for Network Access Protection (NAP) Protocol ([MS-SOH]).

11.3.6 Task Events

11.3.6.1 Task Timers

None.

11.3.6.2 Task Non-Timer Events

This task does not use or respond to any additional non-timer events.

11.3.7 Task Architecture and Communication

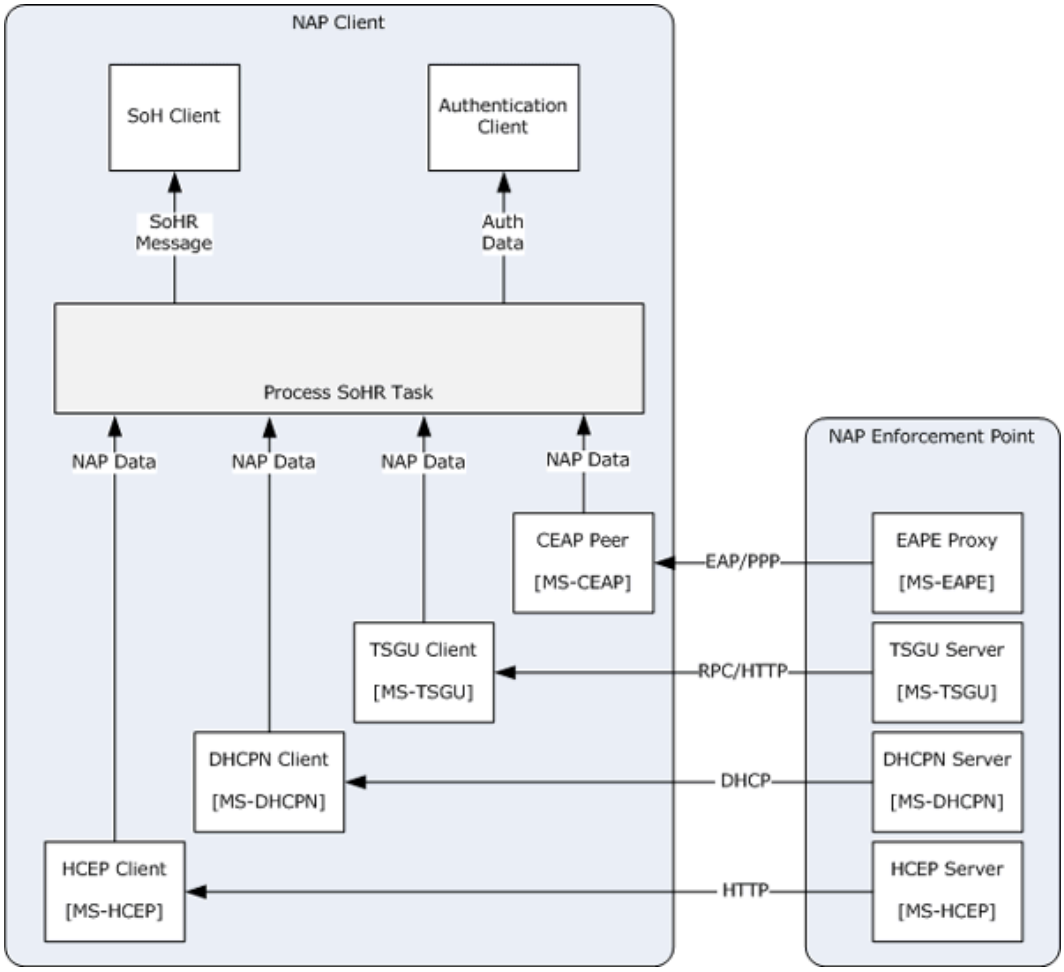

Figure 47: Process SoHR Task architecture and communication overview

11.3.8 Task Processing Rules

The following describes the operational flow of the Process SoHR Task:

1. One of the following occurs by a NAP transport protocol client to trigger the task:
   - The DHCPN Client invokes the task as described in [MS-DHCPN].
   - The HCEP Client invokes the task as described in [MS-HCEP].
   - The TSGU Client invokes the task as described in [MS-TSGU].
   - The CEAP Peer invokes the task as described in [MS-CEAP].
2. The NAP agent calls the ProcessSoHResponse abstract interface with the SoHResponse parameter to process the SoHR message and start remediation.

3. The NAP agent processes the authentication response by sending the AuthResponse parameter to the Authentication Client.

4. If the Authentication Client responds that a new round of authentication is required (due to a challenge response or authentication failure), the NAP Agent sends an NAP re-authentication system event.

11.3.9 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.

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

The NAP agent service is started and initialized correctly on the NAP Client.

For the complete list of task preconditions and assumptions, see section 11.2.3.

11.4.2 Task Initialization of External Entities

None.

