Identity Provider Discovery Service Protocol and Profile

Committee Specification 01
27 March 2008

Specification URIs:

This Version:
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery-cs-01.html
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery-cs-01.odt
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery-cs-01.pdf

Previous Version:
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery-cd-02.html
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery-cd-02.odt
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery-cd-02.pdf

Latest Version:
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery.html
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery.odt
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery.pdf

Technical Committee:
OASIS Security Services TC

Chair(s):
Hal Lockhart, BEA Systems, Inc.
Brian Campbell, Ping Identity Corporation

Editor(s):
Rod Widdowson, Edinburgh University
Scott Cantor, Internet2

Related Work:
This specification is an alternative to the SAML V2.0 Identity Provider Discovery profile in the SAML V2.0 Profiles specification [SAML2Prof].

Declared XML Namespace(s):
urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol

Abstract:

Defines a generic browser-based protocol by which a centralized discovery service implemented independently of a given service provider can provide a requesting service provider with the unique identifier of an identity provider that can authenticate a principal.

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

Table of Contents

1 Introduction

This specification defines a browser-based protocol by which a centralized discovery service can provide a requesting service provider with the unique identifier of an identity provider that can authenticate a principal. Thus, the protocol provides an alternative means of addressing section 4.1.3.2 of [SAML2Prof]. The profile for discovery defined in section 4.3 of [SAML2Prof] is similar, but has different deployment properties, such as the requirement for a shared domain.

Instead, this profile relies on a normative, redirect-based wire protocol that allows for independent implementation and deployment of the service provider and discovery service components, a model that has proven useful in some large-scale deployments in which managing common domain membership may be impractical.

Note that most Web SSO protocols and profiles, including the multiple versions of SAML, share similar properties and requirements for identity provider discovery (although terminology often differs). This protocol, while suited to SAML V2.0 SSO requirements, is not specific to them.

1.1 Terminology

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 [RFC 2119].

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
`md:`	urn:oasis:names:tc:SAML:2.0:metadata	This is the SAML V2.0 metadata namespace defined in the SAML V2.0 metadata specification Error: Reference source not found.
`idpdisc:`	urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol	This is the SAML V2.0 metadata extension namespace defined by this document and its accompanying schema [IDPDisco-XSD].
`xsd:`	http://www.w3.org/2001/XMLSchema	This namespace is defined in the W3C XML Schema specification [Schema1]. In schema listings, this is the default namespace and no prefix is shown.

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

1.2 Normative References

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

[IDPDisco-XSD] S. Cantor et al. Metadata Extension Schema for Identity Provider Discovery Service Protocol, OASIS SSTC January 2007. Document ID sstc-saml-idp-discovery.xsd. See http://www.oasis-open.org/committees/security/.

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

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

[SAML2Meta-xsd] S. Cantor et al. SAML V2.0 metadata schema. OASIS Standard, March 2005. Document ID saml-schema-metadata-2.0. See http://docs.oasis-open.org/security/saml/v2.0/saml-schema-metadata-2.0.xsd.

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

[Schema1] H. S. Thompson et al. *XML Schema Part 1: Structures.* World Wide Web Consortium Recommendation, May 2001. See http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/.

1.3 Non-Normative References

[ShibProt] S. Cantor et al. *Shibboleth Architecture: Protocols and Profiles*. Internet2-MACE, September 2005. Document ID internet2-mace-shibboleth-arch-protocols. http://shibboleth.internet2.edu/shibboleth-documents.html.

2.4 Conformance

2.4.1 Identity Provider Discovery Protocol Profile

An implementation of this profile shall be a conforming Service Provider or a conforming Discovery Service (or both):

1. A conforming Service Provider MUST conform to the normative statements in section 2 that pertain to Service Provider behavior.
2. A conforming Discovery Service MUST conform to the normative statements in section 2 that pertain to Discovery Service behavior.

All conforming Implementations MUST support, at minimim, a discovery service policy value of "urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol:single", which is the default value.

2 Identity Provider Discovery Protocol and Profile

2.1 Required Information

Identification: `urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol`

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

Description: Given below.

Updates: Provides an alternative to the cookie-based discovery profile in section 4.3 of [SAML2Prof].

