UDP Options

IETF 108 – July 2020

Joe Touch
strayalpha.com
-09 update (pending)

• Changes
  – FRAG (integrates prev. FRAG+LITE)
  – ACS to CRC32c (from IETF 106)
  – Fix OCS pseudoheader (from IETF 106)

• Additions
  – UNSAFE option
  – Deprecate RFC 6081 as flawed
Summary of option proc. rules

• On option **failure** -> halt option processing
  – Only options indicating format failure
    • OCS
    • FRAG
  – Zero-len user data received (per fragment) if halt

• On option **unknown** -> halt option processing
  – Only (and all) UNSAFE suboptions
  – User data (if any) received if halt
  – MUST use **before** FRAG to hide user data
New FRAG option

• Used ONLY with zero-len user data
  – I.e., requires equivalent LITE+FRAG behavior

• Allows single fragment packet
  – Enables pre and post-frag options
  – Enables use with UNSAFE to hide user data for unknown options

• Retains “zero-copy” approach
  – Allows reassembly without recopy of bulk of fragment
New FRAG formats

- **Terminal:**

```
+--------+--------+--------+--------+
| Kind=4 | Len=12 |      Offset     |
|        |        |                |
+--------+--------+--------+--------+
|          Identification          |
+--------+--------+--------+--------+
|  Frag. Offset   Reassy. Checksum|    |
+--------+--------+--------+--------+
```

- **Non-terminal:**

```
+--------+--------+--------+--------+
| Kind=4 | Len=10 |      Offset     |
|        |        |                |
+--------+--------+--------+--------+
|          Identification          |
+--------+--------+--------+--------+
|  Frag. Offset   |
+--------+--------+--------+--------+
```

- **Fields (combines those of -08 FRAG+LITE)**
  - Offset = pointer to front of the fragment (for near-zero-copy) from -08 LITE
  - Identification = from -08 FRAG
  - Frag. Offset = for reassembly from -08 FRAG
  - Reassy. Checksum = (in terminal fragment only) from -08 FRAG
  - Single fragment = terminal format with zero frag offset
New UNSAFE option

• Includes suboption kind (UKind)
  – Prevents implied UNSAFE use of all Kinds
  – MUST implement (means MUST parse to UKind)

• Format

  +--------+--------+--------+
  | Kind=6 | Length | UKind  |...
  +--------+--------+--------+

• Length varies
  – Allows 2-byte values when Length == 255
  – As with all options with variable length
## “MUST implement” options

<table>
<thead>
<tr>
<th>Kind</th>
<th>Length</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0*</td>
<td>-</td>
<td>End of Options List (EOL)</td>
</tr>
<tr>
<td>1*</td>
<td>-</td>
<td>No operation (NOP)</td>
</tr>
<tr>
<td>2*</td>
<td>3</td>
<td>Option checksum (OCS)</td>
</tr>
<tr>
<td>3*</td>
<td>6</td>
<td>Alternate checksum (ACS)</td>
</tr>
<tr>
<td>4*</td>
<td>10/12</td>
<td>Fragmentation (FRAG)</td>
</tr>
<tr>
<td>5*</td>
<td>4</td>
<td>Maximum segment size (MSS)</td>
</tr>
<tr>
<td>6*</td>
<td>(varies)</td>
<td>Unsafe to ignore (UNSAFE) options</td>
</tr>
<tr>
<td>7</td>
<td>10</td>
<td>Timestamps (TIME)</td>
</tr>
<tr>
<td>8</td>
<td>(varies)</td>
<td>Authentication and Encryption (AE)</td>
</tr>
<tr>
<td>9</td>
<td>6</td>
<td>Request (REQ)</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>Response (RES)</td>
</tr>
<tr>
<td>11-126</td>
<td>(varies)</td>
<td>UNASSIGNED (assignable by IANA)</td>
</tr>
<tr>
<td>127-253</td>
<td></td>
<td>RESERVED</td>
</tr>
<tr>
<td>254</td>
<td>(varies)</td>
<td>RFC 3692-style experiments (EXP)</td>
</tr>
<tr>
<td>255</td>
<td></td>
<td>RESERVED</td>
</tr>
</tbody>
</table>

UKinds 0 and 255 RESERVED, others UNASSIGNED (IANA assignable)

Kind and UKind require IESG Approval or Standards Action (except EXP EXIDs)
RFC 6081 issue

• **Teredo Extensions (standards track)**
  – Fails to update RFC 786 but claims to redefine UDP length in nonsensical ways
  – Fails to address impact on legacy routers

• **Propagates an error about UDP length**
  – RFC 6081 cites RFC 4830 that claims that RFC 2460 (IPv6) requires UDP length to “be consistent” with IP length
  – RFC2460 and RFC8200 have no such requirement
  – RFC 4830 requires “consistent” but this is undefined

• **RFC 6081 introduces a nonsensical extension**
  – RFC 6081 defines 4830 consistent as zero surplus (IP payload = UDP payload + 8)
  – Defines 6081 consistent to allow negative surplus areas (IP payload <= UDP payload + 8)
  – Intended to allow IP trailer (but they got it backwards)
  – As defined, requires IP parsing UDP header to know about the IP trailer
  – As defined, cannot traverse legacy IP (not extended per this RFC)

• **Our UDP options prohibit this variant**
  – Because it would not traverse an IP router unless they parse and validate ALL UDPlen values
  – Thus we should DEPRECATE that RFC

• **Kudos to Fred Templin for heads-up**