IEEE 1722a
Version 1 headers
0.3

Dave Olsen
dave.olsen@harman.com

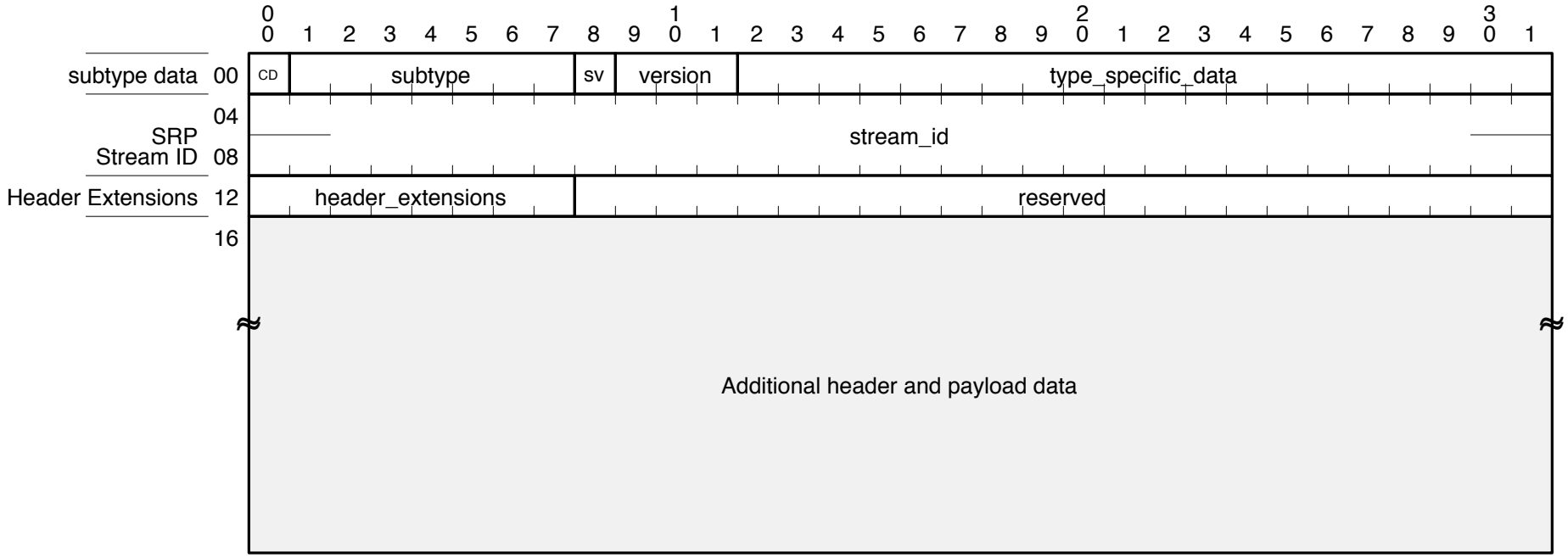
Overview

- We need a method to secure data encapsulated within a 1722 stream
  - This could be control or Audio/Video stream data
- Two methods have been requested, these methods may be used separately or together
  - Encryption
  - Packet Signing
- This is NOT related to content-protection (HDCP/DTCP)
- We are not security experts and do not want to invent anything
- We would prefer to make use of methods described in IEEE Standards 1363 and 1363a

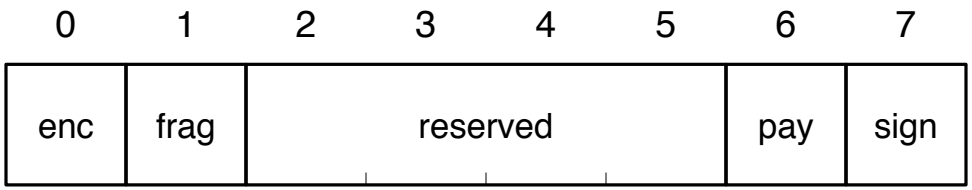
Version 1 Header

- Defines optional header extensions
  - Initially Signed Packet and Encrypted Packet
- A 1722a stream may contain 0 – 8 header extensions
- All packets in a stream shall use identical header extensions
- Header extensions shall always appear in a fixed order in the packet
- Offset to each header.
- Signed Packet header shall always be last
- If no header extensions are used then streams shall use version 0 headers
- All future header extensions shall be quadlet based

