Test IdPs in Metadata

Test IdPs in Metadata

The first IdP an organization introduces into metadata is assumed to be a production IdP. Please do not submit temporary IdP metadata with the intention of changing it later on. IdP metadata that is obviously temporary (e.g., metadata that contains the substring "test" in names and locations) will not be approved.

As a matter of policy, each organization is allowed one IdP entity descriptor in metadata. By request, a second IdP in metadata may be purchased for an extra $1,000 per year. This second IdP may be a test IdP. That said, in almost all cases, it is neither necessary nor advised to register a test IdP in metadata.

> ⓘ **Test IdPs in Metadata**
>
> Test IdPs in InCommon metadata serve little or no purpose. Since test IdPs are indistinguishable from production IdPs to both relying parties and end users, the introduction of explicit test IdP metadata is strongly discouraged.

A General Migration Strategy for IdPs

The following migration strategy does **not** require test IdP metadata to be registered with InCommon:

1. **Optimize the production IdP**. Evaluate the use of back-channel protocols on your production IdP with an eye towards eliminating unused protocols and endpoints. Phase out seldom-used protocols if possible. An optimally configured IdP will support SAML2 on the front channel only.
2. **Deploy a test IdP**. Configure this test IdP to be nearly identical to your production IdP (same entityID, same metadata sources, same attribute release policy, etc.).

   > ⚠ Your test IdP should have *the same entityID* as your production IdP so that the two are indistinguishable by relying parties (such that the two really are **one logical IdP**). Consequently, a single entity descriptor in metadata is sufficient to describe both IdPs. Any SP that consumes that metadata will interoperate with either your test IdP or your production IdP.

   There are at least two deployment options:
   a. *Deploy the test IdP on the same host*. In this case, the endpoint locations of the test IdP will have the same hostname but a different path. This is perhaps the simplest option since then the production IdP and the test IdP can easily share the same signing key. (In this scenario, the test IdP is really an extension of the production IdP environment.)
   b. *Deploy the test IdP on a different host*. In this case, the endpoint locations will have a different hostname but the same path as the production IdP. One option is to copy the production signing key onto the new host (without exposing that key of course). Another option is to use a new signing key (which should be no less secure than the production signing key). The certificate corresponding to this new signing key may be added to the IdP's entity descriptor in metadata so that there are two certificates in metadata, one for the production IdP and one for the test IdP.
3. **Exercise the test IdP**. There are at least two test scenarios depending on how the test IdP is deployed:
   a. Using IdP-initiated SSO on the test IdP, systematically push SAML2 assertions to endpoints at select partner SPs.
   b. If the test IdP is deployed on a different host, map the IdP domain name (in metadata) to the IP address of the test IdP using /etc/hosts on a client machine. Using SP-initiated SSO, systematically test select partner SPs using the client machine.