RIPE
NCC

Personalised Authentication

Tim Bruijnzeels
Assistant Manager Database Group

RIPE 70 | Amsterdam - Netherlands | 14 May 2015

Why?

- Allows for one click creation of person objects
  (if managed by SSO account, will come back to this)
- Maintain your credentials in one place
- Better auditing: can show who made a change to an object to authorised users
- More intuitive authorisation of persons, similar to indicating persons as admin and tech contacts

Schema additions

- Extend person with "auth:" as an optional, multiple attribute, with all current authentication methods
- Extend person “mnt-by:” with a new keyword ‘self’ to indicate that the person object is maintained using its own credentials
- Extend mntner “auth:” attribute to allow a reference to a person object with at least one “auth:” attribute

Persons and SSO - Self-maintained

- mnt-by: ‘self’

- Implies strong one to one relation between a person and an SSO account
  - Need authorisation tokens on both
  - Or create new person
  - Person can have only ONE SSO account (and vice versa)

- Allows user to create and maintain their access account, person object, credentials, name, email, etc.. all in one place with minimal effort.

Clean-up existing auth: sso references?

- If the SSO account appears on a maintainer
- And it’s self maintained
- Then we can replace the “auth: sso” reference with an “auth: person” reference as soon as the association of person and sso is authorised

Persons and SSO - Maintained by others

- mnt-by: <other>
- Propose to not allow “auth:” on person objects managed by others
- Managed persons are fine, but.. people should maintain their own credentials
- We can facilitate an invitation system instead..

Open Issues?

- Should there be business rules to prevent removing the last auth: attribute from a person object that is referenced in an authorisation context?
  - Prevent accidental loss of access to credentials
  - Should dereference this person from mntners first

Questions?