KARP KMP-Using IKEv2 with TCP-AO

draft-chunduri-karp-using-ikev2-with-tcp-ao-01

Uma Chunduri, Albert Tian
Ericsson Inc.
Joe Touch
USC/ISI

IETF 83, Paris, France
March 26-30,2011

Using IKEv2 with TCP-AO

Goals:

Minimize impact on TCP-based Routing Protocols seeking KMP by integrating:

- IKEv2
- Proposed Gatekeeper module and
- TCP-AO's infrastructure (MKTs)

Minimize changes to IKEv2

- SA payload changes to negotiate RP (TCP-AO) SAs
  - With out impacting the current SA proposal/negotiation rules
  - And continuously leverage new IKEv2 features, e.g., pre-shared key only and yet secure authentication
- By adding an external Gatekeeper module
  - To support the mechanisms IKEv2 expects in IPsec, but are not part of TCP-AO

Minimize changes to TCP-AO

- No changes needed.

What is needed from TCP-AO's viewpoint:

- A way to utilize IKEv2-compatible keying
  - IKEv2 assumes IPsec manages SA timers, triggers new SA requests
  - TCP-AO assumes external key management, incl. timers and rekey initiation
  - Need separate key timers, rekey initiation ➔ Gatekeeper (GK) (see: *Ghostbusters*)

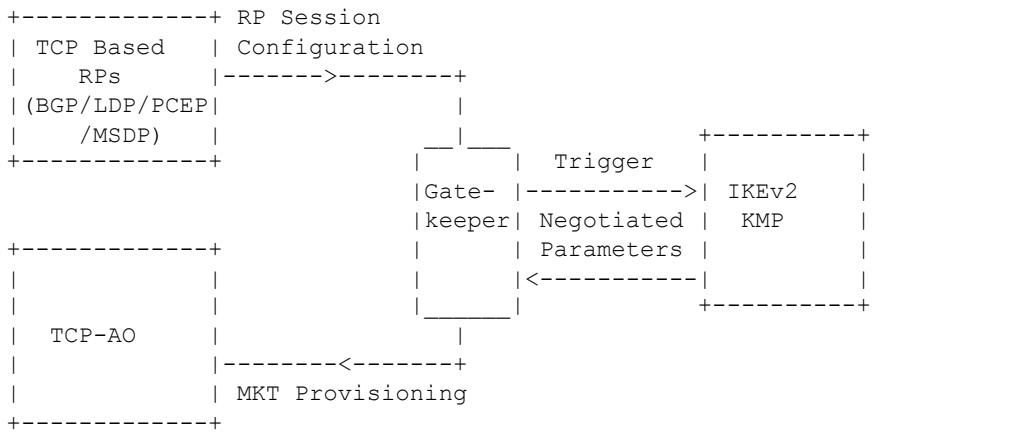
  - Transport-level differentiation of multisession BGP sessions
    - Socket pair must be unique (Unique MKT in AO)
    - Currently use different IP addresses
    - Use different source ports => need code somewhere (BGP source, link library, OS)

- Result
  - IKEv2 generates keys and parameters
  - GK triggers IKEv2 initial and rekeying, inserts info into TCP-AO, revokes keys
  - TCP-AO implements transport authentication based on given info.

What is needed from IKEv2's viewpoint:

- Additional transforms to Security Association (SA) Payload for TCP-based routing protocol SA (TCP-AO SA)
  - Extensions required listed in the draft (non IPsec DOI)
  - No new port required
    - Peer auth mechanisms/messages are not changing
    - Leaves the existing tunneling mechanisms intact
- Simplified Traffic Selectors
- More details in the draft

Proposal:



KARP KMP: Using IKEv2 with TCP-AO

draft-chunduri-karp-using-ikev2-with-tcp-ao-01 captures:

- TCP-based RP interface to GK
- KMP interface to GK
- TCP-AO interface to GK
  (more details in the draft)

This proposal avoids:

