Trust Anchor Management (TAM) Requirements

July 30th, 2008

Carl Wallace
cwallace@cygnacom.com

Background

- Trust Anchor Management (TAM) work migrated to PKIX in December from TAM BOF
- TAM requirements were discussed during PKIX meeting in Philadelphia
  - Initial TAM requirements draft submitted as a working group draft in June (adapted from TAM BOF problem statement)
- Trust Anchor Management Protocol (TAMP) and CMS Content Constraints (CCC) drafts expired in April
  - Both were submitted as individual drafts

Requirements

#	Requirement
3.1	Transport independent
3.1	Session oriented and store-and-forward
3.1	Management message integrity
3.2	Determine which TAs are installed in a particular TA store
3.2	Add one or more TAs to a TA store
3.2	Remove one or more TAs from a TA store
3.2	Replace an entire trust store
3.3	Target all TA stores or list of 1 or more stores
3.4	Transfer management responsibility
3.4	Delegation of specific operations
3.5	Manage TAs used to validate certification paths
3.6	Manage TAs that cannot validate certification paths
3.7	Represent TA as (self-signed) certificate or as DN/key/constraints
3.8	Authenticate TA store that produced a report
3.8	Detect replay of TA store reports
3.9	Authenticate TA management data source
3.10	Reduce reliance on out-of-band data
3.11	Detect replay of TA mgmt. transactions/no reliable clock
3.12	Enable recovery from compromise or loss of TA private key

Existing mechanisms

- Four existing mechanisms were evaluated against requirements from -00 draft
  - RFC 5055 (SCVP)
    - Validation policies (ValPolResponse)
  - RFC 4210 (CMP)
    - Root CA Key Update (CAKeyUpdAnnContent)
  - RFC 5272 (CMC)
    - Publish Trust Anchors control (PublishTrustAnchors)
  - TAMP
    - TA Update, Apex Update, Status Query (TAMPUpdate, TAMPApexUpdate, TAMPStatusQuery)
- Initially planned to evaluate RFC 5280 focusing on cross-certificates and subordination
  - Excluded from review since there would still be TAs to manage and support for certificate-based trust establishment is required (section 3.5)

CMP mechanism

- CAKeyUpdAnnContent can be used to announce CA key pair updates
- Structure only supports bilateral certificate issuance
    - Three fields: oldWithNew, newWithOld, newWithNew
    - Text does not require issuer/subject names to match (it’s implied)
    - One certificate must be self-signed

CMC mechanism

- The Publish Trust Anchors control allows for distribution of a set of trust anchors from a central authority to an EE
  - A list of certificate hashes is included in the payload of a SignedData message
  - The certificates are carried in the certificates bag or are otherwise available
- Many details are allocated to an undefined local policy, including:
  - Rules for processing the list of hashes, i.e., replace entire trust anchor store, add certificates associated with the hashes to the trust anchor store, etc.
  - Authorization of the CMC message signer
- Uses values from certificate extensions as inputs to path validation
  - “Information is extracted from [trust anchor certificates] to set the inputs to the certificates validation algorithm in Section 6.1.1 of [PKIXCERT].”
- Requirement to validate publication time is near current time impacts some possible distribution models (i.e., directory)
- Describes authorization via associating source of a trust anchor with the trust anchor as well as types of messages for which the trust anchor is valid

SCVP mechanism

- ValPolResponse could be used to distribute trust anchors for a particular trust anchor store
  - Structure would work for whole store replacement only
- Still requires means of managing SCVP responder keys used to validate ValPolResponse
- ValidationPolicy field provides alternative to certificate extensions for path validation inputs
  - Would apply to all certificates in store

TAMP mechanism

- Three TA mgmt. messages: TAMPUpdate, TAMPApexUpdate, TAMPStatusQuery
  - Each has an associated trust store generated receipt or confirmation message
  - A few other messages related to community management and sequence number management
- Includes subordination rules
- CertPathControls structure provides inputs for path validation
  - User supplied values may restrict the values contained in CertPathControls

Summary view

#	Requirement	TAMP	SCVP	CMP	CMC
3.1	Transport independent	S	S	S	S
3.1	Session oriented and store-and-forward	S	S	S	PS
3.1	Management message integrity	S	S	S	S
3.2	Determine which TAs are installed in a particular TA store	S	NS	NS	NS
3.2	Add one or more TAs to a TA store	S	NS	PS	?
3.2	Remove one or more TAs from a TA store	S	NS	PS	?
3.2	Replace an entire trust store	PS	S	NS	?
3.3	Target all TA stores or list of 1 or more stores	S	NS	NS	NS
3.4	Transfer management responsibility	S	NS	S	PS
3.4	Delegation of specific operations	S	NS	NS	PS
3.5	Manage TAs used to validate certification paths	S	S	S	S
3.6	Manage TAs that cannot validate certification paths	S	NS	NS	NS
3.7	Represent TA as self-signed certificate or as DN/key	S	PS	PS	PS
3.8	Authenticate TA store that produced a report	S	NS	NS	NS
3.8	Detect replay of TA store reports	S	NS	NS	NS
3.9	Authenticate TA management data source	S	S	S	S
3.10	Reduce reliance on out-of-band data	S	PS	S	?
3.11	Detect replay of TA mgmt. transactions/no reliable clock	S	S	S	PS
3.12	Enable recovery from compromise or loss of TA private key	S	NS	NS	NS

Missing Requirements?

- Support for multiple trust anchor stores
  - Naming, TA store discovery, etc.
- Utilization of TA-based information as default inputs to path validation engine
  - CMC supports and RFC 5280 discusses as an option
    - Neither describes reconciliation with user inputs
  - TAMP describes both TA-based information and reconciliation with user data.
    - TA-based info sets broad enterprise parameters
    - User inputs can provide further restrictions
  - Conflicts with notion that TAM addresses back-end changes only

Suggested Way Forward

- Update requirements draft and progress as Informational
- Adopt modified TAMP draft as a Standards track working group draft
  - Move TrustAnchorInfo specification from TAMP to separate draft
  - Provide capability to manage alternative TA formats
    - Minimally, Certificate and TBSCertificate
    - Extensible to support TrustAnchorInfo (and others?)
  - TAMPUpdate would be the primary structure
    - Suitable for directory-based distribution
- Submit new TrustAnchorInfo and CMS Content Constraints drafts compatible with PKIX TAMP