UDP Options

IETF 110 – