configure-shib-idp

Jump to:

Version requirement | Configuring your IdP | Obtaining metadata signing key

Version requirement

To use the per-entity metadata features (MDQ protocol support) of the InCommon metadata service, you need to run versions 3.0 and higher of the Shibboleth IdP, or other software that supports the protocol.

If you are not running Shibboleth IdP V3 or higher, you should upgrade as soon as possible.

In addition, the requiredSignedRoot property is new as of v3.2.0. Upgrading to the most recent version of the Shibboleth IdP and enabling this feature protects your deployment against man in the middle attacks.

Configuring your IdP

This example illustrates how to configure a DynamicHTTPMetataProvider to consume per-entity metadata with a ten minute minimum cache duration and one day maximum cache duration. Note that we did not configure any timeouts.

The Shibboleth IdP documentation provides more information on all of the options available with the DynamicHTTPMetadataProvider.

Example IdP configuration

```
<!-- InCommon Per-Entity Metadata Distribution Service -->
<MetadataProvider id="incommon" xsi:type="DynamicHTTPMetadataProvider"
                  maxCacheDuration="PT24H" minCacheDuration="PT10M">
  <!-- Verify the signature on the root element (i.e., the
EntityDescriptor element) -->
  <MetadataFilter xsi:type="SignatureValidation" requireSignedRoot="true"
                  certificateFile="%{idp.home}/credentials/inc-md-cert-mdq.
pem" />

  <!-- Require a validUntil XML attribute no more than 14 days into the
future -->
  <MetadataFilter xsi:type="RequiredValidUntil" maxValidityInterval="P14D"
/>

  <!-- The MetadataQueryProtocol element specifies the base URL for the
query protocol -->
  <MetadataQueryProtocol>https://mdq.incommon.org/</MetadataQueryProtocol>
</MetadataProvider>
```

Caching metadata

We strongly recommend that you enable metadata caching to minimize the potential impact of any metadata service interruption.

A short minimum cache duration is recommended in order to prevent failed lookups from being cached for an extended period of time. Note that Shibboleth does not refresh at the minimum cache duration value, so it is okay to have a low minimum cache duration set.

Set the max cache duration to at least one hour, but no longer than one day to avoid your cache from becoming too stale.

Configuring with multiple metadata providers

Related content

- Configure other software
- Configure Shibboleth service provider
- Locating the production metadata
- Migrating to the MDQ Service
- Metadata signing key for the Production environment
- Prefetch an entity with Shibboleth
- Introducing per-entity metadata service
- Metadata Distribution Service Documentation
- Metadata Service
- Configure Shibboleth identity provider

Get help

Can't find what you are looking for?

help Ask the community

If you have more than one metadata provider in your Shibboleth configuration, you will want to put the InCommon Per-Entity Metadata Distribution Service **after** any statically configured metadata providers. If you do not do this, Shibboleth will try to fetch your static entities from InCommon each time it is requested before falling back to your static metadata providers.

attribute-filter.xml

You may also need to review your attribute release policies prior to moving to the Per-Entity Metadata Service. If your Shibboleth IdP has a PolicyRequirementRule that includes a type of InEntityGroup, the filter will need to be rewritten. This policy would be used if there is a certain set of attributes released to all entities in InCommon and eduGAIN. Because the Per-Entity Metadata Service does not provide an EntitiesDescriptor element, a policy filtering on what group an entity is in will fail to release any attributes. Each entity returned through the service will only contain a single EntityDescriptor. The InEntityGroup PolicyRequirementRule will still work with aggregates delivered through the InCommon Per-Entity Metadata Service, but likely what will work better for your IdP is to release the R&S Bundle to All R&S SPs instead of releasing a set of attributes to all of InCommon. You can also tag entity descriptors from a particular metadata source and configure custom attribute policies based on those tags.

relying-party.xml

Additionally, if you are doing a relying party override by group, this configuration will also not work with the Per-Entity Metadata Service. We are working right now on alternatives to this configuration and will update this document with new information. If you have a use-case that relies on relying party override by group, please email help@incommon.org so we can work with you on figuring out a solution.

Obtaining metadata signing key

Download and place the metadata signing key in the credentials folder of the IdP and name it *inc-md-cert-mdq.pem*.

Available Keys

- Metadata signing key for the Production environment