Database of Long-Lived Symmetric Cryptographic Keys

draft-ietf-karp-crypto-key-table

IETF 83 Paris

R. Housley
T. Polk
S. Hartman
D. Zhang

Goals

- Meta Goal: provide secret to routing protocols
- Avoid protocol knowledge in KMPs
- Minimize protocol knowledge in ops model, management, netconf, mibs, and etc
- Constrain protocol-specific knowledge to specified areas
- Uniformity in representation so keys can be shared between vendors

What Key Tables Includes

- Conceptual database of keys
- Description of how a routing protocol fits into the key table architecture
- Textual conventions so one vendor's keys and IDs look like another’s
- Future: operations a protocol uses to interact with a KMP

Key Table Operations

- Protocol looking up key to use to verify received packet
- Protocol looking up key to send a packet
- Administrator deploying, removing or enabling keys

Proposed Changes in Database (1)

- Key names rather than names and Ids
- Protocols can restrict form of names; ours will restrict to numeric Ids
- New entry name for removing rows
- Local and Remote key name only for protocols
- Interface becomes protocol-independent key scope restriction

Proposed Changes in Database (2)

- Peer/group: protocol specific restriction on scope of key
- Protocol specifics: info needed to use key but not to find it
- KDF inputs rolled into protocol specifics
- Many group/unicast distinctions disappear

Textual Conventions

- Avoid WEP problem: multiple ways to turn operator input into keys yields horror. Password? Pass phrase? Raw key?
- Keys are hex strings
- Key Ids are key names represented as hex strings

What Protocols Specify

- Form of key names
- Form of peer/group
- Form of protocol specifics
- Rules for taking operator input turning it into canonical form and validating

What Protocols Specify (2)

- How to take received packet and find the key name and peer
- How to generate cryptographic authentication from a key table entry
- Constraints on KDF, interfaces, and etc.

Issue: Management Approach

- Currently document tends to include all management options from all protocols.
- Alternative: pick best-of and apply a consistent management interface across all protocols
- Over the wire would stay constant but operator experience might change
- More uniform experience

Questions? Comments?