Managing Long-Term Keys for Routing Protocols

July 29, 2010

Russ Housley

Tim Polk

Drafts

- Database of Long-Lived Cryptographic Keys
  - <draft-housley-saag-crypto-key-table-02.txt>
- Routing Authentication Using A Database of Long-Lived Cryptographic Keys
  - <draft-polk-saag-rtg-auth-keytable-03.txt>
- The former defines a conceptual model, the latter describes the model's application to routing protocols

Fundamental Concepts

- Manual key management is today's reality in routing protocols
  - Future key establishment protocols must co-exist with manual keying
- If Key establishment is performed in separate protocols, rather than a handshake in routing protocols, there is no perceptible difference to the routing protocol (RP).

Fundamental Concepts, II

- Modeled as a database or table of shared keys that are available to the routing protocols
  - Textual description of database entries is consistent with current operational practice
- The keytable is a conceptual resource that permits long-term key management to be separated from routing protocol design
  - Protocol designers can concentrate on session-specific key management (e.g., derivation of session keys, rollover, etc.) and cryptographic agility.

Non-Goals

- These documents do not impose any implementation requirements
  - If this conceptual model is adopted, they could be implemented with per-RP keytables or a single keytable.
- These documents do not specify an API.

Database

- Database is characterized as a table, with a row for each key
- Identifies 11 columns for the key and its attributes
- Describes rollover between long-lived keys

Database Columns (1 of 2)

- LocalKeyID
  - A 16-bit integer in hexadecimal, unique in the context of the database. The high order bit differentiates pairwise and group keys.
- PeerKeyID
  - For pairwise keys, the peerKeyID field is a 16 bit integer in hexadecimal provided by the peer or "unknown" if the peer has not yet provided this value.
  - For group keying, the PeerKeyID field is set to "group", which easily accommodates group keys generated by a third party.
- KDF
  - Indicates which key derivation function (KDF) is used to generate short-lived keys (or "none“ when the long-term key is used directly).
- KDFInputs
  - Used when supplemental public or private data is supplied to the KDF.
- AlgID
  - Indicates which cryptographic algorithm to be used with the security protocol.

Database Columns (2 of 2)

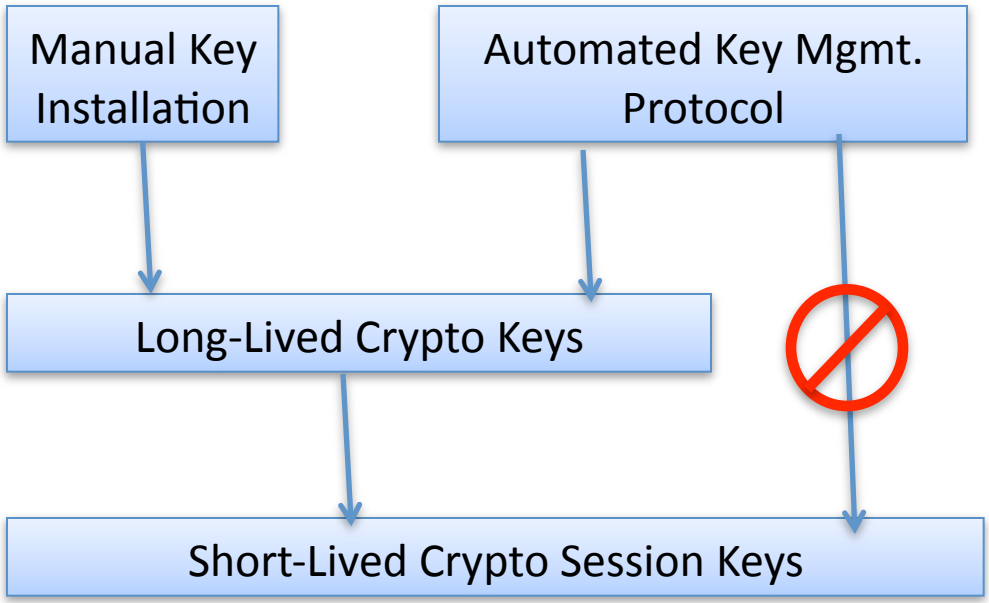
- Key
  - A hexadecimal string representing a ling-lived symmetric cryptographic key.
- KeyDirection
  - Indicates whether this key may be used for inbound traffic, outbound traffic, or both.
- NotBefore
  - Specifies the earliest date and time at which this key should be considered for use.
- NotAfter
  - Specifies the latest date and time at which this key should be considered for use.
- Peers
  - Identifies a peer system or set of peer systems
- Protocol
  - Identifies the security protocol where this key is to be used to provide cryptographic protection.