2.2 Background

Approaches to web single sign-on (SSO) typically fall into the broad categories of "active" and "passive" profiles, based on the capabilities of the client. So-called passive profiles rely on the features common to web browsers without extensions or plugins that would require additional downloads or operating system support, while active profiles require more advanced (and today at least, generally undeployed) capabilities. The SAML standard includes both kinds of profiles.

One problem that distinguishes federated deployment of passive SSO profiles is identity provider discovery. Passive profiles rely on the service provider, often termed a relying party, to relay (using GET or POST) the user agent to the identity provider. Leaving aside some of the security considerations that this introduces, a fundamental problem exists: how does the service provider know where to send the user agent?

There are a wide range of "solutions" to this problem, but they all share the trait of functioning well only in the presence of certain assumptions about the nature of the deployment and the expectations of users. For example, the most straightforward approach is for each service provider to simply ask the user (and possibly cache the result locally). This allows for maximum control over the experience by the service provider, as well as supplying an unambiguous result. It also leads to increased interference with the SSO experience due to per-site prompts, as well as extra work for relying parties and the need for users to understand how to make an unambiguous selection.

At the other extreme, there has been a model promulgated around the idea of discovery as a function of large federations of identity providers, as in the older Shibboleth "WAYF" model [ShibProt], in which the discovery service acts as a proxy for the service provider and relays a request to the selected identity provider. In theory, this seems attractivesince every service provider can share a single point of discovery, and the user experience can be seamless across many/all services. In practice, this model falls apart quickly because of course there is no single point of discovery that accomodates the entire world of federation.

The cookie-based discovery profile in section 4.3 of [SAML2Prof] falls somewhere between these extremes by focusing on deployments in which all the parties can share a presence in a common domain. DNS itself does not constrain the size of such deployments but practical limitations on domain management tends to inhibit truly large-scale use. It also requires a great deal of static pre-configuration, which limits run-time flexibility.

The protocol and profile outlined here is intended as a hybrid of earlier approaches that brings additional benefits, including:

- Independent implementation of the service provider, identity provider, and discovery service components.
- Flexibility with regard to how the discovery process integrates with services.

- Meta-behavior that enables the protocol to "emulate" other, similar proposals observed in existing SAML and non-SAML software.
- Better handling of multi-protocol deployments than the older Shibboleth WAYF model.
- Accomodation for passive SSO (in the SAML 2.0 `IsPassive` sense).
- Less static pre-configuration than a shared domain solution.

All that said, it is not intended as a panacea, but simply an alternative to fill another deployment niche.

2.3 Discovery Policy

To provide for future extensibility, multiple "flavors" of this discovery profile can be defined and selected using a URI-valued `policy` query string parameter.

Currently, only a policy of "urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol:single" is defined and acts as the default value. Additional policies MAY specify alternate or additional behavior to that defined in section 2.4 as long as an alternate policy value is supplied.

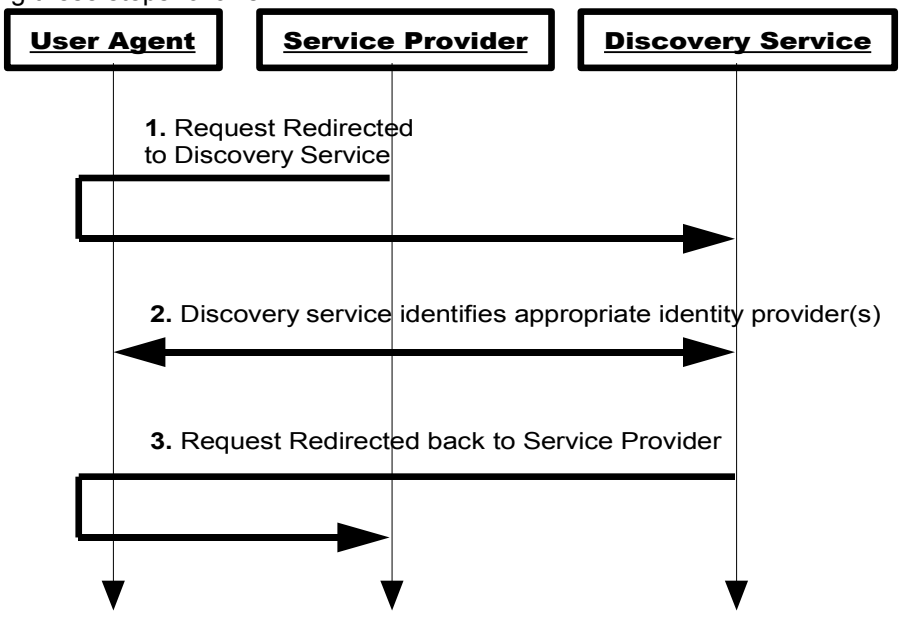
2.4 Protocol Description

This protocol can be used during web-based SSO when a service provider needs to establish an identity provider associated with a principal. It is assumed that the user wields a standard HTTP user agent.

