Measuring metadata propagation in the UK federation

Poster presentation at TNC 2018, Trondheim, Norway

Alex Stuart, Principal Technical Specialist (UK federation) at Jisc

alex.stuart@jisc.ac.uk

Introduction

Service owners and other stakeholders often ask “How long will a metadata change take to propagate to federation members?”

Metadata refresh is a pull mechanism so operations staff typically reply with an estimate of “a few hours, or overnight” based on federation recommendations for metadata refresh rate. This estimation has been made more complicated since we have introduced a just-in-time metadata query (MDQ) service alongside classical publication of SAML metadata aggregates.

This poster outlines a method for measuring propagation of SP metadata to IdPs using SAML 2 AuthnRequest messages to probe whether updated metadata has been read by an IdP, and it presents measurements in the UK federation. It confirms that the metadata propagation time depends on the configuration of IdP software for a federation, although it is possible to estimate an upper bound for the time to propagate through the federation.

The method does not rely on operational details of the UK federation, so it can be generalised for other federations or to interfederation through eduGAIN.

SAML 2 AuthnRequest and IdP responses

In typical SAML 2 SSO, the SP sends an AuthnRequest to the IdP:

```
<samlp:AuthnRequest xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
    AssertionConsumerServiceURL="https://probe.example.ac.uk/Shibboleth.sso/SAML2/POST"
    Destination="https://idp.example.ac.uk/idp/profile/SAML2/Redirect/SSO"
    ID="_7fb5fe0e0b983abdbcf829729a3dfe6b" IssueInstant="2018-06-05T10:44:02Z"
    ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
    Version="2.0">
    <saml:Issuer xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
        >https://probe.example.ac.uk/entity</saml:Issuer>
    <samlp:NameIDPolicy AllowCreate="1" />
</samlp:AuthnRequest>
```

IdPs will successfully respond if they have metadata for the probe SP with Issuer entityID and if the `AssertionConsumerServiceURL` is present for the probe SP. If these are not satisfied, then the IdP will reject the AuthnRequest in a way that depends on the software product.

Product	Probe in metadata?	Probe not in metadata?	Has MDQ client?
Shibboleth v3	All respond with	HTTP 400 or 500 response	Yes
Shibboleth v2	HTTP 200 response	200 response + error text in HTML	No
simpleSAMLphp	and username/	200 response + error text in HTML	Yes
OpenAthens MD	password prompt	HTTP 200 response	Not known

Since the Shibboleth v3 IdP has an MDQ client and it is straightforward to check its response without parsing the HTML generated by the IdPs, I have compiled a test dataset of 268 v3 Shibboleth IdPs.

Probe

Potential methods for introducing new SP metadata include:

1) Register a new SP for each run of the probe. This method requires a new SP deployment and registration for each test run.

2) Add a new endpoint to the SP by deploying a new SP instance with a new domain name, edit this to use an existing entityID. This method requires a new SP deployment or adding a vhost, adding an entry to the DNS, and registration of the new endpoint for each test run.

3) Using a Shibboleth SP, one can change the implicit `handlerURL` in the shibboleth2.xml configuration file to a UUID, which will change the default endpoint from `https://probe.example.ac.uk/Shibboleth.sso/SAML2/POST` to `https://probe.example.ac.uk/42c3b379-1624-4d1d-9a2f-4cec60b2ae77/SAML2/POST`. This method requires a unique `handlerURL`, a SP service restart, and registration of the new endpoint.

To make the SP send an `AuthnRequest` to a specific IdP, a test script running on the same virtual machine as the SP software makes a series of HTTP GET requests to the Shibboleth SP’s `RequestInitiator` URL, once for each IdP entity in the test dataset. The test script follows the HTTP 302 redirects and records the final HTTP response code to determine whether the IdP has consumed the updated metadata with the new endpoint.

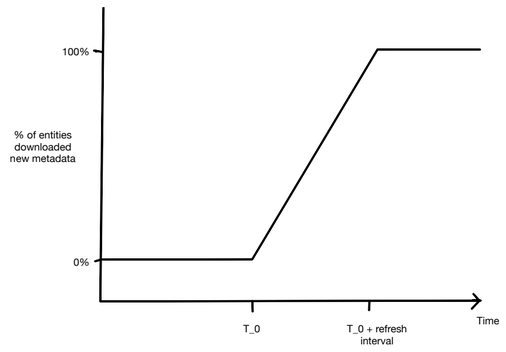
Models of metadata propagation

Model 1: Metadata aggregates with regular refresh

All IdPs refresh their metadata by regularly querying a metadata server using HTTP conditional GETs. An IdP downloads metadata on the first GET after new metadata is published at T_0.

This model assumes there is a single metadata refresh rate. The metadata propagation time is the refresh rate.

