Limits on Sending and Processing IPv6 Extensions Headers

draft-ietf-6man-eh-limits-00

Tom Herbert <tom@sipanda.io>

Overview

- Specify a set of limits that may be applied to various aspects for sending and receiving extension header
- Goals
  - Define practical limits to promote efficient and feasible implementation
  - Don’t restrict functionality in extension headers to the point that they no longer useful!

Draft status

- Became WG item since last IETF 112
- Most discussion thus far seems to be focussed on the ramifications, particularly whether establishing limits would curtail future uses

Types of limits

- Limits on processing of extension headers
- Limits on length of the IPv6 header chain

Ramifications

- With one exception, the limits described in the draft are “soft” limits
  - Senders can exceed limits if they have knowledge about the path
  - Receiver limits are optional (both at intermediate nodes and hosts)
- Suggested defaults for limits are expected to be reasonable, but not overly constraining

The one hard limit is draft

- Intermediate nodes MUST correctly process IPv6 packets with up to **64** bytes of EH
- Per RFC8200, intermediate nodes only have to process IPv6 header (may ignore HBH)
- So this requirement primarily addresses intermediate nodes that have assumed the implicit requirement to access L4 headers

Why 64 bytes?

- Assumes devices can support at least 128 byte parsing buffer
- 64 = 128 - 16(L2 hdr) - 40(v6 hdr) - 8(L4 hdr)
- Anecdotal information that 128 byte parsing buffer is common in routers
- For instance, many router implementations support encapsulation protocols that require more than minimal parsing buffer

Why 64 bytes?

- JAMES (draft-vyncke-v6ops-james-00) is new round of measurements of EH (the authors mentioned what will look into supported EH size)
- Also, note that RFC9000 (QUIC) effectively limits IPv6 EH length to 32 bytes

Next steps

- Please continue discussion on the mailing list
- Need some reference implementation

Thank you!