The discovery protocol encompasses three steps, including two normative message exchanges:

1. The service provider redirects the user agent to the discovery service with a set of parameters that make up the request.
2. The discovery service interacts with the principal via the user agent to establish one or more suitable identity providers.
3. The discovery service redirects the user agent back to the service provider with the selected identity provider(s) or an empty response.

A diagram showing these steps follows:



2.4.1 HTTP Request to Discovery Service

In the first step, a requesting service provider redirects the user agent to the discovery service with an HTTP GET request.

The following parameter MUST be present on the query string (and URL-encoded):

`entityID`

The unique identifier of the service provider the end user is (or will be) interacting with, following successful authentication by an identity provider.

The following parameters MAY be present:

`return`

A URL, which MAY itself include a query string. However, such a query string MUST NOT contain a parameter with the same name as the value of the `returnIDParam` parameter in the request (see below) or the name "entityID" if no `returnIDParam` parameter is supplied. (This guards against the possibility of a multiply-valued query string parameter in the response.)

The discovery service MUST redirect the user agent to this location in response to this request (see section 2.4.3). If metadata is used (as in section 2.5), then this parameter MAY be omitted; the return location MUST then be based on the default `<idpdisc:DiscoveryResponse>` element. Otherwise, if metadata is not used, then this parameter becomes mandatory and MUST be present.

`policy`

A parameter name used to indicate the desired behavior controlling the processing of the discovery service. If omitted, it defaults to a value of "urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol:single".

`returnIDParam`

A parameter name used to return the unique identifier of the selected identity provider to the original requester. If this parameter is omitted, it defaults to a value of "entityID". This parameter can be used to customize the response to the service provider so that software relying on alternate approaches to discovery can be utilized in conjunction with this protocol.

`isPassive`

A boolean value of "true" or "false" that controls whether the discovery service is allowed to visibly interact with the user agent in the second step below. If a value is not provided, the default is "false".

2.4.2 Discovery Service determines appropriate Identity Provider

In this step, the discovery service and user agent interact via unspecified means in order to establish the user's choice of identity provider. This may involve user selection, hints obtained through various means, and filtering based on the service provider (identified by the `entityID` parameter in the first step above), preferred SSO protocols or profiles, etc.

If the `isPassive` parameter is set to “true”, the discovery service MUST NOT visibly take control of the user interface from the requesting service provider and interact with the user agent in a noticeable fashion. Additional redirection is permitted, however, provided the passive guarantee can be met.

The discovery service MAY rely on saved state, such as HTTP cookies, to determine the appropriate identity provider. If a single cookie is used, it SHOULD conform to the name and format specified by the Identity Provider Discovery Profile in section 4.3 of [SAML2Prof].

2.4.3 HTTP Redirect to Service Provider

If the `policy` parameter is omitted or set to "urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol:single", then the single selection policy in effect designates that the discovery service is to respond to the service provider and either return a single selected identity provider or none at all using the processing rules defined in this section. Other `policy` values may define alternate behavior to that defined here.

If an identity provider was determined and other requirements (such as metadata) are satisfied, the discovery service MUST respond by redirecting the user agent back to the requesting service provider with an HTTP GET request, at the location supplied in the `return` parameter in the original request (or to the default location identified in metadata if no such parameter was supplied). The unique identifier of the selected identity provider MUST be included as the value of the query string parameter whose name was specified as the value of the `returnIDParam` parameter in the original request (or `entityID` if no parameter was supplied).

If instead an identity provider was not determined, or the discovery service cannot or will not answer, then the discovery service MAY halt processing by displaying an error to the user agent or MAY redirect the user agent back to the requesting service provider. If the service provider included the `isPassive` parameter in its original request, then the discovery service has no option and MUST redirect the user agent back to the service provider. If it responds, then it MUST NOT include the query string parameter whose name was specified as the value of the `returnIDParam` parameter in the original request (or `entityID` if no parameter was supplied). The absence of this parameter is the indication of failure to return a selection.

