ALTA

Asymmetric Loss-Tolerant Authentication

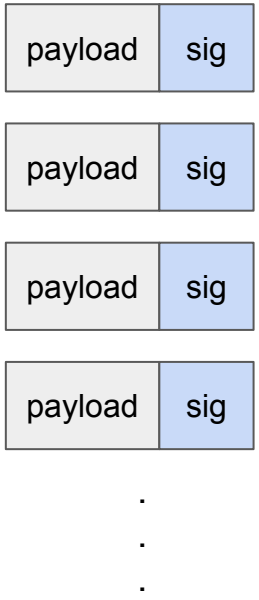
Kyle Rose <krose@krose.org>
Jake Holland <jakeholland.net@gmail.com>
Akamai Technologies

Problem

Authenticating datagrams

- Payloads have a deadline
- Many receivers
- Datagrams are lossy
- Rexmits not appropriate

Signature Per Packet



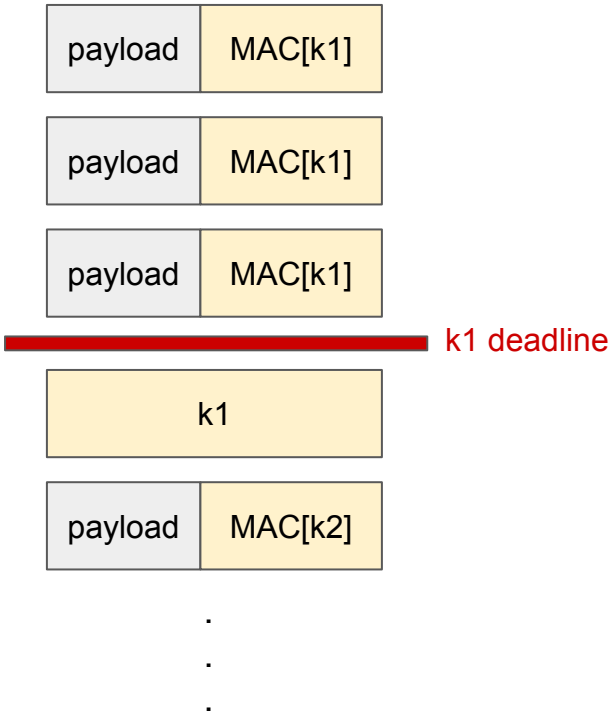
Pros

- Can verify all received packets

Cons

- CPU intensive w/o dedicated HW

TESLA



Release symmetric k1 after all packets using k1 have been delivered

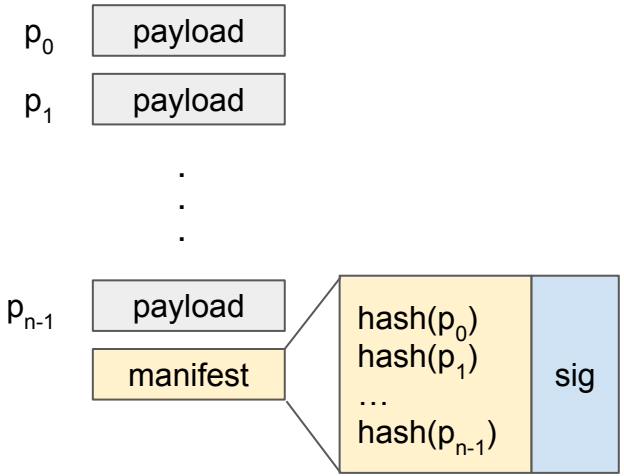
Pros

- Symmetric auth is cheap

Cons

- Requires some weak clock sync (still some delay attacks)
- All bets are off once key is released