11.4.3 Task Event Details

11.4.3.1 Task Timer Details

None.

11.4.3.2 Task Non-Timer Event Details

None.

11.4.4 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.

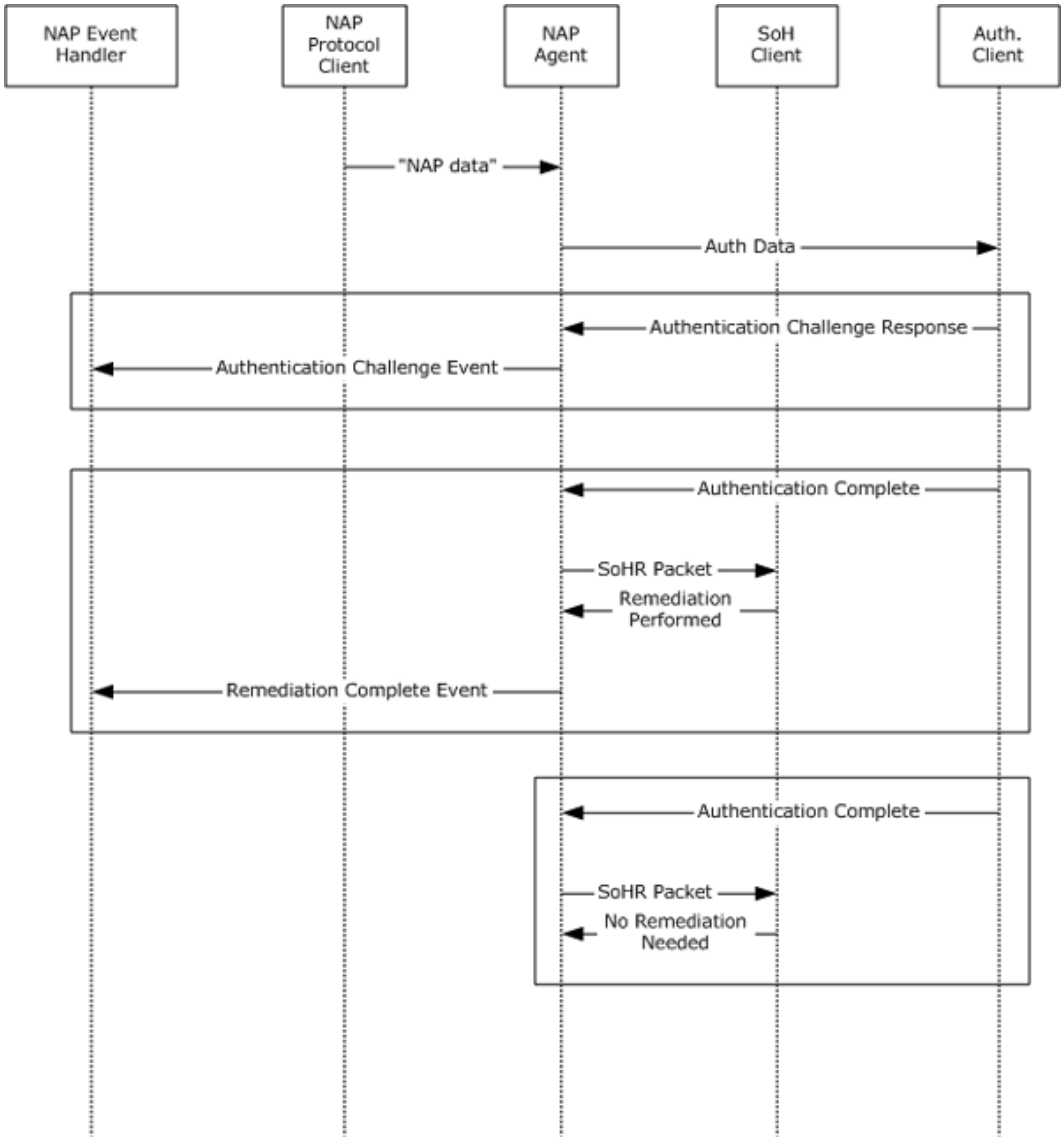

Figure 48: Sequence diagram for the main success scenario of the Process SoHR Task

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 11.4.5.

11.4.5 Task Processing Rule Details

The following describes the operational details of the Process SoHR Task:

1. One of the following occurs by a NAP transport protocol client to trigger the task:

   - The DHCPN Client invokes the task as described in section 3.1.7.2 DhcpClientProcessSoHR in [MS-DHCPN].

- The HCEP Client invokes the task as described in section 3.1.5.2 Processing an HCEP Response in [MS-HCEP].
- The TSGU Client invokes the task as described in section 3.2.4 Message Processing Events and Sequencing Rule in [MS-TSGU].
- The CEAP Peer invokes the task as described in section x.x.x in [MS-CEAP].

