Service Provider Request Initiation Protocol and Profile Version 1.0

Committee Draft 01
4 May 2010

Specification URIs:

This Version:

http://docs.oasis-open.org/security/saml/Post2.0/sstc-request-initiation-cd-01.html

http://docs.oasis-open.org/security/saml/Post2.0/sstc-request-initiation-cd-01.odt (Authoritative)

http://docs.oasis-open.org/security/saml/Post2.0/sstc-request-initiation-cd-01.pdf

Previous Version:

None

Latest Version:

http://docs.oasis-open.org/security/saml/Post2.0/sstc-request-initiation.html

http://docs.oasis-open.org/security/saml/Post2.0/sstc-request-initiation.odt (Authoritative)

http://docs.oasis-open.org/security/saml/Post2.0/sstc-request-initiation.pdf

Technical Committee:

OASIS Security Services TC

Chair:

Thomas Hardjono, M.I.T.
Nate Kingenstein, Internet2

Editor:

Scott Cantor, Internet2

Related Work:

This specification composes with the Identity Provider Discovery Service Protocol and Profile [IdPDisco], and with multiple standards for browser-based Single Sign-On, such as SAML 2.0 and WS-Federation [WS-Fed].

Declared XML Namespace(s):

urn:oasis:names:tc:SAML:profiles:SSO:request-init

Abstract:

Defines a generic browser-based protocol by which a request can be made to a service provider to initiate a protocol-specific request for authentication, and to ask that particular options be used when making such a request.

Status

This document was last revised or approved by the SSTC on the above date. The level of approval is also listed above. Check the current location noted above for possible later revisions of this document. This document is updated periodically on no particular schedule.

TC members should send comments on this specification to the TC's email list. Others should send comments to the TC by using the "Send A Comment" button on the TC's web page at http://www.oasis-open.org/committees/security.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the IPR section of the TC web page (http://www.oasis-open.org/committees/security/ipr.php).

The non-normative errata page for this specification is located at http://www.oasis-open.org/committees/security.

Notices

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The name "OASIS" is a trademark of OASIS, the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see http://www.oasis-open.org/who/trademark.php for above guidance.

Table of Contents

1 Introduction

Modern standards for browser-based Single Sign-On (SSO) typically include the ability to initiate the authentication process from either the identity provider (IdP) or service provider (SP) participating in the exchange. However, the standards typically lack a defined mechanism for asking either end to actually initiate the process, relying on proprietary interfaces, or on the user agent accessing a protected resource at the service provider.

IdP-initiated SSO assumes a variety of information is known at the time of a request, including the identity provider itself and its location, protocol features and binding/profile details to apply, how to express the desired resource to access, etc. In general, it suffers by leaving the service provider "out of the loop" in formulating the request and applying its own decision-making in doing so.

On the other hand, SP-initiated SSO suffers from a lack of standardization, particularly when support for "deep-linking", or unauthenticated access to resources within a protected system, is lacking. Many complex deployments are unable to fully support direct access in that fashion, and require special conventions or work-arounds that are often propagated to links constructed outside of the affected site, creating brittle links and maintenance challenges.

A standard protocol for invoking the SSO functionality available at a service provider in an abstracted, protocol-neutral fashion solves both problems.

1.1 Notation

This specification uses normative text.

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this specification are to be interpreted as described in [RFC2119]:

> …they MUST only be used where it is actually required for interoperation or to limit behavior which has potential for causing harm (e.g., limiting retransmissions)…

These keywords are thus capitalized when used to unambiguously specify requirements over protocol and application features and behavior that affect the interoperability and security of implementations. When these words are not capitalized, they are meant in their natural-language sense.

```
Listings of XML schemas appear like this.
```

```
Example code listings appear like this.
```

Conventional XML namespace prefixes are used throughout the listings in this specification to stand for their respective namespaces as follows, whether or not a namespace declaration is present in the example:

Prefix	XML Namespace	Comments
`saml:`	urn:oasis:names:tc:SAML:2.0:assertion	This is the SAML V2.0 assertion namespace [SAML2Core].
`samlp:`	urn:oasis:names:tc:SAML:2.0:protocol	This is the SAML V2.0 protocol namespace [SAML2Core].
`md:`	urn:oasis:names:tc:SAML:2.0:metadata	This is the SAML V2.0 metadata namespace .
`init:`	urn:oasis:names:tc:SAML:profiles:SSO:request-init	This is the SAML V2.0 metadata extension namespace defined by this document and its accompanying schema [ReqInit-XSD].

