TESLA Overview

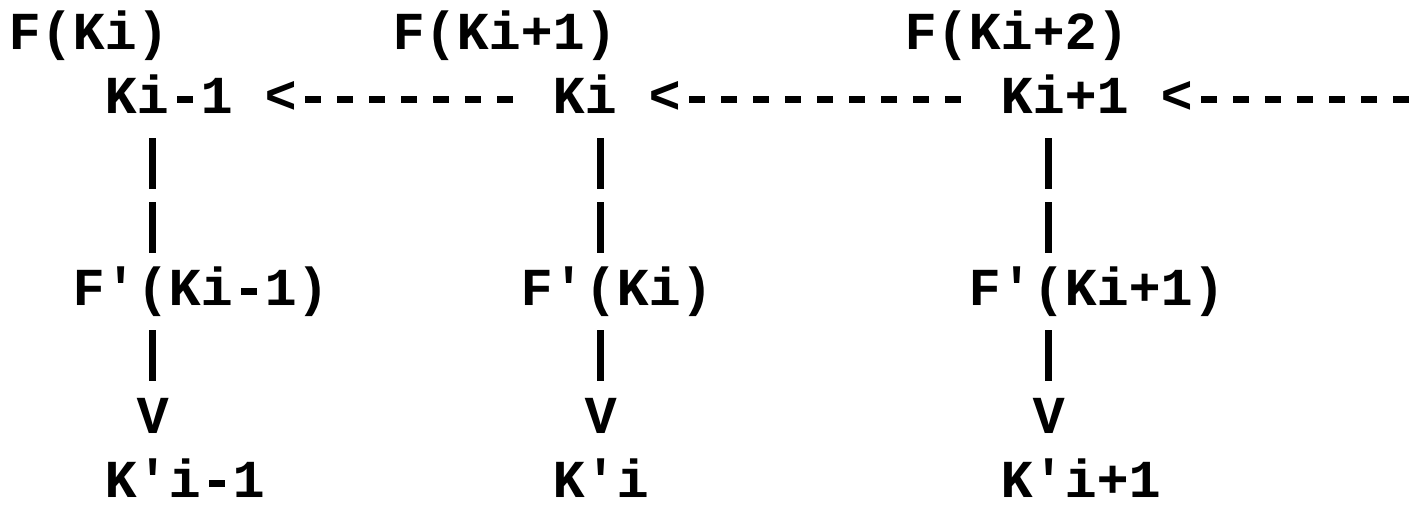
Overview

- TESLA developed by Perrig, Canetti, et. al. as an efficient source authentication transform
- Seems to have advantages over other MAC-bases source authentication schemes
- It is destined to be used by MESP
- There are some complexity issues with TESLA
- Need to consider if this is something that belongs in the kernel

TESLA Properties

- High guarantee of source authenticity for multicast groups
- Does not provide non-repudiation
- Robust against loss and re-ordering
- Low overhead of 12-20 bytes/packet
- Delayed disclosure & receiver buffering
- No sender buffering

Deriving Authentication Keys

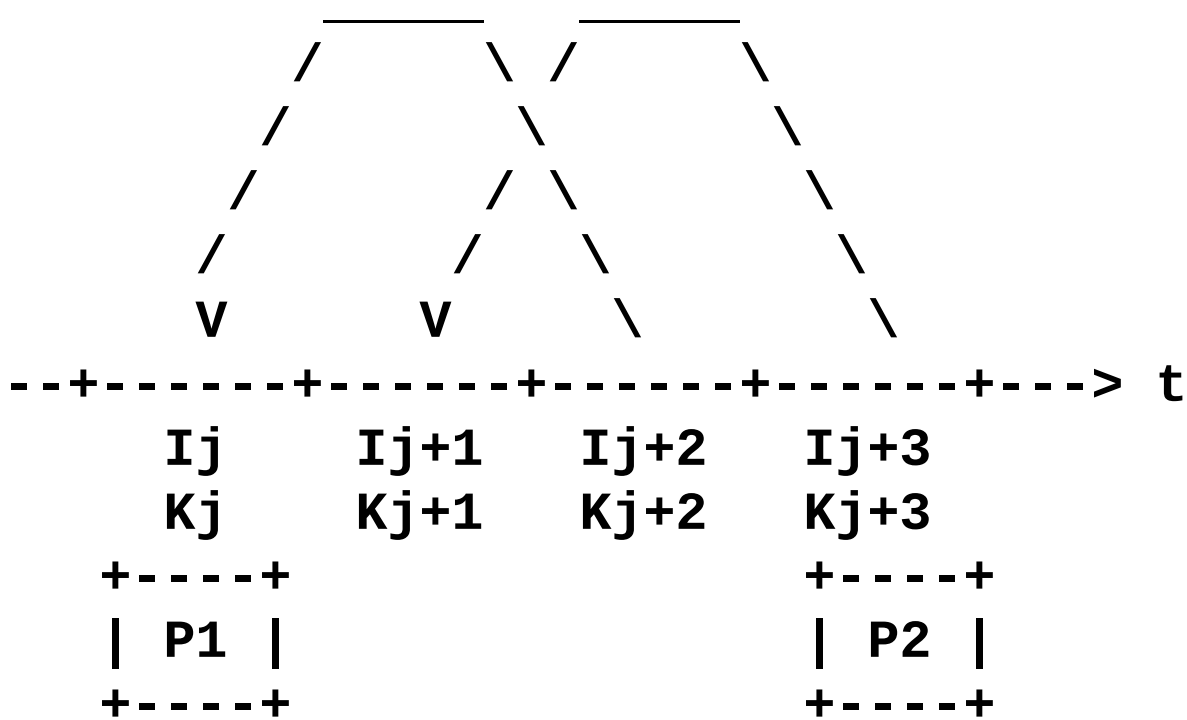


Based on an old scheme: Lamport's One-Way Hash Chain (1981) and S/KEY (RFC 1760). HMAC-SHA1 is just one type of one-way function that can be used.

