**Background**

- Naming extensions provides a mechanism for accessing named information about a subject in GSS-API
- Using naming extensions for Fedauth
- Integrating SAML and RADIUS attributes into GSS acceptors
SAML

- SAML attributes: format specified by a URI, name specified by format, value specific to name
- Naming exts: SAML attributes are named by a URI
- Desirable for an administrator to be able to specify
Section 6.1 gives rules for when authenticated or asserted attributes are used.

No actual names for Kerberos constructs are provided.

The PAC should be authenticated, right?
Section 6.2.1 is entirely unclear when certificate extensions are authenticated.

otherNames can be confused with keyPurposeIDs and extensions
We must fix these issues
**Authentication and Trust**

- *authenticated*: secure association to trusted source of credentials
- Does not imply authorization to assert the attribute
- Kerberos: probably trusted to assert
- SAML or PKIX: trust is complicated
ISSUERS

- No way to determine who issued an attribute
- Very significant for containers
When should containers be expanded

- Compare: authorization data for KDC certificate against data for PKInit certificate
- Containers may hide purpose, issuer and trust boundaries
Criticality

- No way to represent critical or non-critical attributes
- No way to query critical attributes to confirm they are supported
Possible Solutions
ATTRIBUTE NAMES

- URI describing rest of name attribute form
- Default to single URI for backward compatibility
CONTEXT IN ATTRIBUTE NAMES

- Contain enough context to understand trust implications of an attribute
- Container information
- Information on default trust of mechanism
- No mechanism can name arbitrary authenticated attributes