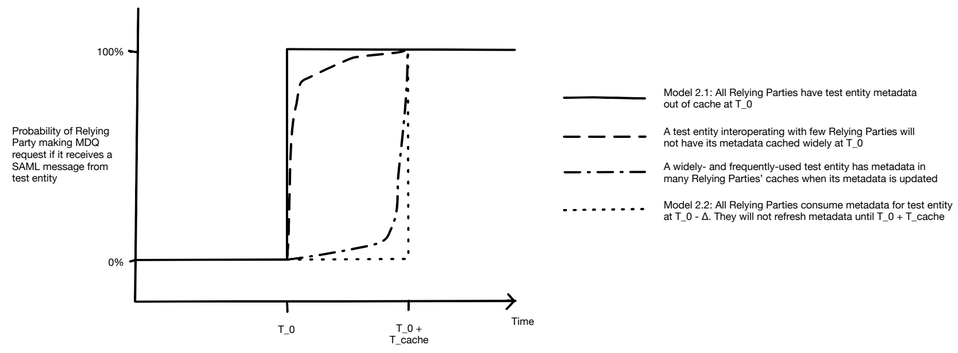
This diagram of % of entities which download metadata over time assumes that there is a large number of IdPs whose conditional GETs are uniformly distributed. This assumption does not change the propagation time.



Model 2: IdPs consume metadata using just-in-time MDQ

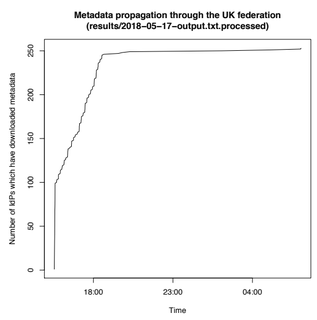
New metadata is published at T_0. IdPs which do not have the probe SP metadata in cache will query for the SP metadata immediately. IdPs which do have the metadata in cache use wait until the metadata is out of cache.

This model assumes all entities have the same cache refresh duration, T_cache. The upper bound to the metadata propagation time is T_cache.



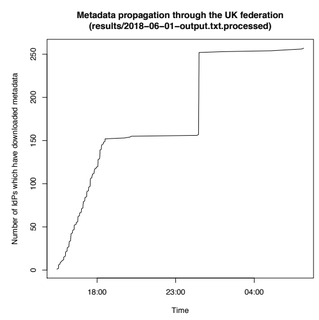
Results

The UK federation makes one publication per day, signing and publishing the metadata aggregates and MDQ responses at approximately 16:30 local time. Here are graphs of the number of IdPs which consume the new metadata over time.



Run 1: Probe metadata out of cache

Probe starts immediately after new metadata published. As soon as probe starts, MDQ clients read refreshed metadata. Almost all clients using aggregates refresh in the following 4 hours.



Run 2: Probe metadata in cache

Probe starts a few minutes before new metadata published. Clients using aggregates (mostly) refresh within 4 hours after publication. MDQ clients refresh when probe metadata goes out of cache, at 00:30 GMT (9 hours later)

Conclusions

- One can measure MDQ usage by remote probe of IdPs (approximately 100 use MDQ)
- IdPs typically use default settings for metadata refresh rate / cache duration
- A few IdPs refresh metadata aggregate at specific times of night
- MDQ can be slower to update than aggregates because default cache duration is larger than default metadata aggregate refresh rate
- We can estimate the upper bound as max{aggregate refresh rate, MDQ cache duration}
- Metadata does indeed take “a few hours, or overnight” to propagate

Potential future directions

Deepen understanding of behaviour of IdP metadata consumption
- Understand caching behaviour of MDQ clients
- Widen range of IdP products that can be tested
- Need to categorise responses and parse the HTML return page

Measure IdP metadata propagating to SPs
- Metadata updates from IdPs are in response to different drivers than updates from SPs
- No analogue of AuthnRequest, although potential to send an unsolicited response
- Wide range of SP software would require compiling a database of responses
- SPs typically don’t use MDQ
- No guarantee that SPs would consume metadata from, or respond to, a Test IdP

Application functionality
- The current probe script uses a naive polling mechanism & doesn’t scale well.
- The probe script has a simple command line interfaces. UX could be improved.
- Script lives on GitHub. Could package and integrate with other tools better.

Measure metadata propagation outside UK federation
- Need permission to probe outside UK federation
- Complexity added by eduGAIN aggregation process
- Metadata propagation times are asymmetric. The time to propagate from UK federation to another federation is not the same as time inbound into UK federation due to interplay between eduGAIN aggregation process and daily UK federation publishing run.

Code, instructions for use, and other information is at https://github.com/alexstuart/MetadataPropagation