InCommon Federation SAML 2.0 Profiles

Working Draft 03
February 18, 2010

Document identifier:
draft-incommon-saml2-profiles-03

Location:
TBD

Editors:
Scott Cantor, Internet2 / The Ohio State University

Contributors:
Andreas Åkre Solberg, UNINETT

InCommon Federation Technical Advisory Committee

Abstract:
This document contains implementation and deployment profiles for SAML V2.0 recommended for use within the InCommon Federation. It includes a set of requirements for implementers of SAML products intended for use within the federation, and a narrower set of guidelines for deployers intended to foster interoperability.

Table of Contents

1 Introduction

SAML V2.0 is a rich and extensible standard that must be profiled to be used interoperably, and the profiles that typically emerge from the broader standardization process usually remain fairly broad and include a number of options and features that increase the burden for implementers and make deployment-time decisions more difficult. The InCommon Federation, in consultation with its peer federations around the world, has developed a set of requirements and recommendations for both implementers and deployers that are intended to promote a baseline set of features and options required to interoperate securely and effectively.

It is the intent of the InCommon Federation to participate in the development and support of profiles more broadly, and this document is a reflection of many such discussions (see [SAML2Int] in particular). The profiles defined here may evolve or be superseded in response to future developments where warranted.

1.1 Notation

This specification uses normative text to describe the use of SAML capabilities.

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]:

> …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:

- The prefix `saml2:` stands for the SAML 2.0 assertion namespace, `urn:oasis:names:tc:SAML:2.0:assertion`
- The prefix `saml2p:` stands for the SAML 2.0 protocol namespace, `urn:oasis:names:tc:SAML:2.0:protocol`
- The prefix `md:` stands for the SAML 2.0 metadata namespace, `urn:oasis:names:tc:SAML:2.0:metadata`
- The prefix `idpdisc:` stands for the Identity Provider Discovery Service Protocol and Profile [IdPDisco] namespace, `urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol`
- The prefix `mdattr:` stands for the Metadata Extension for Entity Attributes Version 1.0 [MetaAttr] namespace, `urn:oasis:names:tc:SAML:metadata:attribute`

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

1.2 Normative References

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

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

[RFC2818] IETF RFC 2818, *HTTP Over TLS*, May 2000. http://www.ietf.org/rfc/rfc2818.txt

[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

[MACEAttr] MACE-Dir Working Group Publication, *MACE-Dir SAML Attribute Profiles*, April 2008. http://middleware.internet2.edu/dir/docs/internet2-mace-dir-saml-attributes-200804.pdf

[MetaAttr] OASIS Committee Specification, *SAML V2.0 Metadata Extension for Entity Attributes Version 1.0*, August 2009. http://docs.oasis-open.org/security/saml/Post2.0/sstc-metadata-attr.pdf

[MetaIOP] OASIS Committee Specification, *SAML V2.0 Metadata Interoperability Profile Version 1.0*, August 2009. http://docs.oasis-open.org/security/saml/Post2.0/sstc-metadata-iop.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

[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

[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

[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

[SAML2Err] OASIS Approved Errata, *SAML V2.0 Errata*. http://docs.oasis-open.org/security/saml/v2.0/sstc-saml-approved-errata-2.0.pdf

[SAML2Int] A. Solberg et. al., *Interoperable SAML 2.0 Web Browser SSO Deployment Profile*, Draft. http://saml2int.org/profile/draftic

2 SAML V2.0 Browser SSO Implementation Profile

This profile specifies behavior and options that implementations of the SAML V2.0 Web Browser SSO Profile [SAML2Prof] are required to support. The requirements specified are *in addition to* all normative requirements of the original profile, as modified by the Approved Errata [SAML2Err], and readers should be familiar with all relevant reference documents. Any such requirements are not repeated here except where deemed necessary to highlight a point of discussion or draw attention to an issue addressed in errata, but remain implied.

SAML leaves substantial latitude to implementations with regard to how software is architected and combined with authentication and application infrastructure. Where the terms "Identity Provider" and "Service Provider" are used, they should be understood to include the total software footprint intended to provided the desired functionality; no specific assumptions are made as to how the required features are exposed to deployers, only that there is some method for doing so.