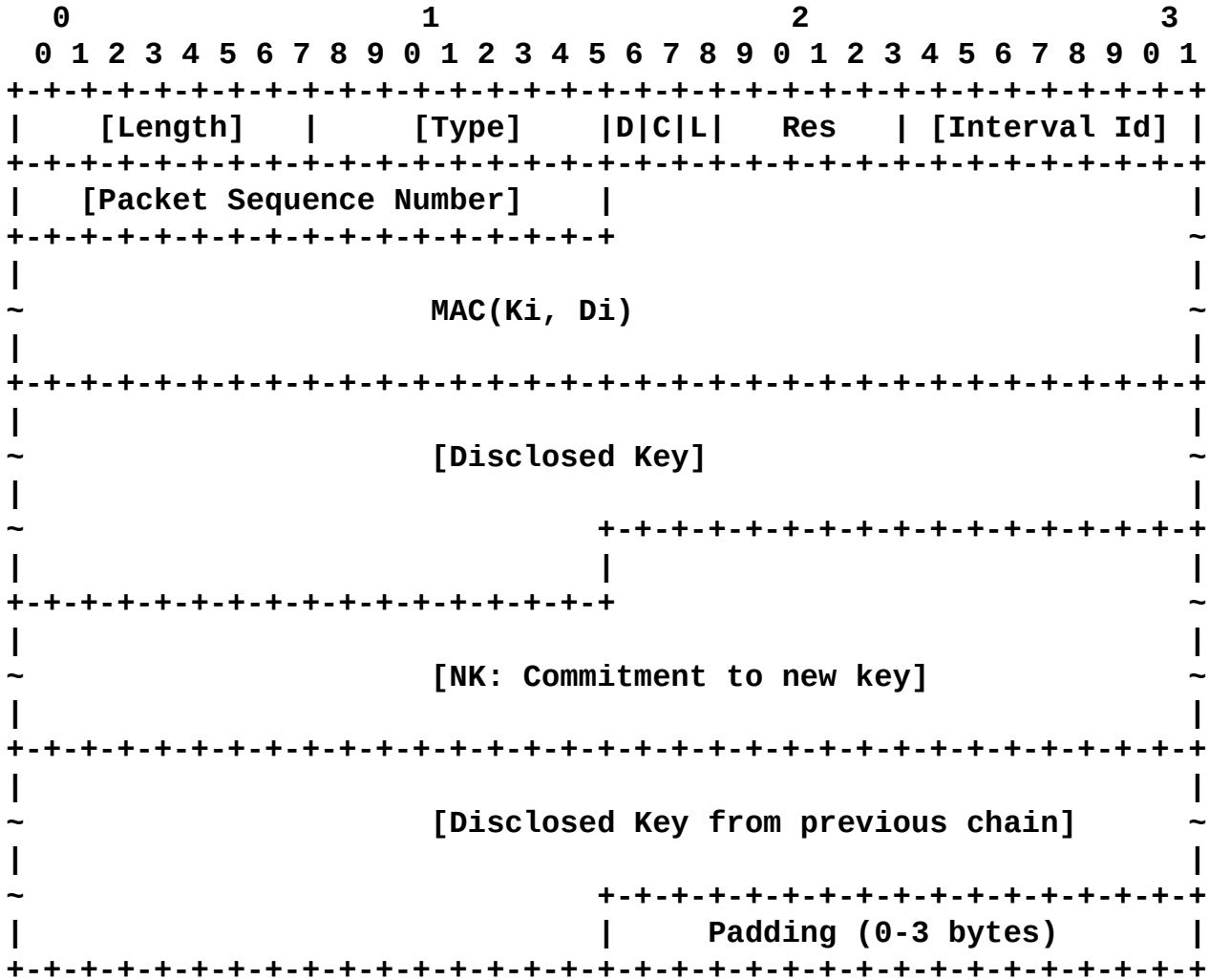
Based on Hashed Key Chain

- $K_i = HMAC(K_{i-1}, 1)$, $K_0 = K$
  - Sender selects chain length N
  - Precomputes chain from N-1 to zero
- K is digitally signed by sender
  - Disseminated e.g. by key management
  - One sig per arbitrarily long “key chain”
- $K_i' = HMAC(K_i, 0)$ is HMAC key for packet
- K_i' used for all packets in interval i

TESLA Packet Processing



TESLA Packet Format



TESLA Issues

- Time synch
  - Packets received after key disclosure
  - Receives with vastly different sender RTTs
- Receiver buffering
  - Problematic in the kernel
- Others?