Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 4, 2015

D. Poehn
S. Metzger
W. Hommel
Leibniz Supercomputing Centre

July 3, 2014

# Integration of Dynamic Automated Metadata Exchange into the SAML 2.0 Web Browser SSO Profile
draft-poehn-dame-01

## Abstract

This document specifies the integration of Dynamic Automated Metadata Exchange (DAME) through an intermediate trusted third party into the Security Assertion Markup Language (SAML) 2.0 Web Browser SSO Profile. It is intended for scenarios in which the a-priori exchange of SAML metadata is neither practical nor mandatory. Besides integrated identity provider discovery, this enables the on-demand, user-initiated, and automated SAML metadata exchange for bi-directional pairing of service providers and identity providers without manual setup, such as joining a federation or configuring a metadata feed.

## Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 4, 2015.

Table of Contents



## 1. Introduction

In a federated identity management scenario, enabling communication between an identity provider and a service provider is possible within trust boundaries, which typically entails joining one or several federations before the exchange of metadata takes place. The exchange of SAML metadata is out of scope of the SAML specifications, but normally done by sharing metadata files of all entities within the trust boundaries.

This document specifies the HTTP-based [RFC7230] integration of SAML metadata exchange into the SAML2 Web Browser SSO [OASIS.saml-profiles-2.0-os] profile. Focusing on the automation and the on-demand initiation of the metadata exchange between an identity provider and a service provider to build a form of opportunistic trust, even if these do not share membership in a common federation a-priori or if regular federation scenarios are not suitable.

The metadata exchange is triggered by a trusted third party, which does not interfere in further communication when identity provider and service provider have established a trust relationship. Integrated identity provider discovery, the mutual exchange of required SAML metadata, and user authentication take place in a fully automated, user-initiated, and on-demand manner. To provide a highly flexible solution, either pull-based metadata exchange, such as MD Query [I-D.young-md-query], or any kind of push mechanism are supported.

This integration does not imply that disclosing personally identifiable information is required from an identity provider by sending it to any particular service provider. This is left to appropriate means, e.g., explicitly acquiring user consent, in compliance with regulations and policies.

The described integration addresses the protocol, content and processing of SAML messages for interoperability, referring to [SAML2Int], but also specifies some deployment details and phases for cross-boundary trust. Fitting in seamlessly with implemented SAML-based SSO workflows and being scalable for a large number of users and entities without exceeding administrative procedures, it enables to participate in dynamically set up federations.

## 1.1. Notation and Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

## 1.2. Terminology

This document uses identity management terminology from [RFC6973] and [OASIS.saml-glossary-2.0-os]. In particular, this document uses the terms identity provider, service provider and identity management. Furthermore, it uses following terminology:

Entity - A single logical construct for which metadata is provided. This is usually either a service provider (SP) or an identity provider (IDP).

Metadata - The SAML metadata specification is a machine-readable description of certain entity characteristics and contains information about identifiers, endpoints, certificates and keys, etc.

Trusted Third Party - An intermediate entity facilitating interaction between different entities, which trust the third party (TTP).

## 2. SAML Profiles and Bindings

Based on [OASIS.saml-profiles-2.0-os], SAML profiles define rules how to embed SAML assertions in or combine them with other objects as files or protocol data units of communication protocols. The profile defined in this document is based on the existing SAML Web Browser SSO profile, which implements the SAML Authentication Request protocol [OASIS.saml-core-2.0-os] enhanced by a trusted entity between an originating party (identity provider) and a receiving party (service provider).

A SAML binding [OASIS.saml-bindings-2.0-os] maps request-response messages of the SAML protocol onto standard communication protocols. For compliance reasons with the underlying Web Browser SSO profile, the SAML HTTP Redirect and HTTP POST Bindings MUST be used.

## 3. Protocol

The protocol defined in this document MUST be divided into two phases:

- Discovery of the appropriate identity provider
- User authentication on behalf of the service provider through a trusted third party
  - A. User authentication to trusted third party
  - B. On-demand metadata exchange
  - C. User authentication to service provider

The protocol defined in this document primarily adds the on-demand metadata exchange between identity provider and service provider, which is triggered by the user. The authentication to the trusted third party step in the latter phase is required due to the security considerations discussed below.

### 3.1. Identifier

entityID - Specifies the unique identity of an entity, whose metadata is exchanged, as specified in [OASIS.saml-metadata-2.0-os] and [OASIS.saml-idp-discovery].

