IAM Provider Notes and Observations

IAM Provider Notes and Observations

Initialization

1. The Resource Provider is known to the IAM Provider service by its UCTrust / InCommon assigned identifier (*i.e.*, its entityID).
2. The initialization function allows for the Resource Provider to pass all relevant parameters associated with any combination of scenarios in a single transaction. It is expected to have a list of attribute names (OIDs) within a data framework that allows the IdP to manage such subscriptions on a per SP basis.
3. Any single scenario or combination of scenarios is acceptable.
4. Any initialization scenario the IAM does not handle will be reflected to the SP via response code.
5. The SP may update its interest through re-initializing to any particular IdP. The most recent subscription fully replaces any prior subscription.
6. The most recent initialization becomes the starting point in time for any future scenario three SP requests. In the case of scenario 3 initializations, the IAM Provider returns a transaction identifier value corresponding to a location in the data journal it maintains.
7. Initializations with an empty attribute list result in nullification of the previous SP to IdP initialization request.
8. The SP may attempt to initialize with any particular attribute regardless of ARP. It is the IdP's responsibility to not return any attributes that may not be shared.
9. With a response to the initialization event, the IdP will indicate which attributes, if any, do not meet ARP.
10. The IdP stores the full SP attribute request regardless of ARP. This enables sharing to commence following an update to the IdP ARP.

Scenario One Operation - Snapshot

1. The SP will make a scenario four process call to the IAM provider indicating the request for a complete refresh of all community members within the IAM IdP that are associated with the SP in question.
2. The IAM has an internally available process for interrogating all authentication repositories behind the IdP processes and building a list of all community members associate with a specific SP.
3. The list that is built contains only those attributes which are associated with this SP and which that have been authorized by the IdP in its Attribute Release Policy (ARP).
4. The only implied data ordering within the file is that the entries are in ascending time. The most recent duplicate entry is the authoritative entry.
5. The list is saved as a local data file that will be made available for out of band transmission upon SP initiated request.
6. Only one data file exists per SP. Multiple scenario four requests result in an overlay of the response data file.
7. The SP can optionally make a scenario four process call indicating a reset, with the subsequent deletion of the data file. This is not essential due to point (6).
8. The IAM Provider process uses a locking mechanism to prevent multiple overlapping requests from the same SP.

Scenario Two Operation - Subscription

1. A multi-valued attribute is needed to contain a value per SP that establishes that community member's relationship with that SP. Similar in scope to entitlement.
2. An update to a community member triggers a search for any relying SPs associated with that community member. If one or more are found, the entire set of attributes specified by the SP within its initialization request are packaged and forwarded to the process that handles the "push" transfer to the SP. This is a one time asynchronous event with no provision for error recovery.
3. Error recovery, if desired, can be implemented within the messaging protocol used for transport.

Scenario Three Operation - Change Log

1. A transaction identifier, as created and maintained by the IAM provider, is used by the SP to indicate the starting point for all changes of interest. This transaction identifier maps in some direct fashion to a location within the IAM provider journal. The immediate next entry in the IAM journal is the starting location for records to be fetched. Specifics of this are in Thoughts about Interchange Formats.
2. SPs may not request data that chronologically precedes the most recent initialization event.
3. The IdP maintains a journal of all attributes associated with all community members who are associated with an SP requesting this service. This is a superset of the initialization attribute request list.
4. There is a single chronologically ascending journal regardless of the number of subscribing SPs.
5. Local policy determines the frequency with which this journal is pruned of older data.
6. Data may not be pruned in a selective manner from within the journal. It must always start with the oldest data.
7. A transaction identifier request that goes farther back than the contents of the journal is not an error.
8. The IAM scenario three process parses the journal according to the specific SP's initialization request and builds a local file containing the relevant data.
9. The scenario three process returns to the SP requestor with the local data file name as a result. The SP is expected to invoke an out-of-band process that returns the data file contents.
10. The SP is expected to make a scenario three process call to the IAM indicating reset. Upon a reset, the IdP deletes the local data file.
11. Scenario three requests by the SP are cumulative and appended to any previous data file still present for that SP. There will therefore only be one data file held for each SP.
12. The IAM Provider process uses a locking mechanism to prevent multiple overlapping requests from the same SP.

Scenario Four Operation - SSO Event

1. A successful SSO logon event returns all attributes specified in the designated SP's initialization request within the successful authentication event assertion. A subsequent SAML query is not required.

Observations

IdPs rarely have a mechanism for remembering individual attribute changes within a person object. We must therefore always reflect every attribute subscribed to by an SP whether or not that specific attribute has changed. If a person object is only updated from a single IAM provider subscription, there are no issues. If the same person object is reflected out of multiple IAM provider subscriptions, there is the possibility that one or more attributes will be improperly overlaid.

Scenario one creates the lightest implementation footprint. The only downside is that a particular attribute may be of value to the SP in some frequently executed asynchronous process prior to the next logon event.
Scenario three and four may result in large unbounded response files. It is recommended that the message transfer protocol employed for the file retrieval be robust.

Scenario three and four may result in an unintended viral growth of response files. A protection mechanism utilizing some form of limiting is recommended.

There is no mechanism for the reflection of changed data within the SP context back to the IdP. Attributes determined to be in error by the consuming organization must be corrected through IAM related processes. Otherwise, the same errata will continue to overlay data stored within the SP.

All error recovery scenarios are the responsibility of the SP receiving or retrieval processes.