Validity Management in SPKI

24 April 2002
Yki.Kortesniemi@hut.fi (author)
Tero.Hasu@hut.fi (presentation)
Overview

• Access control
• Types of certificates (and how the type affects validation and revocation)
• Validation and revocation methods in SPKI
• SPKI validity management protocol
Phases of access control

0. Making the decision

1. Expressing the decision (just once)

2. Enforcing the decision (repeatedly)

3. Changing or revoking the decision
Access control w/certificates

0. Making the decision
1. Expressing the decision (just once)
   - Writing and issuing certificate(s)
2. Enforcing the decision (repeatedly)
   - Validating certificate(s)
3. Changing or revoking the decision
   - Modifying external information
Types of certificates

- **Key Authorization**
  - **Name or identity certificate**: e.g. X.509
  - **ACL or attribute certificate**: e.g. SPKI

- **Subject** (person / computer / software agent)
  - **has**
  - **uses**

- **Authorization**
Identity certificates

• Key - Name - Authorization binding proved during validation
  – no anonymity

• Unique name required for each identity across the system
  – otherwise namesakes share rights
  – management burden

• Grouping of rights
  – revoke just one certificate
Authorization certificates

• Key - Authorization binding proved during validation
  – more straightforward
  – performance

• Anonymity is possible
  – benefits privacy of users
  – identity established if required (when acquiring the public key)
Identity certificate issuers

- Capable of establishing identity
- Considered trustworthy
- Typically have plenty of resources
- Small number of issuers (in a system)
- Small number of CRLs
  - may be practical to distribute to access control points
Authorization certificate issuers

- Anyone can be an issuer
- Large number of issuers
- Large number of CRLs
  - impractical to distribute in advance
  - obtain relevant CRLs online when required
- Verifier can also be the issuer
  - issuer arranges revocation mechanisms
  - verifier normally owns protected resource
  - control revocation to balance risk
Validity control in SPKI

- Has to be considered when issuing a certificate
- Validity period
- Online checks
  - CRL
  - reval
  - one-time
  - limit
  - renew
1. Granted
2. Validity period OK and usage not denied by crl or reval
3. Used if not denied by one-time or limit
4. Usage denied by crl or reval
5. Revoked by crl or reval
6. Revoked by crl or reval
7. Expired by time constraint
8. Expired by time constraint
9. Renew — a new certificate is issued
Validity period

1. Granted

2. Validity period OK

3. Used

available

suspended

expired

7. Expired by time constraint

8. Expired by time constraint
CRL, Reval, Renew

1. Granted

2. Validity period OK and usage not denied by crl or reval

3. Used

4. Usage denied by crl or reval

5. Revoked by crl or reval

6. Revoked by crl or reval

7. Expired by time constraint

8. Expired by time constraint

9. Renew – a new certificate is issued
One-time, Limit

1. Granted

2. Validity period OK

3. Used if not denied by one-time or limit

available

4.

suspended

5.

expired

7. Expired by time constraint

8. Expired by time constraint

3. Used if not denied by one-time or limit

available

suspended

expired
## Summary of methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Typical use</th>
<th>Processing overhead</th>
<th>Revocation speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limit</td>
<td>Quota</td>
<td>High</td>
<td>Immediate</td>
</tr>
<tr>
<td>One-time</td>
<td>Limit usage on non-user specific factors</td>
<td>Moderate</td>
<td>Immediate</td>
</tr>
<tr>
<td>Reval</td>
<td>Revocation</td>
<td>Low</td>
<td>After current reval validity period</td>
</tr>
<tr>
<td>CRL</td>
<td>Revocation</td>
<td>Low</td>
<td>After current crl validity period</td>
</tr>
<tr>
<td>Renew</td>
<td>Revocation</td>
<td>Low</td>
<td>After current certificate expires</td>
</tr>
</tbody>
</table>
Management protocol requirements

• Configuration of SPKI validation server
  – can be done remotely
• All SPKI online checks supported
• Certificate issuer can issue commands
  – others need to prove permission
• Status information available
  – use of limited resource can be followed
  – there may be multiple entities with revocation ability
Management protocol design

- Two messages:
  - command
  - reply

- Command message
  - e.g. revoke, re-enable, change quota
  - static and dynamic rules

- Defined in XML
  - signed messages
  - requires secure transport protocol
Command and reply

- server_update cert, chain?, online_test_hash, delete_request*, test_definition*, status_query*, signature

- server_reply cert_hash, online_test_hash, delete_reply*, test_definition_reply*, status_reply*, service_status, signature
The end

• Questions?
  – (hope not)