2.1 Required Information

Identification: urn:mace:incommon:profiles:saml2:browser-sso:implementation

Contact information: admin@incommonfederation.org

Description: Given below

Updates: Nothing

2.2 Metadata and Trust Management

Identity Provider, Service Provider, and Discovery Service implementations MUST support the use of SAML V2.0 Metadata [SAML2Meta] in conjunction with their support of the SAML V2.0 Web Browser SSO Profile [SAML2Prof]. Additional expectations around the use of particular metadata elements related to profile behavior may be encountered in subsequent sections.

Implementations MUST support the SAML V2.0 Metadata Interoperability Profile Version 1.0 [MetaIOP]. It is OPTIONAL for implementations to support the generation, publication, or exportation of metadata, but implementations MUST support the following mechanisms for the importation of metadata:

- local file
- remote resource at fixed location accessible via HTTP 1.1 [RFC2616] or HTTP 1.1 over TLS/SSL [RFC2818]

In the case of HTTP resolution, implementations MUST support use of the "ETag" header for cache management; other cache control support is OPTIONAL. Implementations SHOULD support the use of more than one fixed location for the importation of metadata, but MAY leave their behavior unspecified if a single entity's metadata is present in more than one source.

In accordance with [MetaIOP], importation of multiple entities' metadata contained within an `<md:EntitiesDescriptor>` element MUST be supported.

Verification of metadata, if supported, MUST include XML signature verification at least at the root element level, and SHOULD support the following mechanisms for signature key trust establishment:

- direct comparison against known keys
- some form of path-based certificate validation against one or more trusted root certificates and certificate revocation lists

The latter mechanism does not impose a particular profile for certificate validation, as no such profile has wide enough adoption across tools and libraries to warrant such a requirement, but should be understood as being consistent with the "usual" practices encountered in the implementation of certificate validation. Where possible, implementations SHOULD document known limitations of the mechanisms they employ.

Implementations SHOULD support the SAML V2.0 Metadata Extension for Entity Attributes Version 1.0 [MetaAttr] and provide policy controls on the basis of SAML attributes supplied via this extension mechanism.

Finally, implementations SHOULD allow for the automated updating/reimportation of metadata without substantial disruption of services.

2.3 Identity Provider Discovery

Service Provider and Discovery Service implementations MUST support the Identity Provider Discovery Service Protocol Profile in conformance with section 2.4.1 of [IdPDisco].

2.4 Name Identifiers

Identity Provider and Service Provider implementations MUST support the following SAML V2.0 name identifier formats, in accordance with the normative obligations associated with them by [SAML2Core]:

- `urn:oasis:names:tc:SAML:2.0:nameid-format:persistent`
- `urn:oasis:names:tc:SAML:2.0:nameid-format:transient`

Support for other formats is OPTIONAL.

2.5 Attributes

Identity Provider and Service Provider implementations MUST support the generation and consumption of `<saml2:Attribute>` elements that conform to the MACE-Dir Attribute Profile for SAML 2.0 [MACEAttr], with the exception that the ability to support `<saml2:AttributeValue>` elements whose values are not simple strings (e.g., `<saml2:NameID>`, or other XML values) is OPTIONAL.

As a non-normative summary, this requirement primarily implies the capability to ensure the use of particular `Name` and `NameFormat` values when generating and consuming `<saml2:Attribute>` elements, rather than relying on hard-wired assumptions or proprietary sets of attribute identifiers.

2.6 Authentication Requests

2.6.1 Binding and Security Requirements

Identity Provider and Service Provider implementations MUST support the use of the HTTP-Redirect binding [SAML2Bind] for the transmission of `<saml2p:AuthnRequest>` messages, including the generation or verification of signatures in conjunction with this binding.

