Proposed IPsec HA Cluster Protocol

Raj Singh

IPsecME WG

IETF-78, Maastricht

The background

- Kalyani presented IKEv2 message ID sync problem in HA cluster on the IPsecME WG mailing list
- Yoav wrote a good draft summarizing all problems in IPsec HA cluster, soon to be an RFC
- IPsecME WG created an HA design team to come up with IPsec HA cluster solution draft
- HA design team presents draft-kagarigi-ikev2-windowssync-04 as team’s output for comments

Intro - IPsec HA cluster solution draft

- This draft solves main problems of “IPsec Cluster Problem Statement” and gives implementation advices for others
- The problem solved are:
  - IKE Message ID synchronization – Newly-active member asks message ID values from the peer
  - IPsec Replay Counter synchronization – Newly-active member tells peer the bumped-up outbound replay counter value and vice versa
- The draft can also be used to sync IKEv2 message in other scenarios where they are mismatched between IKEv2 endpoints

The basic scenario … 1

- Peer establishes IKEv2/IPsec session with active member of hot standby cluster
- Active member syncs IKEv2/IPsec SA states to standby member periodically
- A “failover” event occurs in cluster
- The standby member takes over and becomes the active member
- It takes some time; the IKEv2/IPsec counters on newly-active member are not the same as old active member
- So, now IKEv2/IPsec mismatched between peer and newly-active member

The basic scenario … 2

- The peer is unaware of “failover” in cluster
- The peer keeps sending IKEv2 requests and IPsec packets to the cluster as normal
  - Also, the newly-active member is not aware of un-acknowledged request sent by previous active member
- The peer keeps on re-transmitting old requests and then gives up, tearing down IKEv2/IPsec SA
- The newly-active member doesn’t know the exact outgoing/incoming IPsec replay counter; this can lead to replay attack by sending old counter

The solution … 1

- Peer and active member negotiate the ability to sync SA counters in their original IKE_AUTH exchange using a SYNC_SA_COUNTER_INFO_SUPPORTED notification
- The active member tells the standby member its ability to do SA sync support when the IKEv2/IPsec session is established
- Then, a “failover” event occurs in the cluster
- The standby member takes over and becomes the active member

The solution … 2

- The newly-active member sends a SYNC_SA_COUNTER_INFO authenticated request with the special message ID of 0 asking the peer for its IKE message IDs, and telling the peer its outbound IPsec replay counters
- The peer sends a SYNC_SA_COUNTER_INFO authenticated response, syncing the IKE and first few IPsec SA counters

The solution … 3

- This request is sent when the newly-active member had to send new IKE/IPsec packet, or when it receives a “bad” IKE/ IPsec packet
- In case there are many IPsec SAs to sync, they can synced later using synced-up IKE message id after first SYNC_SA_COUNTER_INFO exchange