Consistency with Current RPs

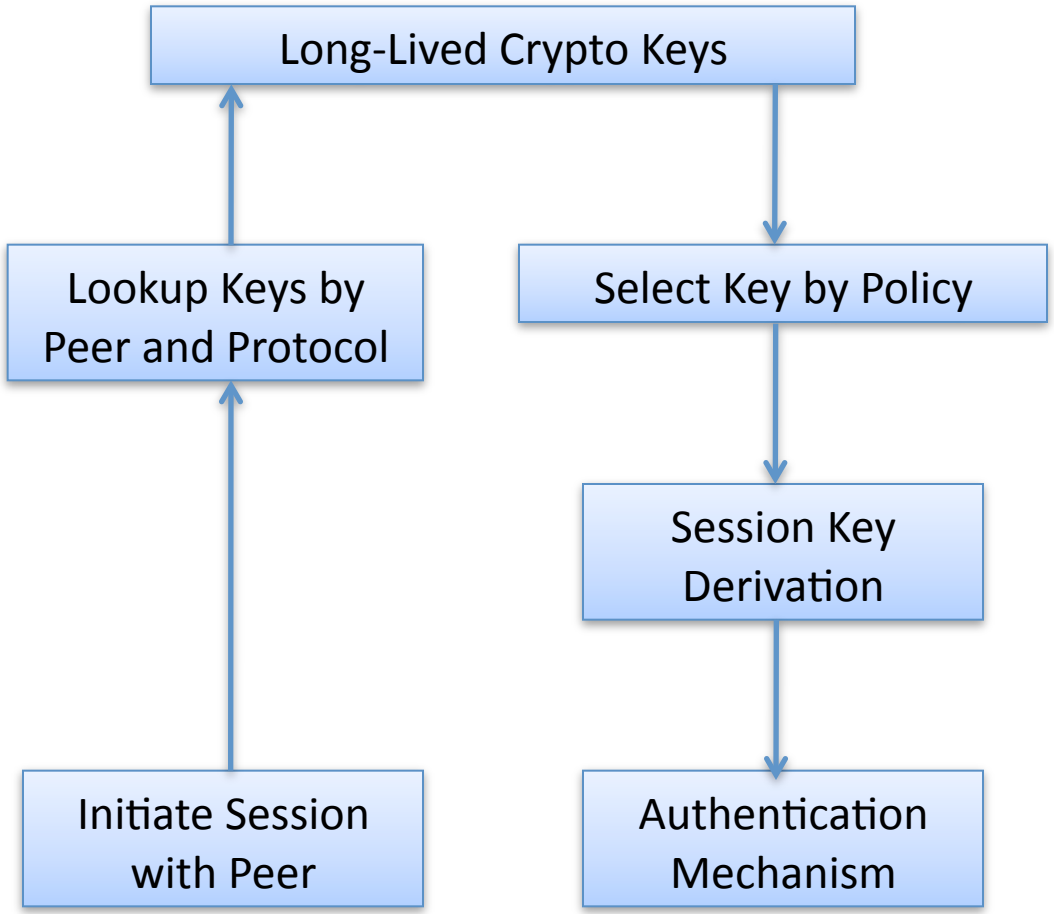
Based on draft-wei-karp-analysis-rp-sa-00:

- Current RPs use a subset of the key attributes in the keytable except OSPFv2
  - OSPFv2 specifies four time/direction attributes
    - {Key Start Accept, Key Start Generate, Key Stop Generate, Key Stop Accept}
  - the table supports two time attributes and one directional attribute
    - {KeyDirection, NotBefore, NotAfter}

