[MS-OAUTH2EX]:

OAuth 2.0 Authentication Protocol Extensions

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 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. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.
- **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.

Preliminary Documentation. This Open Specification provides documentation for past and current releases and/or for the pre-release version of this technology. This Open Specification is final documentation for past or current releases as specifically noted in the document, as applicable; it is preliminary documentation for the pre-release versions. Microsoft will release final documentation in connection with the commercial release of the updated or new version of this technology. As the documentation may change between this preliminary version and the final version of this technology, there are risks in relying on preliminary documentation. To the extent that you incur additional development obligations or any other costs as a result of relying on this preliminary documentation, you do so at your own risk.

Revision Summary

Date	Revision History	Revision Class	Comments
1/20/2012	0.1	New	Released new document.
4/11/2012	0.1	No Change	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	0.1	No Change	No changes to the meaning, language, or formatting of the technical content.
9/12/2012	0.1	No Change	No changes to the meaning, language, or formatting of the technical content.
10/8/2012	1.0	Major	Significantly changed the technical content.
2/11/2013	1.0	No Change	No changes to the meaning, language, or formatting of the technical content.
7/30/2013	1.0	No Change	No changes to the meaning, language, or formatting of the technical content.
11/18/2013	1.0	No Change	No changes to the meaning, language, or formatting of the technical content.
2/10/2014	1.0	No Change	No changes to the meaning, language, or formatting of the technical content.
4/30/2014	1.1	Minor	Clarified the meaning of the technical content.
7/31/2014	1.1	No Change	No changes to the meaning, language, or formatting of the technical content.
10/30/2014	1.1	No Change	No changes to the meaning, language, or formatting of the technical content.
3/16/2015	2.0	Major	Significantly changed the technical content.

Table of Contents

1 Introduction

OAuth 2.0 Authentication Protocol Extensions describes extensions to the OAuth 2.0 Authentication Protocol. These extensions consist of additional parameters in the request **URI** and the **JSON** objects returned in the **HTTP** response body.

Sections 1.8, 2, and 3 of this specification are normative and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in [RFC2119]. Sections 1.5 and 1.9 are also normative but do not contain those terms. All other sections and examples in this specification are informative.

1.1 Glossary

The following terms are specific to this document:

Coordinated Universal Time (UTC): A high-precision atomic time standard that approximately tracks Universal Time (UT). It is the basis for legal, civil time all over the Earth. Time zones around the world are expressed as positive and negative offsets from UTC. In this role, it is also referred to as Zulu time (Z) and Greenwich Mean Time (GMT). In these specifications, all references to UTC refer to the time at UTC–0 (or GMT).

Hypertext Transfer Protocol (HTTP): An application-level protocol for distributed, collaborative, hypermedia information systems (text, graphic images, sound, video, and other multimedia files) on the World Wide Web.

Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS): An extension of **HTTP** that securely encrypts and decrypts webpage requests.

JavaScript Object Notation (JSON): A text-based, data interchange format that is used to transmit structured data, typically in Asynchronous JavaScript + XML (AJAX) web applications, as described in [RFC4627]. The JSON format is based on the structure of ECMAScript (Jscript, JavaScript) objects.

principal: An authenticated entity that initiates a message or channel in a distributed system.

realm: An administrative boundary that uses one set of authentication servers to manage and deploy a single set of unique identifiers. A realm is a unique logon space.

Representational State Transfer (REST): A class of web services that is used to transfer domain-specific data by using **HTTP**, without additional messaging layers or session tracking, and returns textual data, such as XML.

security token service (STS): A web service that issues claims (2) and packages them in encrypted security tokens.

tenant: A protocol client or protocol server that accesses a partition in a shared service database.

token: A word in an item or a search query that translates into a meaningful word or number in written text. A token is the smallest textual unit that can be matched in a search query. Examples include "cat", "AB14", or "42".

Uniform Resource Identifier (URI): A string that identifies a resource. The URI is an addressing mechanism defined in Internet Engineering Task Force (IETF) Uniform Resource Identifier (URI): Generic Syntax [RFC3986].

Uniform Resource Locator (URL): A string of characters in a standardized format that identifies a document or resource on the World Wide Web. The format is as specified in [RFC1738].

X.509: An ITU-T standard for public key infrastructure subsequently adapted by the IETF, as specified in [RFC3280].

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

1.2 References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the Errata.

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.

[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

1.2.2 Informative References

None.

1.3 Overview

This document describes extensions to the OAuth 2.0 Authentication Protocol. These extensions consist of additional parameters in the request URI and the JSON objects returned in the HTTP response body. These extensions provide additional functionality, such as finer granularity of token expiry periods, which is incremental over what is provided by the base specification.