Version 1 Header



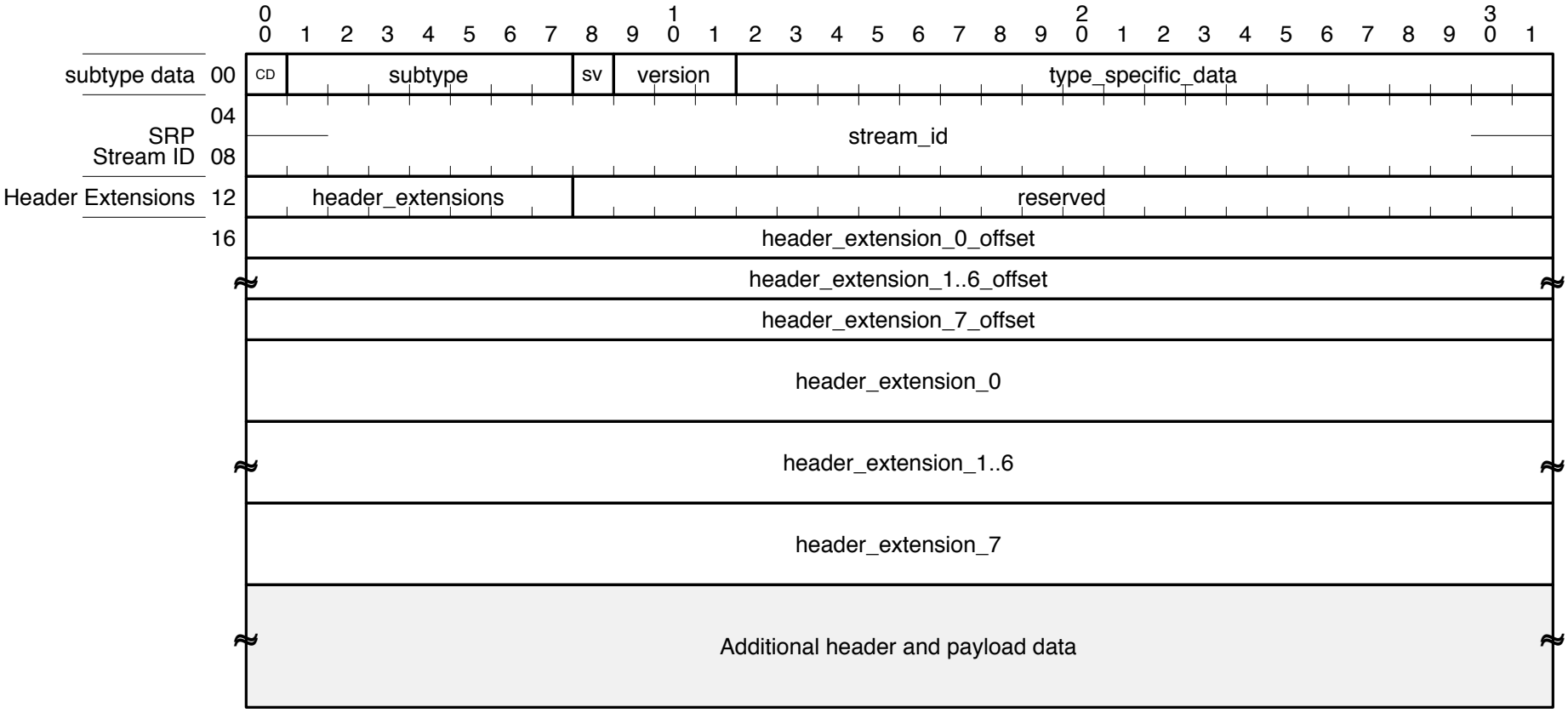
Header Extensions Detail



header_extensions

Enc – Encryption Header
Frag – Fragmentation Header
Pay –Payload
Sign – Signed Packet Header

