

Limits on Sending and Processing IPv6 Extensions Headers

draft-ietf-6man-eh-limits-02

Tom Herbert <tom@sipanda.io>

Changes in -02

- Disclaimer that mechanisms to determine when a limit may be exceeded are out of scope for draft
- Reference HBH Processing draft, particularly for handling high order two bits for unknown option
- Adding motivation for allowing an intermediate node to stop HBH option processing and forward the packet for an unknown option with first two bits of type not 00
- Edits

Motivation: Default # non-padding opts

- Default for supported number of options: 8
 - Only thirteen non-deprecated options in history
 - Rate of adding new options, especially could be used with others, is low
 - Anecdotal evidence that hardware like P4 could handle eight options (hardware parallelism, CAM)
 - Conclusion: 8 is sufficiently future-proof

Motivation: Padding options defaults

- Default: Maximum seven bytes, one padding options (PAD1, or PADN for 2-7 bytes)
 - Next field alignment is predominant use of padding
 - Defaults allow up to 8 byte alignment
 - No practical reason why source host would use more than one consecutive padding options

Motivation: v6 header chain length

- Default: Must handle 128 byte chain length, sixty-four bytes of extension headers
 - Smaller parsing buffer sizes on devices are atypical
 - Many device process encapsulation so they already need a larger parsing buffer than the minimum
 - Larger default size? Already other limiters (e.g. RFC 9000, QUIC, limits EH to 32 bytes for minimum IPv6 MTU)

Next steps

Requesting WGLC

Thank you!