### 3.2. IDP Discovery

A web user attempts to access a secured resource provided by a service provider via an HTTP user agent. Missing an established technical trust relationship, a certain identity provider MUST be discovered by the discovery functionality that is integrated into or accessible via the trusted third party.

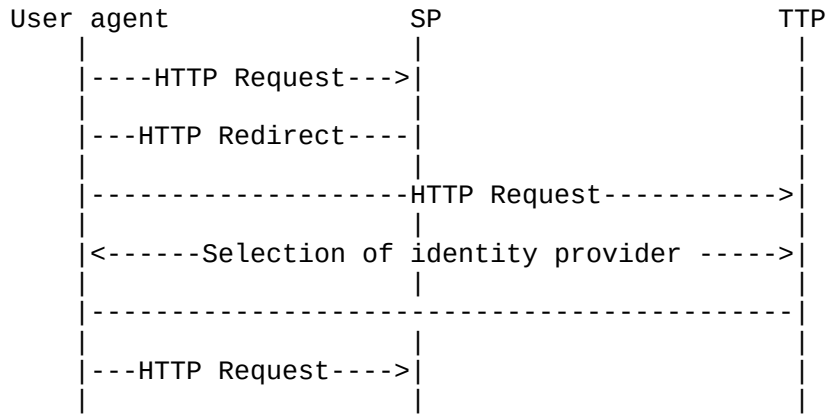


Figure 1

#### 3.2.1. Redirect to trusted third party

Analogous to the SAML identity provider discovery profile [OASIS.saml-idp-discovery], the service provider redirects the user agent to the trusted third party with an HTTP GET request including the REQUIRED or OPTIONAL parameters specified in [OASIS.saml-idp-discovery].

In distinction to the existing discovery profile, the OPTIONAL "isPassive" parameter, which controls the visibly interaction with the user agent, MUST NOT be set to "true" in this profile.

The URLs of the participating entities MUST include "DAME" as last path element before the query element [RFC3986], indicating the usage of this profile for dynamic metadata exchange.

#### 3.2.2. Response to service provider

The trusted third party MUST respond by redirecting the user agent back to the requesting service provider with an HTTP GET request message to the location specified in the return parameter of the original request. The unique identifier of the selected identity provider MUST be included as value of the query string parameter specified as returnIDParam or the entityID if no parameter was supplied.

#### 3.2.3. Failure processing

If the identity provider was not determined or the discovery service cannot answer or an unspecified communication error occurs, the discovery service MAY halt further processing, either displaying an error message to the user agent or redirecting the user agent back to the service provider.

#### 3.2.4. Further actions

After receiving the information about the selected identity provider it is RECOMMENDED that the service provider verifies acceptance. If the identity provider has not been accepted, the service provider halts processing and displays an error message to the user agent.

### 3.3. Authentication Request Protocol using a TTP

In the second phase of the protocol user authentication MUST be performed. The trusted third party authenticates the user on behalf of the service provider. This is REQUIRED to ensure that a metadata exchange will be initiated only if the user has successfully been authenticated by the selected identity provider.