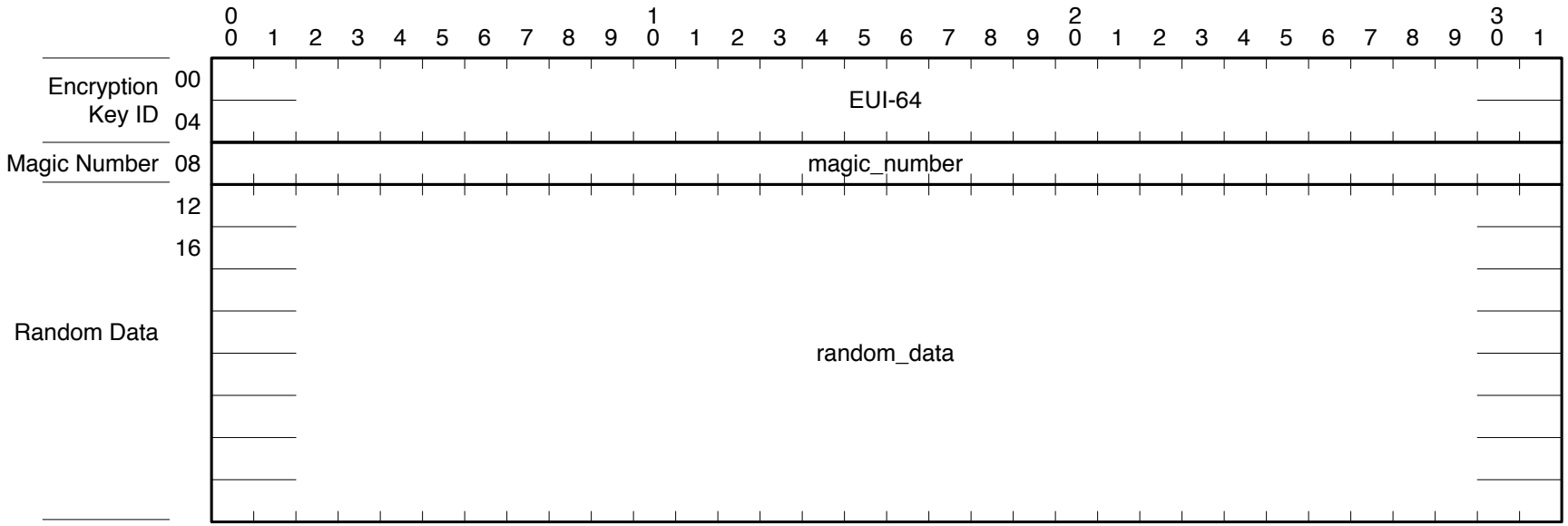
Version 1 Header with Extensions



Version 1 Header with Extensions

- Header extension are in a fixed order
  - Extension 0 through extension 7
  - We need to consider the proper order since encryption should be the first and signed should be the last.
  - Payload is the standard 1722 data area, with the ability to map this data prior to the packet signature
- For each optional header extension an offset is included to the actual header extension (32 bits)
- Note: we may need to move the StreamID to after the header extensions in order for the StreamID to be encrypted, but this causes structural problems.

Encryption Header



Encrypted Packet Question

- Encryption begins with the Magic Number and all contents of the packet are then encrypted.
- All sections of the packet are encrypted, including all subsequent header extensions
- Do we need an encryption trailer to guarantee packet integrity?
- The goal is to use IEEE Standard 1363 and 1363a to define encryption methods

Fragmentation Header

REJECTED

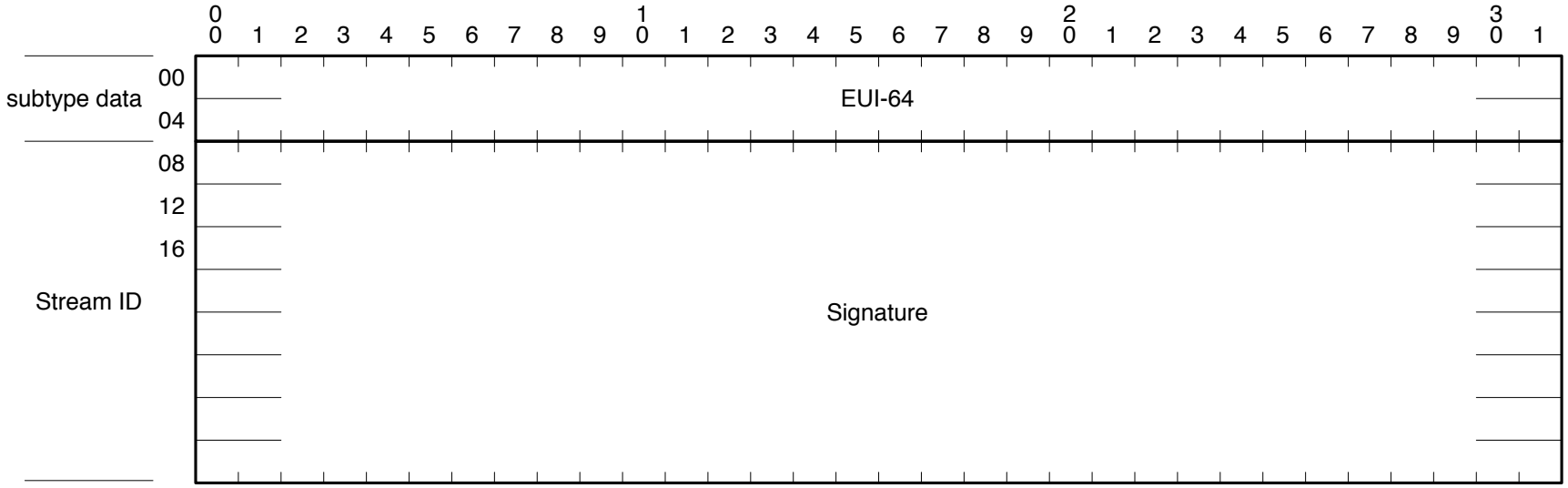
Fragmentation Header Questions

- This header extension is not related to security
- Allows transmission of management objects that are greater than 1 packet length

Payload

- The payload is the standard 1722a payload.
  - This could be data or control
  - This allows the payload to be mapped before the packet signature
  - The Payload extension should only be used in conjunction with the Signature extension

Signed Packet Header



Signed Packet Questions

- Signature includes all packet data from DA to end of packet not including the Ethernet CRC
- Do we need a trailer for Signed Packets?
  - Does the new payload extension fill this requirement, by moving the payload before the Signed packet extension
- The packet signature should be at the end of the packets, especially if the packet is a data packet. Having to store the entire packet and then calculate the signature would increase latency.
- The goal is to use IEEE Standard 1363 and 1363a to define packet signing methods