1.4 Relationship to Other Protocols

1.5 Prerequisites/Preconditions

Applications using these extensions must have their **principal** identifiers registered as members of the **realm** managed by the **security token service (STS)** and are able to authenticate to it by exchanging secrets (public key material carried in **X.509** certificates). The applications must also trust tokens issued by the STS. These preconditions are established during deployment of new **tenants**.

1.6 Applicability Statement

These extensions are used in OAuth 2.0 Authentication requests for obtaining an access **token** for use with **REST** bases service requests.

1.7 Versioning and Capability Negotiation

None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

No specific parameters are passed to the transport layer. The transport layer, **HTTPS**, is used for securing messages and tokens sent over the wire.

2.2 Message Syntax

Parameters in these extensions are either transmitted in the request body in **URL**-encoded form or in the response as part of a JSON object. Tokens returned in conjunction with use of this protocol conform to the JWT specification.

XML based serialization is not used with these extensions or underlying protocols.

3 Protocol Details

3.1 Common Details

This specification defines additional parameters for use with OAuth 2.0 Authentication Protocol messages. These parameters convey additional semantics related to the access token being requested and about the access token returned as part of these messages. This specification does not define any additional messages over the base specification.

3.1.1 Abstract Data Model

The following parameters are used in conjunction with the OAuth 2.0 Authentication Protocol.

Mandatory parameters when using the extensions defined in this specification.
Parameter name: resource
Parameter usage location: Request body, URL encoded
Type: string, URI value expected
Semantics: This parameter indicates the target resource that the caller wants to talk to. Value is expected to be URI format as defined in RFC 3986
Parameter name: not_before
Parameter usage location: Response body in JSON object
Type: string, **UTC** value expected
Semantics: Token returned in the OAuth parameter access_token as part of this response is not valid before the time specified in this parameter.
Parameter name: expires_on
Parameter usage location: Response body in JSON object
Type: string, UTC value expected
Semantics: Token returned in the OAuth parameter access_token as part of this response will no longer be valid at the time specified in this parameter.
Optional parameters when using the extensions defined in this specification.
Parameter name: realm
Parameter usage location: Request body, URL encoded or Response body in JSON object
Type: string
Semantics: Indicates the realm that the caller is part of.
Parameter name: created_on
Parameter usage location: Request body, URL-encoded or Response body in JSON object
Type: string, UTC value expected
Semantics: Token returned in the OAuth parameter access_token as part of this was created at the time specified in this parameter.
Parameter name: expires_in
Parameter usage location: Response body in JSON object
Type: string, int value expected.
Semantics: Lifetime of the token returned in the OAuth parameter access_token as part of this response specified in seconds.

3.1.2 Timers

None.

3.1.3 Initialization

Connections made using the underlying OAuth 2.0 Authentication Protocol are what initiate use of the extensions defined in this specification.

3.1.4 Higher-Layer Triggered Events

None.

3.1.5 Message Processing Events and Sequencing Rules

The token request contains the elements indicated in section 3.1.1.

3.1.5.1 Protected Resource

This resource indicates the protected resource that the user is trying to get access to.

3.1.5.1.1 Request Access

For gaining access to a resource, an access token with the elements mentioned in section 3.1.1 need to be presented.

3.1.5.1.1.1 Request Body

Refer to the section 4.1.

3.1.5.1.1.2 Response Body

Refer to the section 4.2.

3.1.5.1.1.3 Processing Details

None.

3.1.6 Timer Events

None.

3.1.7 Other Local Events

None.

4 Protocol Examples

4.1 Example request for token to access a resource

The following is an example request for token to access a resource:

```
POST https://example.com/tokens/OAuth/2 HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Host: example.com
Content-Length: 660
Expect: 100-continue
grant_type=http%3a%2f%2foauth.net%2fgrant_type%2fjwt%2f1.0%2fbearer&assertion=eyJ0eXAiOiJKV1Q
iLCJhbGciOiJSUzI1NiIsIng1dCI6IkM5Qi1lUHpvYnVtejVoM3lZb3FJbjhHWFFwNCJ9.eyJhdWQiOiIwMDAwMDAwMS0
wMDAwLTAwMDAtYzAwMC0wMDAwMDAwMDAwMDAvc3RzLWludC1zbjEtMDAzLmFjY2Vzc2NvbnRyb2wuYWFkaW50LndpbmRv
d3MtaW50Lm5ldEBtZ29vZG5lci5jb20iLCJpc3MiOiJDTj1BQ1MyQ2xpZW50Q2VydGlmaWNhdGVAbWdvb2RuZXIuY29tI
iwibmJmIjoiMTMyMTY1NTMzOSIsImV4cCI6IjEzMjE2NTg5MzkifQ.XcO8kUte4PKjPwpmXiTBKo4bDAAVJW1eB9d69nM
paumjl9vqw-
N0wLIDN4M6btn_G1BYgcPh0l3hl7Lsxs7E2SfssIegY_KuHZJIdb3pyc0KNzfcoolxIZRFQDK3zmYKICBEQCJ9nHBLcL4
Y8AXkZVddYjGo-104M4rg0AcXMZU&resource=example.com/resouce
```

