An IdAS look at IGF

Jim Sermersheim

IGF overview

- <brief igf overview here>
- Relevant papers:
  - <IGF overview>
  - <MRD>
  - CARML
  - AAPML

Some functional IGF requirements

- allow for intended usage statement in requests
  - intended attributes as well as intent to propagate, store, cache, or need to update
  - can be passed in advance or as part of exchange
- allowable usage can be associated with data returned
- discovery based on requirements

Functional req'ts cont.

- fine-grained error reporting
  - i.e. allow a partial subject to be returned with specific errors indicating why certain attributes were withheld
- auditability of actions
- access control model
  - ability to manage (update permissions)
  - ability to query (can Joe perform a read on Alice's telephoneNumber attribute?)
  - enforcement

Functional req'ts cont.

- schema advertisement
- function/feature advertisement
- mapping/obfuscation/filtering/minimization
  - name xlat, masking, value xform, default/fab'd vals
- attributes differ from properties
  - attributes are traditional identifier/value form
  - properties are always true or false
    - examples:
      - IsOverEighteen,
      - Last4SSNDigits is “1234”,
      - PoliticalAffiliation is neither “republican” nor “democrat”

Interesting modular IGF requirement

- one API to allow an app to consume from different sources
  - example is similar to an RP which consumes some identity data from an RSTR and other identity data from a local DB

What IdAS can do today

- allows part of intended usage statement
  - IdAS allows a caller to state which attributes will be read when fetching a subject
  - nothing else is conveyed (intent to propagate, cache, etc.)
  - can't convey in a stateful way
- verifySubjectAttributes allows some types of compare operations similar to IGF “properties” usages, getDigitalSubjects(IFilter) allows others
  - IGF authors see “properties” as a simpler interface

What IdAS can do (cont.)

- IdAS elements allow metadata. This can be used to convey (from producer to consumer) what is allowed to be done to an element (subject, attribute, value, etc.)
  - IGF likely expects this as an AAPML statement
- schema is discoverable, but probably not in any format IGF expects

What IdAS can't yet do

- no ACM or enforcement
- no discovery based on capabilities, schema, access control, etc.
- no way to assert intended usage
- no partial attribute support
- no mapping (only via special CP's)
- no auditing or recommended audit callouts