Thoughts about Interchange Formats

Thoughts about Interchange Formats

Following the principle of leveraging our existing UCTrust / Shibboleth infrastructure, identity information will be transmitted as SAML assertions. To support our three access scenarios, though, we will need to wrap those SAML assertions in another structure that can allow SPs to process transactions and snapshots in a reasonable manner, as well as allow for verification that the information is complete and is not forged.

The User Provisioning Interchange Format (UPIF) consists of a set of one or more SAML objects along with the following information:

- the earliest transaction ID represented by these SAML objects (earliestTransactionID),
- the latest transaction ID represented by these SAML objects (latestTransactionID), and
- a cryptographic signature for the entire UPIF object

The transaction IDs are set according to the access scenario being employed:

- For Snapshot requests, earliestTransactionID will be 0, and latestTransactionID will be the latest transaction that was applied for this snapshot.
- For Change Log requests, the earliestTransactionID will be one greater than the last latestTransactionID returned for this SP by a Snapshot, Change Log, or Subscription request. latestTransactionID will be the latest transaction that was returned for this request.
- For Subscription requests, the first earliestTransactionID will be one greater than the last latestTransactionID returned for this SP by a Snapshot, Change Log, or Subscription request. latestTransactionID will be equal to earliestTransactionID. (Note that Subscription requests return multiple UPIF objects over the lifetime of the subscription. earliestTransactionID and latestTransactionID will increase by one for each UPIF object returned.)

Error Detection and Recovery

For various reasons, an SP and an IdP may get their transaction IDs out of synchronization. It is the responsibility of SPs to detect this by validating the values of earliestTransactionID and latestTransactionID in every UPIF object they receive.

When transaction IDs do get out of sync, SPs must recover by obtaining a Snapshot and assuring that it matches the information they have stored internally before making Change Log or Subscription requests. Failing to do this will result in the SP having incomplete information about its users. The SP can use either of two methods to assure proper matching:

- Delete the internal copy of the information and replace it with the information from the Snapshot. This is probably the easiest to implement, but could cause a service outage while the processing is done.
- Compare the internal copy of the information with the Snapshot to generate updates to the internal information to make it match the Snapshot. While this is probably somewhat more complex to implement, it probably can be done while the service is running.

The choice between these two methods is left to the SP's administrators.