SAML 2.0 FAQ

The Security Assertion Markup Language (SAML), developed by the OASIS Security Services Technical Committee, is an XML-based framework for communicating security information across administrative domains. SAML allows entities (like IdPs) to issue identity assertions about a subject to other entities (like SPs).

What's new with SAML V2.0? Read What's New in SAML 2? to learn what's new in SAML V2.0.

General Questions

Are InCommon SPs and IdPs required to support SAML V2.0?

Support for SAML V2.0 is **strongly recommended** in the InCommon Federation. See What's New in SAML 2? for the benefits.

Does the InCommon Discovery Service support SAML V2.0?

Yes, the Discovery Service (DS) supports the SAML V2.0 Identity Provider Discovery Protocol.

Does my software support SAML V2.0?

That depends on your software of course. The Shibboleth 2 software certainly does, as do most other software implementations. Check your software documentation or ask your vendor. (**Note well!** Few implementations support all SAML V2.0 features so let the buyer beware!)

Does InCommon still support SAML V1.1?

Yes, but entities (IdPs and SPs) that support **only** SAML V1.1 are strongly encouraged to upgrade as soon as possible. IdP operators can migrate to SAML V2.0 without having to deploy a second IdP. A careful migration path for SPs is outlined as well.

What are the likely sources of runtime error if I upgrade to SAML V2.0?

SAML Web Browser SSO exhibits two common types of errors (regardless of protocol):

1. An IdP that correctly advertises endpoints in metadata may still be misconfigured in software.

2. An SP with correctly configured software may still have ill-formed or incomplete metadata.

Both of these situations **will** cause runtime errors, but more importantly, both types of **errors will occur at the IdP**.

Questions about IdPs

As an IdP, what must I do to support SAML V2.0?

For a simple and safe migration to SAML V2.0, IdP operators should perform the following sequence of steps (in order):

1. Configure the IdP software to respond to SAML V2.0 authentication requests
2. Test the new SAML V2.0 endpoint(s) by triggering an unsolicited response
3. Add the corresponding SAML V2.0 endpoint(s) to IdP metadata

If you add SAML V2.0 endpoints to your metadata but your software is **not** configured to handle SAML V2.0 authentication requests, users **will** experience runtime errors.

How do I know that the SAML V2.0 endpoints in my IdP's metadata actually work?

If you are unsure, you should probably remove those endpoints from metadata and follow an orderly migration path to SAML V2.0. Once the IdP has been tested, you can add the SAML V2.0 endpoints back to metadata.

Does the choice of protocol affect the attributes asserted by my IdP?

Yes, SAML V1.1 attributes are syntactically different than SAML V2.0 attributes, but the semantics are the same. IdP software automatically asserts the correct attribute format while SP software can parse either one.

Does InCommon have an SP that I can use to test my SAML V2.0 IdP?

Yes, that SP has entityID https://service1.internet2.edu/shibboleth in InCommon metadata. It supports both the SAML V2.0 HTTP-POST binding and the SAML V2.0 HTTP-Artifact binding. You can use these endpoints to test your IdP's support of SAML V2.0 Web Browser SSO. See the article on migrating your IdP to SAML V2.0 for an example.

Questions about SPs

As an SP, what must I do to support SAML V2.0?

For an orderly migration to SAML V2.0 with as little disruption to services as possible, SP operators should perform the following sequence of steps (in order):

1. Add one or more SAML V2.0 endpoints to your metadata
2. Add an encryption key to your metadata
3. Wait for the newly updated metadata to propagate throughout the Federation
4. Configure your software with the corresponding decryption key
5. Configure your software for SAML V2.0 Web Browser SSO

If your software is configured to issue SAML V2.0 authentication requests (which it probably is out of the box) but your metadata does **not** contain SAML V2.0 endpoints, users **will** experience runtime errors.

What will break if I add SAML V2.0 endpoints to my SP's metadata?

If your software is **not** configured to issue SAML V2.0 authentication requests, adding SAML V2.0 endpoints to SP metadata has no effect. This fact forms the basis for an SP's recommended migration path to SAML V2.0.

If I upgrade to SAML V2.0, can I still support SAML V1.1 at the same time?

Absolutely! If the SP software supports both SAML V1.1 and SAML V2.0, it will choose the correct protocol at runtime based on the endpoints found in IdP metadata.

Does InCommon have an IdP that I can use to test my SAML V2.0 SP?

Yes, that IdP has entityID urn:mace:incommon:idp.protectnetwork.org in InCommon metadata. It will issue an **unsolicited response** to an arbitrary SAML V2.0 SP in the InCommon Federation. For example, to issue an unsolicited response to the SP with entityID https://service1.internet2.edu/shibboleth , copy-and-paste the following URL into your browser:

```
https://idp.protectnetwork.org/protectnetwork-idp/profile/SAML2/Unsolicited/SSO?providerId=https%3A%2F%
2Fservice1.internet2.edu%2Fshibboleth
```

To obtain a username/password for the above IdP, visit the ProtectNetwork home page.

To test a specific endpoint at the SP, append a `shire` parameter to the query string:

```
...&shire=https%3A%2F%2Fservice1.internet2.edu%2FShibboleth.sso%2FSAML2%2FPOST
```

Note that the HTTP parameters used to trigger an unsolicited response (`providerId` and `shire`) are the same parameters used in a Shibboleth 1.x `AuthnRequest`, but since the endpoint at the IdP is a SAML V2.0 endpoint, a SAML V2.0 flow is initiated.