This specification uses the following typographical conventions in text: `<SAMLElement>`, `<ns:ForeignElement>`, `Attribute`, **Datatype**, `OtherCode`.

1.2 Normative References

[ReqInit-XSD] OASIS Committee Draft, *Metadata Extension Schema for Service Provider Request Initiation Protocol and Profile Version 1.0*, May 2010. http://docs.oasis-open.org/security/saml/Post2.0/sstc-request-initiation.xsd

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

[RFC2616] R. Fielding, et. al. *Hypertext Transfer Protocol 1.1*. IETF RFC 2616, June 1999. http://www.ietf.org/rfc/rfc2616.txt

[SAML2Bind] OASIS Standard, *Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0*, March 2005. http://docs.oasis-open.org/security/saml/v2.0/saml-bindings-2.0-os.pdf

[SAML2Core] OASIS Standard, *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0*, March 2005. http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf

[SAML2Errata] OASIS Approved Errata, *SAML V2.0 Errata*, October 2009. http://docs.oasis-open.org/security/saml/v2.0/sstc-saml-approved-errata-2.0.pdf

[SAML2Meta] OASIS Standard, *Metadata for the OASIS Security Assertion Markup Language (SAML) V2.0*, March 2005. http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf

[SAML2Prof] OASIS Standard, *Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0*, March 2005. http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf

1.3 Non-Normative References

[IdPDisco] OASIS Committee Specification, *Identity Provider Discovery Service Protocol and Profile*, March 2008. http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery.pdf

[WS-Fed] OASIS Standard, *Web Services Federation Language V1.2*, May 2009. http://docs.oasis-open.org/wsfed/federation/v1.2/os/ws-federation-1.2-spec-os.pdf

2 Service Provider Request Initiation Protocol and Profile

2.1 Required Information

Identification: `urn:oasis:names:tc:SAML:profiles:SSO:request-init`

Contact information: security-services-comment@lists.oasis-open.org

Description: Given below.

Updates: None.

2.2 Protocol Description

This protocol is used to ask that a service provider supporting a federated authentication protocol produce a request for authentication using particular options or assumptions. It is assumed that the user wields a standard HTTP user agent. The protocol is specified between the user agent and the service provider. Any technical means may be used to cause the user agent to submit a request using this protocol, including static or dynamic links on any web site, client-side scripting, manual entry by a user, etc.

The request initiation protocol consists of a single HTTP [RFC2616] request/response, a normative request followed by an arbitrary response from the service provider. This response MAY be a request for authentication using a selected protocol (the format of which is left to the definition of that protocol), or it MAY be a refusal to perform the requested action or any other response the service provider deems appropriate.

In the event of failure, the response SHOULD inform the user as to the nature of the problem or prompt for additional information as required. For example, in the event that the request does not identify the identity provider to use, the response could be in the form of a request to a discovery service, per [IdPDisco].

2.3 HTTP Request Format

The request to the service provider MUST use the GET method, and MAY contain one or more URL-encoded query string parameters, as defined below. Parameter names are case-sensitive.

Implementations that do not recognize a parameter defined other than within this specification (i.e., an extension defined privately or separately) MUST ignore that parameter.

2.3.1 Defined Parameters

`entityID`

The unique identifier of an identity provider the service provider is instructed to use. If it cannot or will not do so, the service provider MUST NOT return a request for authentication to a different identity provider (i.e., it MUST NOT ignore the choice). If this parameter is omitted, the service provider is free to respond in any fashion it wishes, including but not limited to the use of any supported discovery mechanism to determine the identity provider itself.

`target`

The location of a resource to which the user agent should be returned, when possible, following successful authentication. If this parameter is omitted, the service provider MUST use a default value (which it unilaterally determines).

`isPassive`

A boolean value of "true" or "false" that indicates whether the request generated by the service provider should include an option to prevent visible user interaction with the identity provider. This corresponds to the SAML 2.0 `IsPassive` attribute in a `<samlp:AuthnRequest>` message.

If this parameter is present and "true", and the authentication protocol supported by the service provider and identity provider in common does not support this feature, then the service provider MUST redirect the user agent to the value of the `target` parameter.

