Metadata Query Server

Metadata Query Server

To support the Per-Entity Metadata Pilot, a m*etadata query server* that implements the Metadata Query Protocol has been deployed. This metadata query server (mdq-beta.incommon.org) is an instance of Ian Young's mdq-server reference implementation of the Metadata Query Protocol. If you find a bug in the server implementation, please record the issue in the project's issue tracker.

> To browse the server's metadata repository of over 5000 entities, visit the all-entities-mdq-beta web page

The metadata query server (mdq-beta) draws its metadata from the InCommon preview aggregate. Metadata is refreshed every hour using the secure metadata refresh process outlined on the Metadata Consumption wiki page. When an entity descriptor is first served, it is extracted from the source metadata, signed, and cached for future use. Each entity descriptor is signed with a private key generated specifically for this Pilot study. (See below for a procedure to bootstrap trust in this metadata query server.)

This metadata query server (mdq-beta) is monitored 24x7. If a problem is detected, we will do our best to restore normal server operation as soon as possible.

Protocol Overview

A typical request to the metadata query server is a GET request at a URL constructed from the following base URL:

- http://mdq-beta.incommon.org/global

To construct a metadata location, append the URL-encoded entityID of the desired entity to the base URL as specified in the Metadata Query Protocol. For example, for entity https://webauth.cmc.edu/idp/shibboleth the full URL is:

- http://mdq-beta.incommon.org/global/entities/https%3A%2F%2Fwebauth.cmc.edu%2Fidp%2Fshibboleth

When you click the above link, the metadata query server (mdq-beta) returns an HTML page suitable for browsing by a human. For machine-readable XML metadata, use some other (non-browser) HTTP client. For instance, you can use `curl` at the command line to retrieve per-entity metadata from this metadata query server in machine-readable form:

> Use a shell script (mdq_url.sh) to fetch SAML metadata at the command line

```
# Fetch a signed SAML entity descriptor
$ /usr/bin/curl --silent http://mdq-beta.incommon.org/global/entities
/https%3A%2F%2Fwebauth.cmc.edu%2Fidp%2Fshibboleth
```

The output of the above command is a single entity descriptor, signed with a 2048-bit private key. See the next section for instructions how to obtain an authentic copy of the corresponding metadata signing certificate.

Bootstrapping Trust

To ensure the security of your dynamic metadata query process, you must verify the XML signature on each and every entity descriptor you consume. To do that, you need *an authentic copy of the metadata signing certificate*. The certificate must be obtained securely since all subsequent operations depend on it.

To obtain an authentic copy of the metadata signing certificate for the metadata query server (mdq-beta), perform the following steps:

1. Download a copy of the metadata signing certificate via a secure channel
2. Compute the SHA-1 and SHA-256 fingerprints of the metadata signing certificate so obtained
3. Compare the computed fingerprints to the **actual fingerprints** obtained via an independent secure channel

The latter two steps guarantee the integrity of the metadata signing certificate so obtained.

> **Check the integrity of the metadata signing certificate!**
>
> To bootstrap your trusted metadata query process, you MUST check the integrity of the metadata signing certificate configured into that process. It is **not** sufficient to fetch the certificate via a TLS-protected HTTPS connection.

You may check the integrity of the downloaded certificate in a variety of ways. For example, on a GNU/Linux system, you could use `curl` and `openssl` to perform the first two steps of the bootstrap process:

```
# Step 1: Download a copy of the metadata signing certificate via a secure
channel
$ MD_CERT_LOCATION=https://ds.incommon.org/certs/mdq-beta-cert.pem
$ MD_CERT_PATH=/path/to/mdq-beta-cert.pem
$ /usr/bin/curl --silent $MD_CERT_LOCATION > $MD_CERT_PATH

# Step 2: Compute the SHA-1 and SHA-256 fingerprints of the metadata
signing certificate
$ /bin/cat $MD_CERT_PATH | /usr/bin/openssl x509 -sha1 -noout -fingerprint
SHA1 Fingerprint=F0:1D:23:CC:D8:4D:01:53:93:14:26:0A:C6:18:1A:70:BA:B0:00:
E9
$ /bin/cat $MD_CERT_PATH | /usr/bin/openssl x509 -sha256 -noout -
fingerprint
SHA256 Fingerprint=CA:2B:2D:DA:C5:0D:AB:CA:4C:94:43:A4:F7:EF:09:2C:B7:3B:
84:07:2B:6F:05:F0:D1:36:A8:74:2D:6C:B5:32
```

Step 3: The final step is to *compare the computed fingerprints to the actual fingerprints*. The latter are displayed on the following authoritative web page:

- https://ops.incommon.org/mdq_beta_cert.html

If the computed fingerprints match the actual fingerprints, you are done. You may now safely use the certificate to verify the signature on a signed entity descriptor served from mdq-beta.incommon.org.