Because verification of signatures by Identity Providers cannot be guaranteed in deployments, Service Provider implementations MUST NOT rely on the integrity of a signed request for the enforcement of requirements derived from options such as the `ForceAuthn` attribute or the `<saml2p:RequestedAuthnContext>` element. Rather, Service Providers MUST enforce such requirements based on the content of the `<saml2p:Response>` messages they receive.

Support for other bindings is OPTIONAL.

2.6.2 Message Content

In addition to standard core- and profile-driven requirements, Service Provider implementations MUST support the inclusion of at least the following `<saml2p:AuthnRequest>` child elements and attributes (when appropriate):

- `AssertionConsumerServiceURL`
- `ProtocolBinding`
- `ForceAuthn`
- `IsPassive`
- `AttributeConsumingServiceIndex`
- `<saml2p:RequestedAuthnContext>`
- `<saml2p:NameIDPolicy>`

Identity Provider implementations MUST support all `<saml2p:AuthnRequest>` child elements and attributes defined by [SAML2Core], but MAY provide that support in the form of returning appropriate errors when confronted by particular request options. However, implementations SHOULD fully support the options enumerated above. Implementations MAY limit their support of the `<saml2p:RequestedAuthnContext>` element to the value "exact" for the `Comparison` attribute.

2.7 Responses

2.7.1 Binding and Security Requirements

Identity Provider and Service Provider implementations MUST support the use of the HTTP-POST binding [SAML2Bind] for the transmission of `<saml2p:Response>` messages.

Support for other bindings is OPTIONAL.

Identity Provider and Service Provider implementations MUST support the signing of `<saml2:Assertion>` elements in responses; support for signing of the `<saml2p:Response>` element is OPTIONAL.

Identity Provider and Service Provider implementations MUST support the use of XML Encryption via the `<saml2:EncryptedAssertion>` element; support for the `<saml2:EncryptedID>` and `<saml2:EncryptedAttribute>` elements is OPTIONAL.

2.7.2 Message Content

The Web Browser SSO Profile allows responses to contain any number of assertions and statements. Identity Provider implementations MUST allow the number of `<saml2:Assertion>`, `<saml2:AuthnStatement>`, and `<saml2:AttributeStatement>` elements in the `<saml2p:Response>` message to be limited to one.

In turn, Service Provider implementations MAY limit support to a single instance of those elements when processing `<saml2p:Response>` messages.

It is OPTIONAL for Identity Provider implementations to support the inclusion of a `Consent` attribute in `<saml2p:Response>` messages.

Service Provider implementations that provide some form of session semantics MUST support the `<saml2:AuthnStatement>` element's `SessionNotOnOrAfter` attribute.

3 SAML V2.0 Browser SSO Deployment Profile

This profile is layered on, and supplements, the Interoperable SAML 2.0 Web Browser SSO Deployment Profile [SAML2Int] and identifies InCommon-specific requirements and recommendations that go beyond that specification.

> **Note: The current reference to [SAML2Int] is to a draft version. This profile will remain in draft form until such time as a stable version of that profile is available for reference.**

3.1 Required Information

Identification: urn:mace:incommon:profiles:saml2:browser-sso:deployment

Contact information: admin@incommonfederation.org

Description: Given below, in conjunction with [SAML2Int].

Updates: Nothing

3.2 Metadata and Trust Management

It is the responsibility of each deployment to incorporate the metadata supplied by InCommon into its trust management infrastructure. It is RECOMMENDED that use of the metadata conform to the SAML V2.0 Metadata Interoperability Profile Version 1.0 [MetaIOP] and that metadata be updated at least daily.

3.3 Attributes

It is RECOMMENDED that any `<saml2:Attribute>` elements exchanged via any SAML 2.0 messages, assertions, or metadata conform to the MACE-Dir Attribute Profile for SAML 2.0 [MACEAttr].

Appendix A. Open Issues

- Do we care enough about SessionNotOnOrAfter to require support for it?
- Is making IOP "RECOMMENDED" a sufficient statement for deployers? What would it mean to consume the metadata in a different fashion? Seems like we should make it REQUIRED.
- Should we make IdP-initiated SSO explicitly OPTIONAL, or just allow that Shibboleth is non-conformant for the time being?