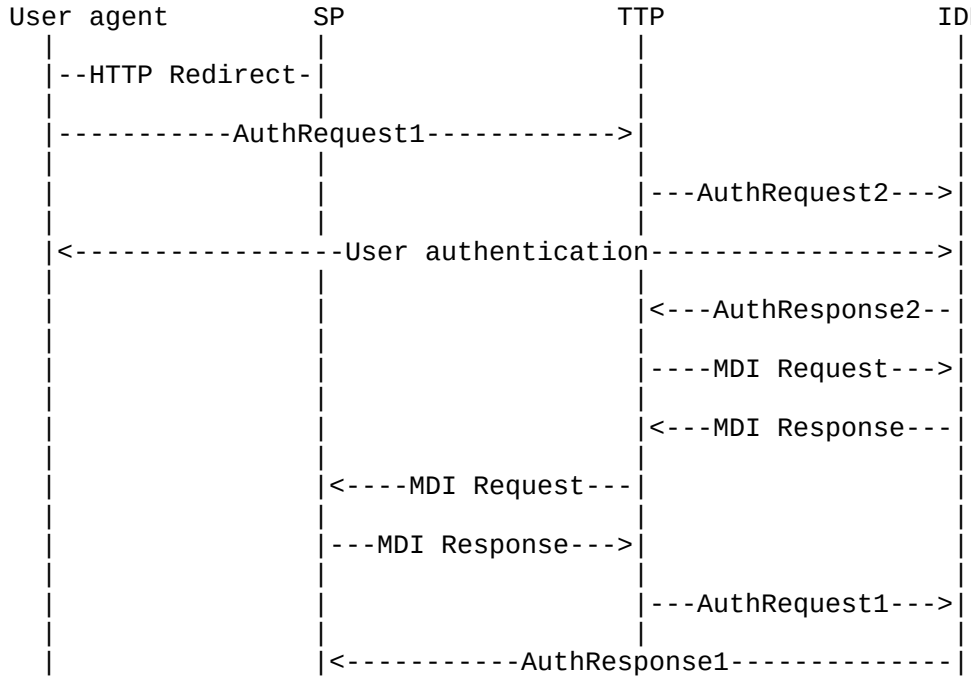


Figure 2: User authentication Request Protocol using a TTP.

Figure 2

### 3.3.1. User authentication to trusted third party

In the first subphase the user MUST be authenticated by the selected identity provider, but in distinction to [OASIS.saml-idp-discovery], the trusted third party initiates the authentication.

#### 3.3.1.1. Authentication Request of SP to TTP

After accepting the selected identity provider, the service provider creates and sends a SAML Authentication Request message (AuthnRequest) to the trusted third party using an HTTP Redirect to transfer the message through the user agent. It is RECOMMENDED that this request message is signed or otherwise authenticated and integrity-protected by the requesting service provider.

#### 3.3.1.2. Store AuthnRequest at TTP

The trusted third party MUST temporarily store the SAML AuthnRequest message by means out of scope of this specification.

#### 3.3.1.3. Authentication Request of TTP to IDP

After that, a second SAML AuthnRequest message MUST be sent by the trusted third party to the selected identity provider using a HTTP redirect message to authenticate the user. The OPTIONAL "ForceAuthn" parameter MAY be included in the request. The AuthnRequest message SHOULD be signed or otherwise authenticated and integrity protected by the trusted third party or by the protocol binding used to deliver the message.

#### 3.3.1.4. User authentication

The user MUST be identified and authenticated by the identity provider by some means out of scope of this profile. Either a new act of authentication MUST be performed or on a previous authenticated session MAY be relied on. A previous session MUST NOT be reused if the request contains a "ForceAuthn" parameter.

#### 3.3.1.5. Authentication Response to TTP

The identity provider MUST issue a SAML AuthnResponse message to the trusted third party containing one or more assertions or an error message with a status describing the error occurred. The HTTP POST binding MUST be used to transfer the message. It is RECOMMENDED that the message is signed by the identity provider or otherwise authenticated or integrity-protected.

### 3.3.2. Metadata Exchange orchestrated by TTP

In the second subphase, the metadata of service provider and identity provider are exchanged in a way that is orchestrated and synchronized by the trusted third party.

#### 3.3.2.1. MDI Request

After the user has been authenticated, the trusted third party MUST initiate the metadata integration (MDI) between identity provider and service provider by a metadata integration request. The MDI request MUST contain "DAME" as last path element. It MUST contain the query elements "action=fetchmeta" and the key element "entityID" with the value element entityID of the requested entity.

The means used for the metadata exchange are implementation-dependent. The trusted third party MAY trigger a metadata query as described by the work in progress about the Metadata Query Protocol [I-D.young-md-query].

Identity provider and service provider MUST integrate each other's metadata in their configuration. It is RECOMMENDED that the identity provider is triggered regarding metadata integration before the service provider because it MAY object to accepting certain service providers. But any kind of concurrent operation MAY be supported.

#### 3.3.2.2. MDI Response

After each other's metadata is integrated, each entity MUST send a metadata integration response message to the trusted third party containing the status of the integration.

If an entity was not able to integrate the metadata before sending the response, the status MUST indicate this state and a new request MUST be sent by the entity containing the status after the integration.

If an error occurs integrating the metadata, the message MUST contain a status describing the error and the trusted third party MUST halt further processing by displaying an error message to the user agent. It is RECOMMENDED to roll back any configuration changes by some means out of scope of this specification.

### 3.3.3. User authentication to service provider

In last step the stored AuthnRequest of the service provider MUST be presented by the trusted third party to the identity provider if no error occured beforehand. Because of the successful user authentication already initiated by the trusted third party, the identity provider SHOULD respond with an assertion transferred to the service provider without further act of authentication, except for the case where the request contains a "ForceAuthn" parameter.

