Scenarios for a Trust Anchor Management Protocol

Paul Hoffman, Directory

VPN Consortium

Topics covered in this talk

- Why we are talking about scenarios
- One administrator and/or multiple administrators
- Session-oriented and/or store-and-forward
- Additional scenario choices?

Scenarios for trust anchor management

- Because trust anchor management is a big topic, different people have different perceptions about what it is supposed to be
- Before designing a protocol, we should agree on who it will serve, and how
- This discussion feeds into the requirements document discussion

Terminology

- Trust anchor (TA): definition to be nailed down later
- Client: the system which receives trust anchors using the protocol
- Trust anchor administrator (TAA): the system which gives trust anchor instructions to clients
- TA store: the set of TAs a client has installed at the moment

One administrator and/or multiple administrators

- On the mailing list, the “enterprise scenario” has meant that the client has just one TAA managing their TA store
- A different scenario is that the client gets TAs from multiple independent TAAs

Examples of multiple-administrator use cases

- A person who wants to take trust advice from his IT department, his government, and an independent trust service
- A cell phone or firewall that needs to trust the manufacturer for updates, but that is also used as a web browser
- A contractor who works for multiple enterprises

Design difference between the single-TAA and multiple-TAA scenarios

- In a single-TAA scenario, the TAA can always control the state of the TA store; in a multi-TAA scenario, no single TAA controls the whole store
- A single-TAA protocol needs only one instruction: “here’s your store”
- A multiple-TAA protocol needs more instructions: “add this TA”, “delete that TA”, “here is what is in my store”

Session-oriented and/or store-and-forward

- If both the client and TAA are online, a session-oriented protocol makes sense and is what the IETF is most familiar with
- If they are not both online, a store-and-forward protocol is needed
- A session-oriented protocol can be based on the objects of a store-and-forward protocol

Examples of disconnected systems that need trust management

- A cell phone or firewall whose external interface has not yet been secured
- A device that will never be on the Internet but can read and write messages, such as with a USB drive

Additional scenario choices?

- There may be other scenario choices that need to be made before protocol design is started
- Maybe we want to design just a base protocol that can be easily extended for additional scenarios later