Note that the discovery service MUST take care to preserve any query string that may already be present within the `return` URL.

2.5 Use of Metadata

All redirection-based SSO protocols share a common property in that the service provider is permitted to (and in most cases must) redirect the user agent to the identity provider. This creates opportunities for phishing attacks against the user's authentication credentials when weak (but extremely common) forms of authentication such as passwords are used.

This protocol has the potential for creating additional opportunities for phishing if arbitrary web sites are permitted to utilize the protocol and obtain the user's identity provider, the key piece of knowledge required to fake the expected authentication experience. To mitigate this threat, metadata can be used to limit the sites authorized to use a discovery service, without introducing more complex (though stronger) approaches such as message authentication.

A discovery service SHOULD require that the service providers making use of it supply metadata (out of band or using techniques such as those described in the SAML V2.0 Metadata specification [SAML2Meta]).

An extension element, `<idpdisc:DiscoveryResponse>`, of type **md:IndexedEndpointType**, is used to define the acceptable locations to which the discovery service should respond with the user's identity provider. The `Binding` attribute of the extension element MUST be set to:

```
urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol
```

Upon receiving a request, the discovery service SHOULD ensure that it recognizes the requesting service provider, as identified by the `entityID` parameter in the request. The location supplied in the `return` parameter (if any) SHOULD then be compared to the `Location` attribute of any `<idpdisc:DiscoveryResponse>` elements found in the `<md:Extensions>` element of the service provider's `<md:SPSSODescriptor>` element. (Note that the `ResponseLocation` endpoint attribute is

unused in this profile.) When metadata is used, the requesting servce provider MAY also omit the `return` parameter in its request in favor of the default endpoint supplied in its metadata.

In the case that the `return` parameter includes a query string, the discovery service MUST ignore it for the purposes of this comparison.

The schema for the `<idpdisc:DiscoveryResponse>` element is as follows:

```
<schema
    targetNamespace="urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-
protocol"
    xmlns:idpdisc="urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-
protocol"
    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-saml-idp-discovery
            Location: http://www.oasis-open.org/committees/documents.php?
wg_abbrev=security
            Revision history:
            V1.0 (January 2007):
              Initial version.
        </documentation>
    </annotation>
    <import namespace="urn:oasis:names:tc:SAML:2.0:metadata"
        schemaLocation="saml-schema-metadata-2.0.xsd"/>
    <element name="DiscoveryResponse" type="md:IndexedEndpointType"/>
</schema>
```

Appendix A. Acknowledgments

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

- George Fletcher, AOL
- Hal Lockhart, BEA Systems, Inc.
- Steve Anderson, BMC Software
- Jeff Bohren, BMC Software
- Rob Philpott, EMC Corporation
- Carolina Canales-Valenzuela, Ericsson
- Lakshmi Thiyagarajan, Hewlett-Packard
- Anthony Nadalin, IBM
- Scott Cantor, Internet2
- Bob Morgan, Internet2
- Eric Tiffany, Liberty Alliance Project
- Tom Scavo, National Center for Supercomputing Applications (NCSA)
- Peter Davis, Neustar, Inc.
- Jeff Hodges, Neustar, Inc.
- Frederick Hirsch, Nokia Corporation
- Abbie Barbir, Nortel Networks Limited
- Paul Madsen, NTT Corporation
- Prateek Mishra, Oracle Corporation
- Brian Campbell, Ping Identity Corporation
- Anil Saldhana, Red Hat
- Eve Maler, Sun Microsystems
- Emily Xu, Sun Microsystems
- Kent Spaulding, Tripod Technology Group, Inc.
- David Staggs, Veterans Health Administration

Appendix B. Revision History

- Draft 01, initial draft based on document prepared by Rod for Shibboleth project
- Draft 02, default various parameters, add policy extension parameter, clarify some processing rules.
- Draft 03, add background material, clarify DS error handling and query string constraints, add mini-outline of protocol and diagram, switch to IndexedEndpointType for metadata element for easier defaulting.
- Committee Draft 01, boilerplate edits for CD status.
- Draft 04, add conformance section, clarify a metadata issue.
- Committee Draft 02, boilerplate edits for CD status