Current DevID Data Objects and LDevID/IDevID Linkage Revisited

Mike Borza
Decided at last meeting

- Structure of DevID data objects
- LDevIDs will not be linked to IDevIDs as part of the standard-defined data objects
- Linkage may be made at the protocol level or through higher-layer mechanisms
  - e.g. maintain a correspondence database of LDevIDs to IDevIDs
  - e.g. define an option transform in I&A reference protocols that incorporates both IDevID and LDevID
Current structure of DevID

- Presently common structure for both LDevID and IDevID
- DevID structure can be authenticated by itself
- but liveness not assured
  - requires additional information for use in I&A protocols
  - remote-party challenge (random number) will be signed as part of any robust I&A
The case against tying LDevIDs to IDevIDs

- Gets around (some) questions about privacy and anonymity
- Keeps common structure between {IL}DevIDs (so far)
- Captures most common use-cases of interest to enterprises
- Is pretty much minimal structure capable of doing the job
The case for...

- Supports automatic service provisioning with end-user defined & assigned LDevIDs
- Allows tight binding of locally significant identity to physical asset identity
- Interoperability is guaranteed in cases where tying them together is desired
  - this is not true for ad hoc methods
How might this work?

LDevID incorporates the base elements of DevID, is subsequently signed by the corresponding IDevID.

--binding is cryptographic, order is correct (IDevID “vouches” for LDevID)
- still not complete without challenge
- IDevID signing may be optional to support unlinked applications
- both forms may exist simultaneously
- denote unlinked LDevID by NULL IDevID part

<table>
<thead>
<tr>
<th>LDevID</th>
<th>DevID</th>
</tr>
</thead>
<tbody>
<tr>
<td>•issuerID</td>
<td>•IDissuerID</td>
</tr>
<tr>
<td>•uniqueID</td>
<td>•IDuniqueID</td>
</tr>
<tr>
<td>•pubKey</td>
<td>•IDpubKey</td>
</tr>
<tr>
<td>•version</td>
<td>•IDsignature</td>
</tr>
<tr>
<td>•reserved</td>
<td></td>
</tr>
<tr>
<td>•signature</td>
<td></td>
</tr>
</tbody>
</table>
Achieving the same objective with a mutual I&A protocol

DID, LDID are new data objects
- LDID may exist only for purposes of I&A protocols
- ordering of signing is important: IDevID “vouches” for LDevID
- challenges A & B (nonces) assure “liveness” of exchange
- required to prevent playback
Discussion

Do the static signatures in current {IL}DevID structures serve any useful purpose?

Are the reference protocols sufficient to achieve linkage?

- If so, should they be normative?
- define both unilateral and mutual protocols
Decisions

- static signatures are required
  - provide binding to issuer
- LDevID data objects don't need to incorporate IDevID
  - protocol descriptions in standard need to be present and normative for these optional capabilities when provided by the module
- describe both “long-form” and “short-form” exchanges