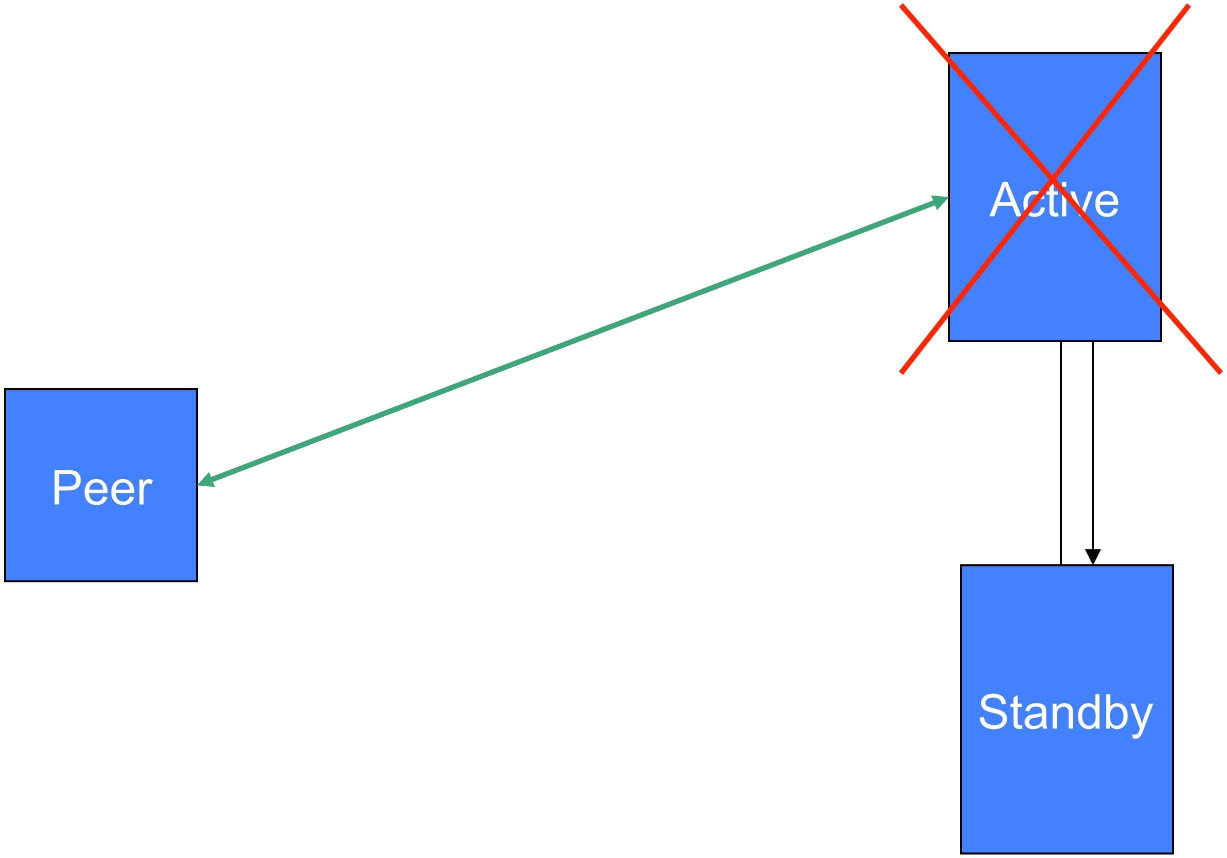
Security considerations

- The draft is as secure as IKEv2. However, there are two kind of DoS attacks, each of which has solutions:
  - Replay of SA sync request
    - Perform rate limiting at the peer
  - Replay of SA sync response
    - The nonce in SA sync request avoids this

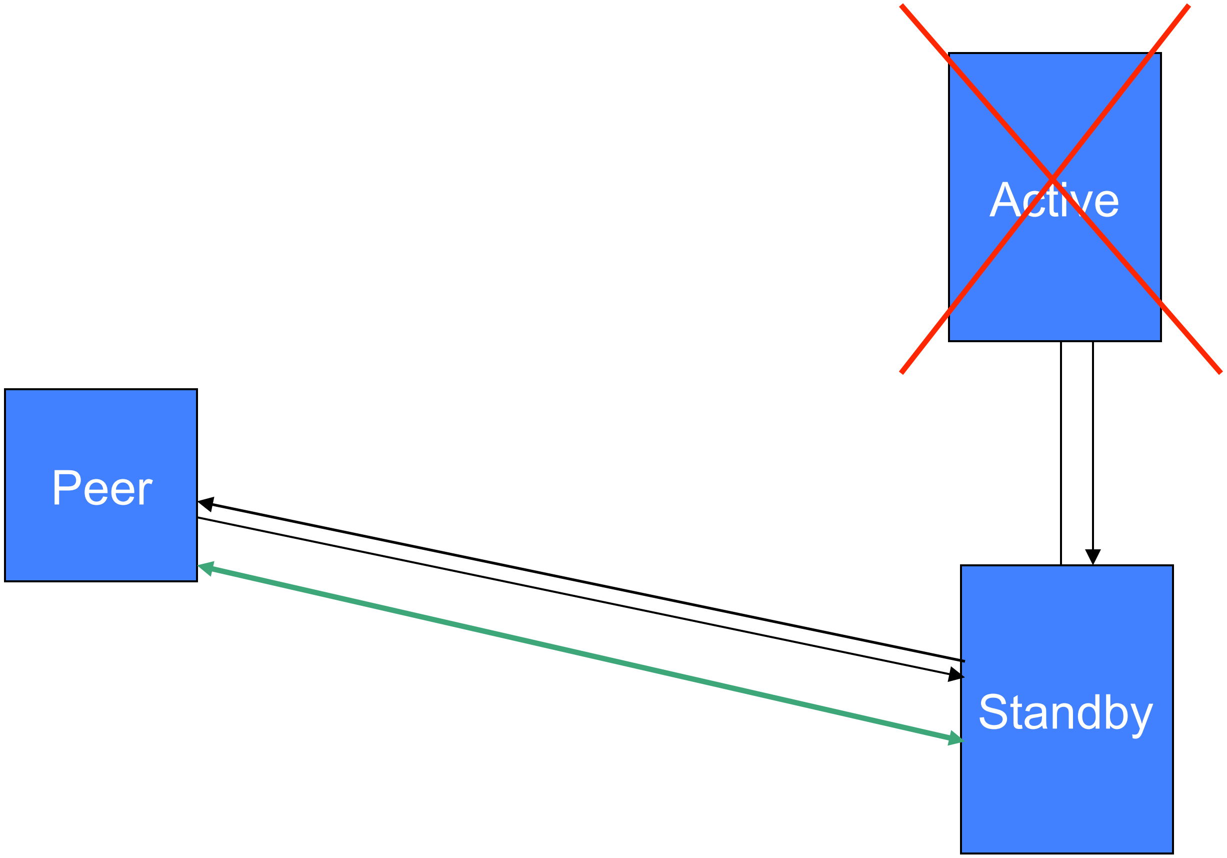
Interaction with other IPsec protocols

- Session Resumption
  - Client and peer re-establish the session instead of full session establishment from scratch
  - Mutually exclusive with HA with SA counter sync
  - Helps loosely coupled HA cluster
- Redirect
  - Used during session setup and scheduled maintenance
  - Mutually exclusive with HA with SA counter sync
- Crash Detection
  - Solves the similar problem where peer detects cluster member has crashed
  - Mutually exclusive with HA with SA counter sync

Protocol animation



Protocol animation


Active
Peer
Standby