[MS-OXWSAUTID]:
Authentication Identification Extension

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.** This documentation is 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's Open Specification Promise (available here: http://www.microsoft.com/interop/osp) or the Community Promise (available here: http://www.microsoft.com/interop/cp/default.mspx). 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.

Revision Summary

Date	Revision History	Revision Class	Comments
02/10/2010	1.0.0	Major	Initial Availability.
05/05/2010	1.1.0	Minor	Updated the technical content.
08/04/2010	1.1.0	No change	No changes to the meaning, language, or formatting of the technical content.
11/03/2010	1.1.0	No change	No changes to the meaning, language, or formatting of the technical content.

Contents

1 Introduction

This document specifies the Authentication Identification extension, which defines the authentication schemes that are supported by a client.

1.1 Glossary

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

header
Hypertext Transfer Protocol (HTTP)
NTLM
Transport Layer Security (TLS)

The following terms are specific to this document:

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as described in [RFC2119]. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 References

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-N2HT] Microsoft Corporation, "Negotiate and Nego2 HTTP Authentication Protocol Specification", December 2008, http://msdn.microsoft.com/en-us/library/dd303576(PROT.10).aspx

[MS-NTHT] Microsoft Corporation, "NTLM Over HTTP Protocol Specification", July 2006, http://msdn.microsoft.com/en-us/library/cc237488(PROT.10).aspx

[MS-OXDSCLI] Microsoft Corporation, "Autodiscover Publishing and Lookup Protocol Specification", April 2008.

[MS-OXWAVLS] Microsoft Corporation, "Availability Web Service Protocol Specification", April 2008.

[MS-OXWCONFIG] Microsoft Corporation, "Web Service Configuration Protocol Specification", April 2009.

[MS-OXWMT] Microsoft Corporation, "Mail Tips Web Service Extensions", April 2009.

[MS-OXWOAB] Microsoft Corporation, "Offline Address Book (OAB) Retrieval File Format", April 2008.

[MS-OXWOOF] Microsoft Corporation, "Out of Office (OOF) Web Service Protocol Specification", April 2008.

[MS-OXWSMSHR] Microsoft Corporation, "Folder Sharing Web Service Protocol Specification", November 2009.

[MS-OXWSMTRK] Microsoft Corporation, "Message Tracking Web Service Protocol Specification", July 2009.

[MS-OXWUMS] Microsoft Corporation, "Voice Mail Settings Web Service Protocol Specification", April 2008.

[MS-RPCH] Microsoft Corporation, "Remote Procedure Call over HTTP Protocol Specification", July 2006, http://msdn.microsoft.com/en-us/library/cc243950.aspx

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

[RFC2246] Dierks, T., and Allen, C., "The TLS Protocol Version 1.0", RFC 2246, January 1999, http://www.ietf.org/rfc/rfc2246.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

[RFC4121] Zhu, L., Jaganathan, K., and Hartman, S., "The Kerberos Version 5 Generic Security Service Application Program Interface (GSS-API) Mechanism: Version 2", RFC 4121, July 2005, http://www.ietf.org/rfc/rfc4121.txt

[RFC5234] Crocker, D., Ed., and Overell, P., "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008, http://www.ietf.org/rfc/rfc5234.txt

1.2.2 Informative References

[MS-OXGLOS] Microsoft Corporation, "Exchange Server Protocols Master Glossary", April 2008.

1.3 Overview

In most implementations, clients that support the Negotiate and Nego2 HTTP Authentication protocol, as specified in [MS-N2HT], support either the Kerberos or **NTLM** authentication providers. In cases where servers do support the use of custom Negotiate-capable authentication providers without fallback to Kerberos or NTLM, an additional client capability protocol must be indicated to the server.

The **X-Nego-Capability header** is used to specify the authentication providers that are supported by a client within the Negotiate or Nego2 HTTP Authentication protocol. The server compares the authentication providers received in the **X-Nego-Capability** header to Negotiate-capable authentication providers that are available to the server. The server sends a response to the client that specifies the Negotiate or Nego2 HTTP Authentication protocol in **WWW-Authenticate** response headers if there is at least one capable authentication service provider that is shared by both the server and client. The client uses the values from the **WWW-Authenticate** headers to determine which authentication scheme to use.

1.4 Relationship to Other Protocols

The **X-Nego-Capability** header is an extension to the Hypertext Transfer Protocol (HTTP), as specified in [RFC2616].

1.5 Prerequisites/Preconditions

The successful use of the **X-Nego-Capability** header is based on the assumption that both the client and the server can authenticate users and that the client and the server can identify the available authentication services.

1.6 Applicability Statement

This extension is applicable to clients and server applications that dynamically determine the best authentication scheme that is shared by both the client and server. The **X-Nego-Capability** header is applicable to environments where the client and server support the Negotiate and Nego2 HTTP Authentication protocol, as specified in [MS-N2HT].

This extension is used to identify client authentication providers when the following are used:

- Autodiscover Publishing and Lookup protocol, as specified in [MS-OXDSCLI]
- Availability Web Service protocol, as specified in [MS-OXWAVLS]
- Web Service Configuration protocol, as specified in [MS-OXWCONFIG]
- Mail Tips Web Service extensions, as specified in [MS-OXWMT]
- Offline Address Book (OAB) Retrieval file format, as specified in [MS-OXWOAB]
- Out of Office (OOF) Web Service protocol, as specified in [MS-OXWOOF]
- Folder Sharing Web Service protocol, as specified in [MS-OXWSMSHR]
- Message Tracking Web Service protocol, as specified in [MS-OXWSMTRK]
- Voice Mail Settings Web Service protocol, as specified in [MS-OXWUMS]

This protocol is not applicable to the Remote Procedure Call over HTTP protocol, as specified in [MS-RPCH].