Signed Manifest



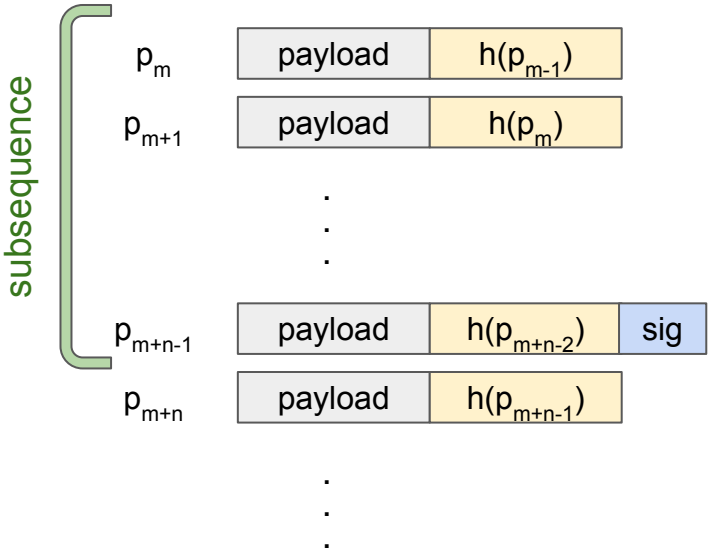
Pros

- Lots of fast hashes
- Small number of slow signatures

Cons

- What if you lose the manifest?
- Fate of data disconnected from authentication info

Chained Integrity



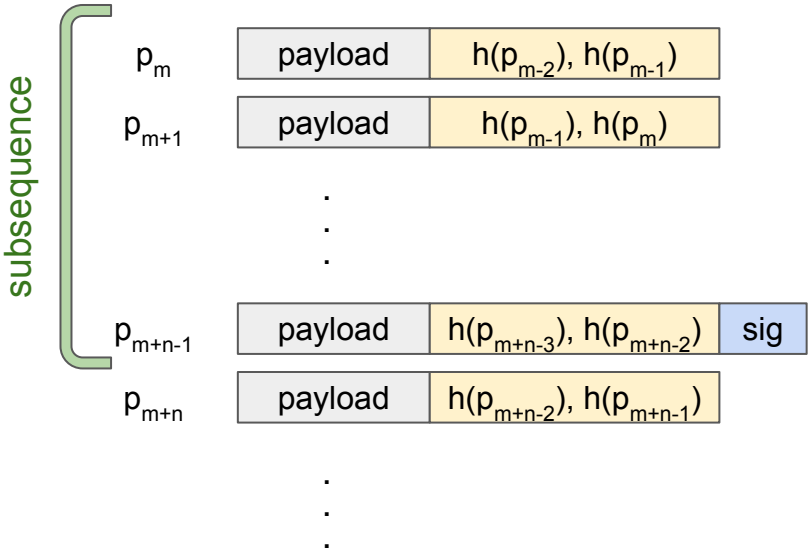
Pros

- Sparse signatures
- Tolerance for signature loss
- Fate of data connected to auth info

Cons

- Every loss breaks the chain

Redundant Integrity



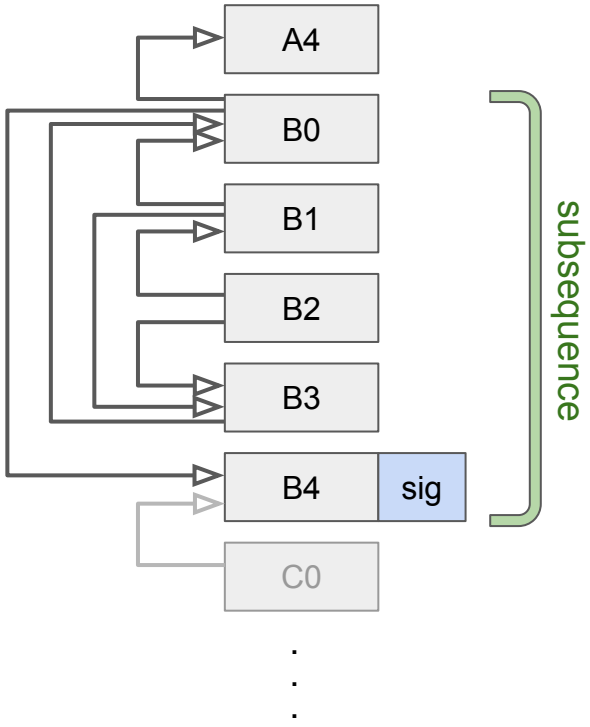
Pros

- Two chances to get a packet hash

Cons

- Loss rate p → lose a subsequence with probability p^2
- Maybe more often if loss is bursty!

Golle and Modadugu (2001)



A DAG of hashes, with periodic signatures

Pros

- Also two chances to get each packet hash, but better distributed

Cons

- Complicated construction (moreso even than the diagram)
- Variable number of hashes per packet (up to 5)

Key Properties

- Optimal resistance to bursty packet loss
- Tolerance for signature loss

Next Steps

- Running code
  - Will be made public soon!
- Making design choices: opacity vs. overhead
- Fleshing out the draft
  - https://github.com/squarooticus/draft-alta

Questions?