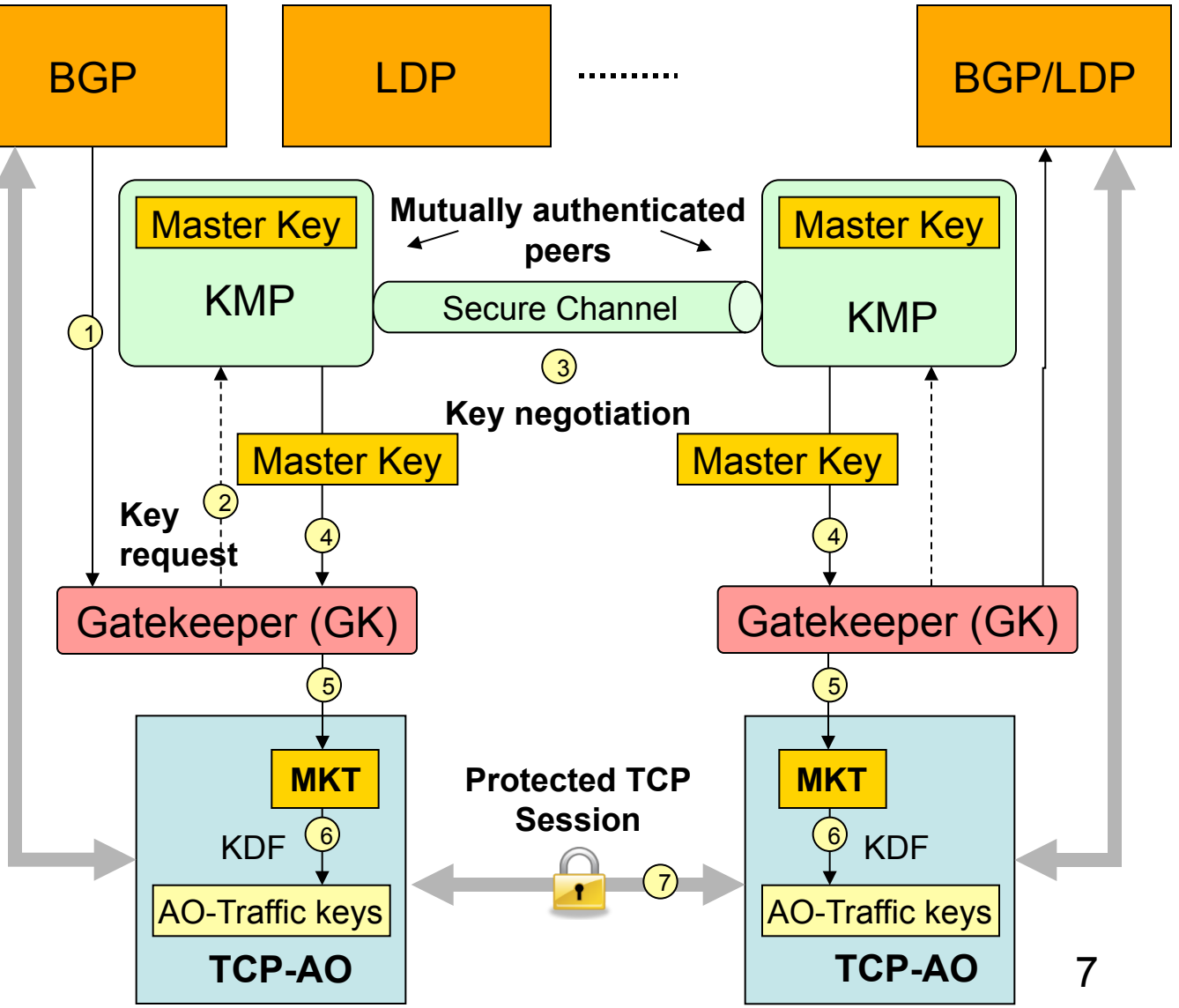
- All routing protocols need to trigger KMP to get the SA
- All routing protocols need to maintain the SA with the lifetime
- and rekey when lifetime expires

(Essentially complete SA management at each RP level)

Using IKEv2 with TCP-AO (cont.)

Solution

1. BGP/LDP sets configured Auth/KDF/ lifetime info and initiate TCP connection
2. GK triggers KMP (IKEv2)
3. IKEv2 negotiate Master key
4. Master keys added to GK
5. GK converts IKEv2 keys into MKTs; revokes and/ or retriggers IKE as needed
6. Use KDF to derive TCP-AO traffic-keys
7. TCP session protected



Advantages:

- No TCP based routing protocol changes to do SA management
  - Transparent to keys, key management and KMP
  - Configuration can be *similar* to manual keys with TCP-AO
- No changes to TCP-AO (5925)
- Utilizes integrated extensibility in IKEv2 (5996) to negotiate non-IPsec SA for RPs
  - Simplified configuration for RPs
- Gatekeeper isolates how TCP-AO mimics IPsec to IKEv2
  - Manages the state/timers that IKEv2 expects IPsec to manage

Other Discussions (detailed in Appendix)

BGP Multi-session requirements (optional)

- Transport level differentiation: stems from the possible need for different security services
- possible ways to address

Peer AUTH methods available

- Brief discussion on all methods suitable for RPs

Questions & Comments?

Thank You!