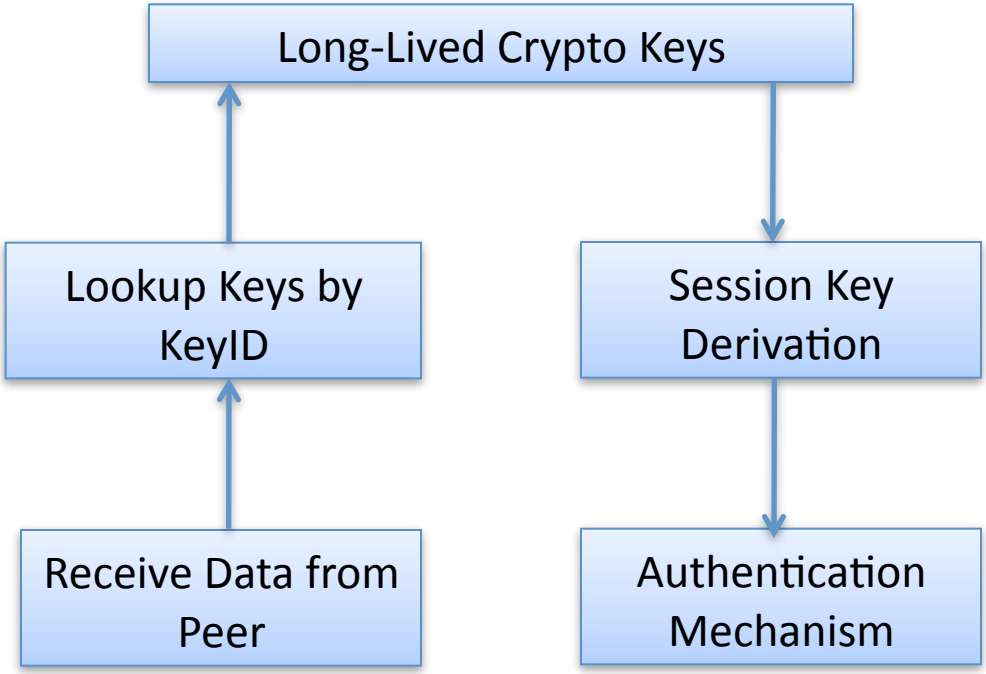
The Overall Model



Initiator's View



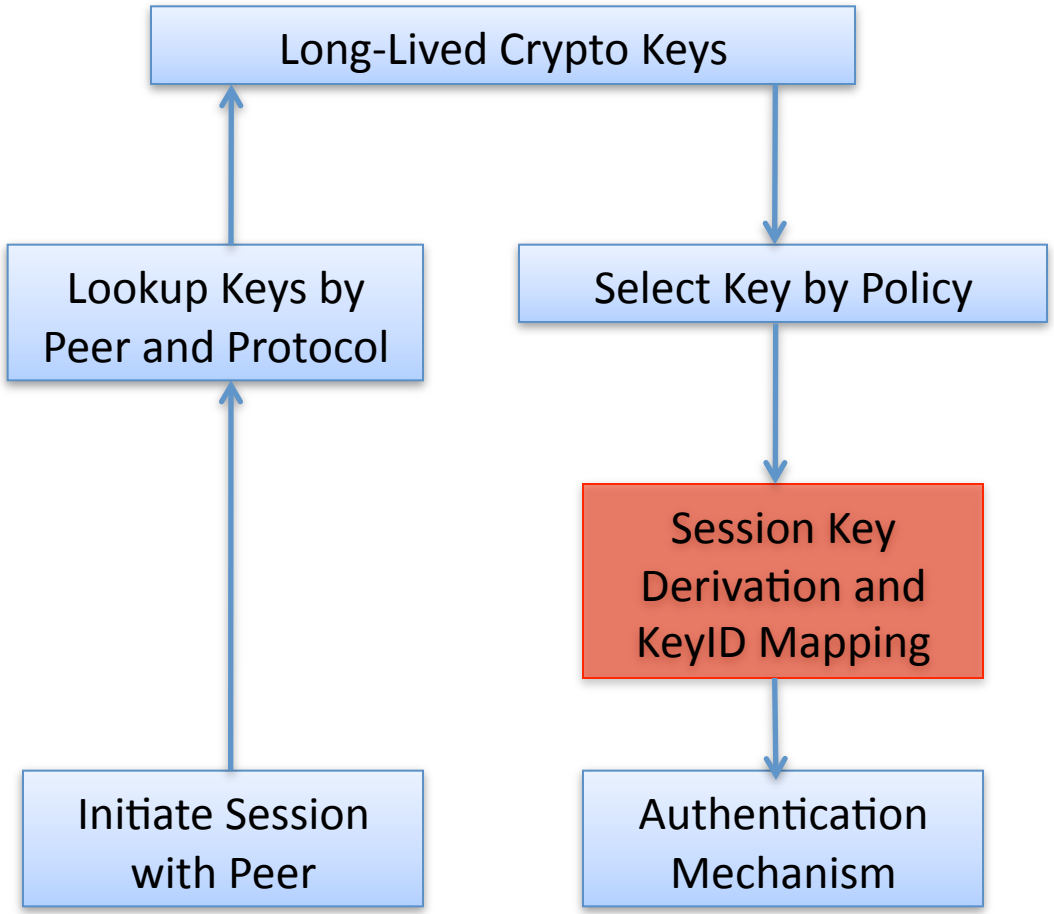
Receiver's View



KeyID Mapping

- Database specification mandates a 16-bit KeyID
- KeyID in the table may not be the KeyID used on the wire
  - Need to support more than just one security protocol
  - Allow translation to any needed format or size
  - Overlapping ranges may unnecessarily limit the total number of keys that can be maintained
- Mapping can resolve size mismatch and overcome overlapping range issues
  - Only applicable to local KeyID values
  - Peer's KeyIDs are not unique in the context of the table

Initiator's View with Mapping



Recent changes and TBDs

- Recent Changes
  - Softened text regarding automated key management in routing protocols to place out of scope rather than not expected to exist
    - If such mechanisms emerge, these protocols would not have any reason to make use of this database.
  - Added brief section on database maintenance to cover key removal
- TBDs for draft-polk-…
  - Simpler examples
  - Handling simultaneous open
  - Security Considerations

Questions?