Trust Anchor Management (TAM) Specifications

March 23rd, 2009

Carl Wallace
cwallace@cygnacom.com

Suggested Way Forward (from Minneapolis)

- New working group last call for new requirements draft after IETF
  - draft-ietf-pkix-ta-mgmt-reqs-03.txt
- Hold working group last call for revised TrustAnchorInfo draft as soon as practical
  - draft-ietf-pkix-ta-format-01.txt
- Revise TAMP spec
  - draft-ietf-pkix-tamp-01.txt
  - Aim for last call shortly after San Francisco

Since Minneapolis

- One revision of each spec
- Current drafts
  - draft-ietf-pkix-ta-mgmt-reqs-03
  - draft-ietf-pkix-ta-format-01
  - draft-ietf-pkix-tamp-01
  - draft-housley-cms-content-constraints-extn-01
- TA mgmt requirements completed WGLC
  - Two edits were made from -02 to -03, which was submitted after WGLC (see next slide)

TAM Requirements changes

- Now limit scope to “push-based” protocols in the abstract
- Removal of section 3.12, which described usage of constraints in cert path validation as a functional requirement
  - This text was moved to the Security Considerations section, stating that application owners must confirm whether the implementations support constraints

TA Format changes

- Removed the taInfo field. The components removed from this structure will now be appear as extensions.
- Replaced references to PublicKeyInfo structure with SubjectPublicKeyInfo. The structures had the same definition and stuff was already imported from RFC 5280.
- Reset the version field to v1.
- Changed tag number of the extensions field since taInfo numbering is no longer an issue. Dropped the [0] tag on the version field as unnecessary.
- Defined TrustAnchorList and the associated object identifier for use with CMS.
- Removed introductory text describing various TA types as irrelevant given relocation of taInfo field contents.
- Relaxed the requirement to enforce TA-based constraints due to similar comments on the requirements draft.
- Removed references to TAMP. This draft is now wholly independent.
- Changed ASN.1 module name to align with registered OID names
- One new OID: id-ct-trustAnchorList

TAMP changes

- Minor wordsmithing throughout including more migration away from "cryptographic module" to "trust anchor store"
- Changed sequence number handling
  - When Certificate and TBSCertificate were added in the last version, sequence numbers were tied to the certificates via the TrustAnchorChoiceWithSeqNumber structure. This structure was cumbersome and has been replaced by a list of pairs of key identifiers and sequence numbers.
  - A field of the new type appears in the following structures: VerboseStatusResponse, TAMPUpdate, VerboseUpdateConfirm and VerboseApexUpdateConfirm. Also added a seqNum field to TAMPApexUpdate.
- Added two new options to TargetIdentifier: URI and otherName. This provides one simple means of addressing a specific store and a means of supporting more complex alternatives.
- Import TrustAnchorChoice from TAF and AnotherName from RFC 5280.
  - TrustAnchorChoice used to be in TAMP but is now in TAF. AnotherName is now used in TargetIdentifier.
- Use SubjectPublicKeyInfo instead of PublicKeyInfo, which was the same structure with a different name.

TAMP changes (continued)

- Added context tags to TBSCertificateChangeInfo. These were missing before and are necessary. Same thing for VerboseStatusResponse.
- Removed taType field from TrustAnchorChangeInfo to align with changes to TrustAnchorInfo.
- Added section describing usage of TrustAnchorList as alternative to TAMPUpdate.
  - Mainly done to align with SIDR (adds an extra SEQUENCE tag in front of the payload they planned to use).
- Added security consideration highlighting replay risk when using TrustAnchorList.
- Changed ASN.1 module names to align with registered OID names

CCC changes

- Changed title to reflect individual submission not working group submission.
- Added Subordination Processing section.
  - This text is mostly unaltered from TAMP. Changes were primarily to shift from references to taType field to extensions field.
- Changed the meaning of extension absence in a certificate.
  - Formerly, absence was equivalent to asserting anyContentType.
  - Absence now results in setting the state variable to empty, which results in the EE has no CCC privileges.
  - Changed to simplify introduction of CCC to existing PKIs.

Suggested Way Forward

- Hold working group last call for revised TrustAnchorInfo draft
  - draft-ietf-pkix-ta-format-01.txt
- Revise TAMP spec
  - draft-ietf-pkix-tamp-01.txt
  - Hold WG last call as soon as practical
- Submit new individual submission that discusses usage of TA-based constraints