`forceAuthn`

A boolean value of "true" or "false" that indicates whether the request generated by the service provider should include an option to bypass an existing security context and require explicit user interaction during authenticaton to the identity provider. This corresponds to the SAML 2.0 `ForceAuthn` attribute in a `<samlp:AuthnRequest>` message.

If this parameter is present and "true", and the authentication protocol supported by the service provider and identity provider in common does not support this feature, then the service provider MUST NOT return a request for authentication.

2.3.2 Extensions

Parameters whose name begins with the case-sensitive string "`ext_`" are reserved for future use by this or related specifications from this Technical Committee and MUST NOT be used for third-party extensions of this protocol.

Parameters other than those specified above, or with the "`ext_`" prefix, MAY be present, but their meaning is undefined by this specification.

The conventions for naming extensions are somewhat counter-intuitive but are necessary for compatibility with existing implementations.

2.4 Use of Metadata

This protocol exists outside the purview of actual authentication protocols, but for documentation purposes, or as an aid in the dynamic construction of links in support of this protocol, service providers that are described using the SAML V2.0 Metadata specification MAY document endpoints supporting this protocol using an extension element, `<init:RequestInitiator>`, of type **md:EndpointType**. The `Binding` attribute of the extension element MUST be set to:

`urn:oasis:names:tc:SAML:profiles:SSO:request-init`

The schema for the `<init:RequestInitiator>` element is as follows:

```
<schema
    targetNamespace="urn:oasis:names:tc:SAML:profiles:SSO:request-init"
    xmlns:init="urn:oasis:names:tc:SAML:profiles:SSO:request-init"
    xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
    xmlns="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="unqualified"
    attributeFormDefault="unqualified"
    blockDefault="substitution"
    version="1.0">
    <annotation>
        <documentation>
            Document identifier: sstc-request-initiation
```

```
            Location: http://www.oasis-open.org/committees/documents.php?
wg_abbrev=security
            Revision history:
            V1.0 (March 2010):
              Initial version.
        </documentation>
    </annotation>
    <import namespace="urn:oasis:names:tc:SAML:2.0:metadata"
        schemaLocation="saml-schema-metadata-2.0.xsd"/>
    <element name="RequestInitiator" type="md:EndpointType"/>
</schema>
```

2.5 Security Considerations

Some authentication protocols may involve the use of digital signatures or other cryptography, and thus the generation of requests by a service provider may be computationally intensive. In such cases, support for this protocol could provide a Denial of Service opportunity for an attacker, but not typically a new or distinct one.

The ability to externally specify an identity provider could give an attacker the ability to derive information about the sources of authentication trusted by a service provider based on its willingness or lack thereof to respond with an authentication request or an error.

Exposing control over portions of the authentication request process to an outside agency could introduce vulnerabilities if a service provider implementation is not careful in interpreting authentication responses on their own merits rather than making assumptions about its requests. This is not dissimilar from the requirements associated with handling IdP-initiated responses and should not generally create new complications.

Finally, values of the `target` parameter should always be sanitized where used in the generation of responses to user agents, to protect against cross-site scripting attacks and related problems.

3 Conformance

3.1 Service Provider Request Initiation Profile Version 1.0

A conforming Service Provider MUST conform to the normative statements in section 2 that pertain to Service Provider behavior, and MUST properly interpret all the parameters defined in section 2.3.1 in the manner prescribed in that section.

Appendix A. Acknowledgements

The editors would like to acknowledge the contributions of the OASIS Security Services Technical Committee, whose voting members at the time of publication were:

- Rob Philpott, EMC Corporation
- John Bradley, Individual
- Scott Cantor, Internet2
- Nate Klingenstein, Internet2
- Bob Morgan, Internet2
- Thomas Hardjono, M.I.T.
- Anthony Nadalin, Microsoft Corporation
- Tom Scavo, National Center for Supercomputing Applications (NCSA)
- Frederick Hirsch, Nokia Corporation
- Thinh Nguyenphu, Nokia Siemens Networks Gmb
- Paul Madsen, NTT Corporation
- Ari Kermaier, Oracle Corporation
- Hal Lockhart, Oracle Corporation
- Emily Xu, Oracle Corporation
- Anil Saldhana, Red Hat
- David Staggs, Veterans Health Administration

Appendix B. Revision History

- Draft 01, first working draft based on Shibboleth implementation of the protocol.
- Draft 02, clarify handling of unrecognized parameters.
- Committee Draft 01, CD edits.