## 4. Security Considerations

### 4.1. Integrity

As SAML metadata contains information necessary for the secure operation of interacting services, it is strongly RECOMMENDED that a mechanism for integrity checking is provided to clients. This MAY include the use of SSL/TLS at the transport layer, digital signatures present within the metadata document, or any other such mechanism.

It is RECOMMENDED that the integrity checking mechanism provided by a responder is a digital signature embedded in the returned metadata document, as defined by [OASIS.saml-metadata-2.0-os] section 3:

- SHOULD use an RSA keypair whose modulus is no less than 2048 bits in length.

- SHOULD NOT use the SHA-1 cryptographic hash algorithm as a digest algorithm.

- MUST NOT use the MD5 cryptographic hash algorithm as a digest algorithm.

- SHOULD otherwise follow current cryptographic best practices in algorithm selection.

### 4.2. Confidentiality

In many cases service metadata is public information and therefore confidentiality MAY NOT be required. In those cases, where such functionality is required, it is RECOMMENDED that both the requester and responder support SSL/TLS. Other mechanisms, such as XML encryption, MAY also be supported for privacy concerns.

### 4.3. Use of SHA-1

This protocol mandates the availability of a identifier synonym mechanism based on the SHA-1 cryptographic hash algorithm. Although SHA-1 is now regarded as weak enough to exclude it from use in new cryptographic systems, its use in this profile is necessary for full support of the SAML 2.0 standard.

Because the SHA-1 cryptographic hash is not being used within this protocol in the context of a digital signature, it is not believed to introduce a security concern over and above that which already exists in SAML due to the possibility of a post-hash collision between entities whose "entityID" attributes hash to the same value. Implementations may guard against this possibility by treating two entities whose "entityID" values have the same SHA-1 equivalent as an indicator of malicious intent on the part of the owner of one of the entities.

### 4.4. Inappropriate Usage

This protocol mandates the authentication of users before any trust between service provider and identity provider is technically established. Although this requires a further step for users, it protects against inappropriate usage of the user-initiated trust

establishment process. Therefore, the user MUST be authenticated before the metadata is exchanged.

### 4.5. Trust

This protocol enables the user to trigger the SAML metadata exchange between two entities and establish the bi-directional technical trust relationship. This is a prerequisite of the subsequent exchange of user information.

For entities, which require a higher level of trust, it is RECOMMENDED to either make use of implementation depending mechanisms in order to secure sensitive information, or to take organizational measures, such as requiring written contracts between service providers and identity providers.

## 5. IANA Considerations

This document has no actions for IANA.

## 6. Acknowledgements

The research leading to these results has received funding from the European Community's Seventh Framework Programme under grant agreement no 605243 (GN3plus).

## 7. References

### 7.1. Normative References

[OASIS.saml-bindings-2.0-os]
Cantor, S., "Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0", March 2005.

[OASIS.saml-core-2.0-os]
Cantor, S., "Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0", March 2005.

[OASIS.saml-idp-discovery]
Cantor, S., "Identity Provider Discovery Service Protocol and Profile", March 2008.

[OASIS.saml-metadata-2.0-os]
Cantor, S., "Metadata for the OASIS Security Assertion Markup Language (SAML) V2.0", March 2005.

[OASIS.saml-profiles-2.0-os]
Cantor, S., "Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0", March 2005.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", January 2005.

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", July 2013.

[RFC7230] Fielding, R. and J. Reschke, "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", June 2014.

## 7.2. Informative References

[I-D.young-md-query]
Young, I., "Metadata Query Protocol, draft-young-md-query-02 [work in progress]", April 2014.

[OASIS.saml-glossary-2.0-os]
Hodges, J., "Glossary for the OASIS Security Assertion Markup Language (SAML) V2.0", March 2005.

[SAML2Int]
Solberg, A., "Interoperable SAML2.0 Web Browser SSO Deployment Profile", .

## Authors' Addresses

Daniela Poehn
Leibniz Supercomputing Centre
Boltzmannstrasse 1
Garching n. Munich, Bavaria 85748
Germany

Phone: +49 (0) 89 35831 8763
Email: poehn@lrz.de

Stefan Metzger
Leibniz Supercomputing Centre
Boltzmannstrasse 1
Garching n. Munich, Bavaria 85748
Germany

Phone: +49 (0) 89 35831 8846
Email: metzger@lrz.de

Wolfgang Hommel
Leibniz Supercomputing Centre
Boltzmannstrasse 1
Garching n. Munich, Bavaria 85748
Germany

Phone: +49 (0) 89 35831 7821
Email: hommel@lrz.de