2. The NAP agent calls the ProcessSoHResponse abstract interface (section 3.2.7.2 ProcessSoHResponse in [MS-SOH]) with the SoHResponse parameter to process the SoHR message and start remediation.
3. The NAP agent processes the authentication response by sending the AuthResponse parameter to the Authentication Client.
4. If the Authentication Client responds that a new round of authentication is required (due to a challenge response or authentication failure), the NAP Agent sends an NAP re-authentication system event.

11.5 Task Security

There are no task-specific security considerations. For additional information about security considerations, see section 12, as well as the Security sections of the referenced protocol Technical Documents.

12 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.

This is illustrated in the following figure, where the client is shown communicating with the PDP for network access.

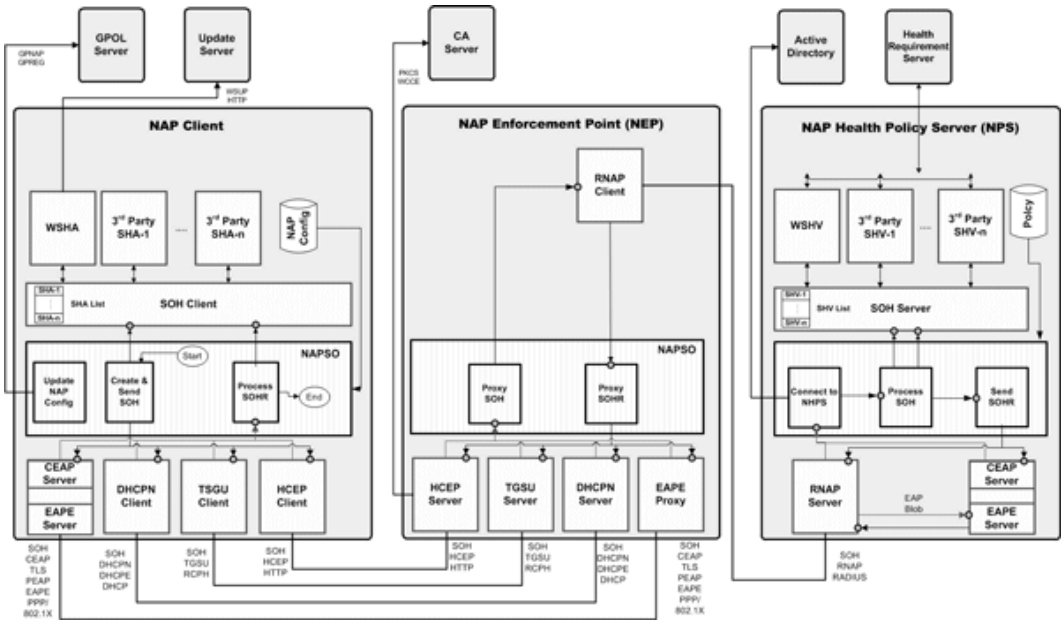

Figure 49: NAP Abstract Task Architecture Overview

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 [MS-SOH] or its contents sent to the NAP health policy server (NPS). The NPS blindly trusts the SoH messages received on the NEP channel and has no means to verify the integrity of the SoH message.

13 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

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.

14 Change Tracking

This section identifies changes that were made to the [MS-NAPSO] protocol document between the May 2011 and June 2011 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

The revision class **New** means that a new document is being released.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.
- An extensive rewrite, addition, or deletion of major portions of content.
- The removal of a document from the documentation set.
- Changes made for template compliance.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **Editorial** means that the language and formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical or language changes were introduced. The technical content of the document is identical to the last released version, but minor editorial and formatting changes, as well as updates to the header and footer information, and to the revision summary, may have been made.

Major and minor changes can be described further using the following change types:

- New content added.
- Content updated.
- Content removed.
- New product behavior note added.
- Product behavior note updated.
- Product behavior note removed.
- New protocol syntax added.
- Protocol syntax updated.
- Protocol syntax removed.
- New content added due to protocol revision.
- Content updated due to protocol revision.
- Content removed due to protocol revision.
- New protocol syntax added due to protocol revision.

- Protocol syntax updated due to protocol revision.
- Protocol syntax removed due to protocol revision.
- New content added for template compliance.
- Content updated for template compliance.
- Content removed for template compliance.
- Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated.**

Some important terms used in the change type descriptions are defined as follows:

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.
- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

The changes made to this document are listed in the following table. For more information, please contact protocol@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
1.2 References	Added explanatory statement regarding the removal of the publishing year from Microsoft Open Specification document references.	N	Content updated.
Global	Revised document for technical clarity and comprehensiveness.	Y	Content updated.

15 Index

D

R

S

T

U

V

W