IPv6 Minimum Path MTU
Hop-by-Hop Option

Bob Hinden
Gorry Fairhurst

November 2020
IETF109 On-Line
Background

- Current RFC8201 PMTUD isn’t working well.

- This hop-by-hop option came from the idea that it will be more reliable for the Destination to send Path MTU feedback to the Source.
  - Better trust relationship than RFC8201 PMTUD.

- It may not work in all places [RF7872] etc., but we suggest it can help some places.
Path MTU HBH Option

<table>
<thead>
<tr>
<th>Option Type</th>
<th>Option Data Len</th>
<th>Option Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>BB</td>
<td>00</td>
<td>Skip over this option and continue processing.</td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>Option data can change en-route to the packet's final destination.</td>
</tr>
</tbody>
</table>

TTTTT 10000 Option Type assigned from IANA [IANA-HBH].

Length: 4 Note the size of the each value field in Option Data field supports Path MTU values from 0 to 65,535 octets.

Min-PMTU: n 16-bits. The minimum PMTU in octets, reflecting the smallest link MTU that the packet experienced across the path. This is called the Reported PMTU. A value less than the IPv6 minimum link MTU [RFC8200] should be ignored.

Rtn-PMTU: n 15-bits. The returned minimum PMTU, carrying the 15 most significant bits of the latest received Min-PMTU field. The value zero means that no Reported MTU is being returned.

R n 1-bit. R-Flag. Set by the source to signal that the destination should include the received Reported PMTU in Rtn-PMTU field.
Changes Since IETF108 (1)

- Current draft: draft-ietf-6man-mtu-option-04
- Rewrite terminology e.g. in 6.3:
  - Method aligned to the way common API's send/receive HBH option data.
  - Added reference to DPLPMTUD - now RFC8899 and clarified upper layer usage.
  - Fixes typos.
Changes Since IETF108 (2)

- Defines two ways to use this:
  - Direct use of the Rtn-PMTU Value by IPv6
  - Using as a Hint for Probing by (D)PLPMTUD

- Added validating before use of HBH option data

  Upper Layer discards any received packet when the packet validation fails..., the Upper Layer MUST also discard the associated Option Data

- Completed security considerations section. i.e.:
  - Network Layer Host Processing
  - Validating use of the Option Data
  - Impact of Middleboxes
Experiment Probes

IPv6 Probes currently deployed

- 1 DC in USA
- 1 Home in USA (Comcast)
- 1 in UK Academic Core (JANET NOC)
- 1 in Academic Edge at Aberdeen (JANET)
- 1 in Home in UK

Looking for people who have IPv6 *AND* a larger MTU!
- Please contact us

20 November 2020
Some Results (Internet Paths)

<table>
<thead>
<tr>
<th>Outcome</th>
<th>IPv4 (0.5M targets)*</th>
<th>IPv6 (0.3M targets)*</th>
</tr>
</thead>
<tbody>
<tr>
<td>PMTU &gt;= 1500 B</td>
<td>~95.7% of paths</td>
<td>~96.5% of paths</td>
</tr>
<tr>
<td>BROKEN – PTB received but no PMTU discovered</td>
<td>.13%</td>
<td>0.07%</td>
</tr>
<tr>
<td>PMTUD works when needed (PTB received)</td>
<td>0.7%</td>
<td>0.05%</td>
</tr>
<tr>
<td>PMTUD Fails, but DPLPMTUD works</td>
<td>3.3%</td>
<td>3.3%</td>
</tr>
<tr>
<td></td>
<td>(with PTB info for 0.07% of cases)</td>
<td>(&lt;0.01% useful PTB info)</td>
</tr>
</tbody>
</table>

* Fraction of probe packets for usable paths to target IPv4/IPv6 web servers

- Around 96% of all paths we tested supported an MTU of 1500B.
- IPv6 PMTUD isn’t as good as we originally hoped 😞.
- DPLPMTUD (RFC8899) would help 😊.
# Larger Internet PMTU?

<table>
<thead>
<tr>
<th>Advertised MTU</th>
<th>Exchange Points*</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTU default</td>
<td>312</td>
<td>38%</td>
</tr>
<tr>
<td>MTU 1500B</td>
<td>420</td>
<td>51%</td>
</tr>
<tr>
<td>MTU 1501-9KB</td>
<td>4</td>
<td>0.5%</td>
</tr>
<tr>
<td>MTU &gt;= 9KB</td>
<td>58</td>
<td>7%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>812</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>

* Fraction of probe packets for usable paths to target IPv4/IPv6 web servers

Peeringdb interconnection database  
(not ISP data)  
(not direct peering to the content edge)
EH Support?

- 30% of paths to supported EH (measurements to DNS servers)*
  * Presented IETF-108 in 6MAN

- Only two home networks
  - One worked from a home network (UK).
  - One did not work (Comcast), who said:
    “The routers punt extension headers to control plane. Control plane bandwidth is limited and can be impacted if excess transit extension header traffic is allowed.” In current implementations, … it is good to filter transit packets that contain extension headers to protect the control plane capacity.
  - Most IPv6 Firewalls by default filter packets with EH.
    See draft-ietf-v6ops-ipv6-ehs-packet-drops-03

- We plan to look at edge network support by next IETF

20 November 2020
Next Steps

- We would love people to implement in routers and hosts!
- We’ll be collecting new data for March – if interested talk to us.
- Please read this version!
QUESTIONS / COMMENTS?