4.2 Response with token to access requested resource

The following is an example of a response with token to access the requested resource:

```
HTTP/1.1 200 OK
Cache-Control: public, no-store, max-age=0
Pragma: no-cache
Content-Type: application/json; charset=utf-8
Expires: Fri, 18 Nov 2011 22:28:59 GMT
Last-Modified: Fri, 18 Nov 2011 22:28:59 GMT
Vary: *
Server: Microsoft-IIS/7.5
Set-Cookie: ASP.NET_SessionId=5v3qtoiaxr3m21gdbbai0cxu; path=/; HttpOnly
X-AspNetMvc-Version: 2.0
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
X-Content-Type-Options: nosniff
Date: Fri, 18 Nov 2011 22:28:59 GMT
Content-Length: 1146
{
"access_token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng1dCI6Ilhxcm5GRWZzUzU1X3ZNQnBIdkYwcFRuc
WVhTSJ9.eyJhdWQiOiJtaWNyb3NvZnQuZXhjaGFuZ2UvbG9jYWxob3N0OjQ1NzA0QEVENjhGQTczLTNCRTYtNDYxMi1BO
TQ0LTI3ODY1OUVEOTAwNCIsImlzcyI6IjAwMDAwMDAxLTAwMDAtMDAwMC1jMDAwLTAwMDAwMDAwMDAwMEBFRDY4RkE3My
0zQkU2LTQ2MTItQTk0NC0yNzg2NTlFRDkwMDQiLCJuYmYiOjEzMjE2NTUzNDAsImV4cCI6MTMyMTY1ODk0MCwibmFtZWl
kIjoiQ049QUNTMkNsaWVudENlcnRpZmljYXRlQEVENjhGQTczLTNCRTYtNDYxMi1BOTQ0LTI3ODY1OUVEOTAwNCIsImlk
ZW50aXR5cHJvdmlkZXIiOiIwMDAwMDAwMS0wMDAwLTAwMDAtYzAwMC0wMDAwMDAwMDAwMDBARUQ2OEZBNzMtM0JFNi00N
jEyLUE5NDQtMjc4NjU5RUQ5MDA0In0.j0KHTj0VGKN35-
NOLnN14MWOrUzKcWKXXCwOzj6LFZNrBgEBzIUvSksZ9Av4nxB_wYzflp9QEFDOMUkSkzlBt4t4eUZxbb78RSMHJLnetnL
CUgL7POuE4e_-x3kq_LNnYc_-VQ9MLw4KIblwJrVD4LyMNVlo-5VGIybKMGM3fVvtrazgvgFC-
7MmQbmjJ8m249J8InL7Qt2fbyBuPRA0Ey9LhLDqZ7t_LWUGc_ufNxP7T0crzSun6MhUM332-
lW7OFdq6HXx1oD6qDc8te5cQ9IloF6WYonDTiCBCxfiTt-ouEHskOZDwZ781wBTn9TJujMYo1vjurXUVqiBNZCQ-A",
"token_type":"http://oauth.net/grant_type/jwt/1.0/bearer",
"expires_in":"3599",
"scope":"example.com/resource@guid",
"not_before":"1321655340",
"expires_on":"1321658940"
}
```

5 Security

5.1 Security Considerations for Implementers

Security considerations described in underlying specifications and in profiles referencing this extension specification should be considered when using these extensions.

5.2 Index of Security Parameters

None.

6 Appendix A: Full JSON Schema

For ease of implementation, the following is the full JSON schema for this protocol.

```
"not_before" : {
"type" : "string",
"format" : "date-time"
},
"expires_on" : {
"type" : "string",
"format" : "date-time"
},
"realm" : {
"type" : "string",
"format" : "uri"
},
"created_on" : {
"type" : "string",
"format" : "date-time"
},
"expires_in" : {
"type" : "integer"
}
```

7 Appendix B: 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.

- Microsoft Skype for Business (formerly Lync 2013)
- Microsoft Lync Server 2013
- Microsoft Exchange Server 2013
- Microsoft SharePoint Server 2013
- Microsoft SharePoint Foundation 2013
- Skype for Business
- Skype for Business Server

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.

8 Change Tracking

This section identifies changes that were made to this document since the last release. 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.
- The removal of a document from the documentation set.

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 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 changes were introduced. Minor editorial and formatting changes may have been made, but the technical content of the document is identical to the last released version.

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.
- 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 dochelp@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
7 Appendix B: Product Behavior	Updated list of supported products.	Y	Content updated due to protocol revision.

9 Index