1.7 Versioning and Capability Negotiation

Versioning and capability negotiation is handled at the **HTTP** layer of the protocol stack, as specified in [RFC2616]. Values for capability negotiation of the actual supported authentication schemes come from system registry entries on both the client and server.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignment

None.

2 Messages

2.1 Transport

The X-Nego-Capability header is carried in HTTP (as specified in [RFC2616]) requests.

2.2 Message Syntax

Negotiation capability information is indicated in HTTPrequests by the X-Nego-Capability header.

2.2.1 X-Nego-Capability Header

The **X-Nego-Capability** header specifies the Negotiate-capable authentication schemes that are supported by a client. The **X-Nego-Capability** header MUST be sent in a request to the server before an **Authorization** header (as specified in [RFC2616] section 14.8) is sent so that the server can indicate to the client which authentication schemes it can use. The server MUST return **WWW-Authenticate** headers that identify each authentication scheme the server and the client have in common.

The format for the **X-Nego-Capability** HTTP request header in ABNF (as specified in [RFC5234]) is as follows:

```
X-Nego-Capability = "X-Nego-Capability" ":" SP Nego-scheme-range

Nego-scheme-range = 1*Negotiable-security-package ( "," ( SP ) )

Negotiable-security-package = 1*CHAR
```

Clients SHOULD NOT include the **X-Nego-Capability** header on requests that are not protected by **Transport Layer Security (TLS)** (as specified in [RFC2246]), because this can allow a man-in-the-middle to downgrade the authentication scheme that is exposed to clients<1>. Correspondingly, servers can accept **X-Nego-Capability** headers on requests that are not protected by TLS (as specified in [RFC2246]).

3 Protocol Details

3.1 Common Details

3.1.1 Abstract Data Model

The abstract data model for negotiate capability has one abstract data element that lists the authentication schemes that are supported by the client. Additionally, the intersection of the supported client and server authentication schemes results in a list of **WWW-Authentication** headers in the response that is received by the client.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Higher-Layer Triggered Events

None.

3.1.5 Message Processing Events and Sequencing Rules

The **X-Nego-Capability** header is only sent from the client.

3.1.5.1 Client Message Processing Events and Sequencing Rules

The **X-Nego-Capability** header MUST be populated with tokens that identify the Negotiate-capable authentication schemes that are supported by the client. The format for the **X-Nego-Capability** header is specified in section 2.2.1. The client's initial request SHOULD NOT attempt authorization. The initial request to obtain shared authentication schemes MUST include the **X-Nego-Capability** header.

3.1.5.2 Server Message Processing Events and Sequencing Rules

The server SHOULD respond to a client request that contains the **X-Nego-Capability** header with a list of tokens that represent the common HTTP authentication schemes that are shared by the client and server if the credentials sent in the initial request prompt an HTTP 401 response. The authentication scheme tokens are contained in **WWW-Authenticate** headers that are sent by the server. The server SHOULD include the **WWW-Authenticate: Nego2** or the **WWW-Authenticate: Negotiate** response headers when the **X-Nego-Capability** header that is received from the client includes at least one Negotiate-capable authentication provider that is mutually supported by the server, and the response is an HTTP 401 error (as specified in [RFC2616]).<2>

3.1.6 Timer Events

None.

3.1.7 Other Local Events

None.

4 Protocol Examples

An HTTP 1.1 client requests a resource from a server by sending an HTTP **GET** request, as shown in the following example.

```
GET /autodiscover/autodiscover.xml HTTP/1.1

Host: autodiscover.contoso.com

Content-Type: text/xml

User-Agent: Microsoft Office/14.0 (Windows NT 6.1; Microsoft Office Outlook 14.0.1234

X-Nego-Capability: Nego2, Negotiate, Kerberos, NTLM
```

In this request, the **X-Nego-Capability** header identifies four supported authentication schemes:

```
Nego2, Negotiate, Kerberos, NTLM
```

The server is expected to respond to this request with an HTTP 401 with a set of **WWW-Authenticate** headers that identify the shared authentication schemes. The client responds with the best available authentication scheme by using the **Authorization** header.

5 Security

5.1 Security Considerations for Implementers

Clients MUST NOT emit an **X-Nego-Capability** header over a non-TLS-protected HTTP connection.

5.2 Index of Security Parameters

The following table lists the authentication mechanisms that are supported with the **X-Nego-Capability** header.

Security Parameter	Reference
NTLM	[MS-NTHT]
Kerberos	[RFC4121]
Negotiate	[MS-N2HT]
Nego2	[MS-N2HT]

6 Appendix B: Product Behavior

The information in this specification is applicable to the following Microsoft products:

- Microsoft® Exchange Server 2010
- Microsoft® Outlook® 2010

Exceptions, if any, are noted below. If a service pack number appears with the product version, behavior changed in that service pack. 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 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 product does not follow the prescription.

<1> Section 2.2.1: Outlook 2010 does not send the **X-Nego-Capability** header unless the request is protected by Transport Layer Security (TLS) (as described in [RFC2246]).

<2> Section 3.1.5.2: Only the Datacenter Edition of Exchange 2010 supports this behavior.

7 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

8 Index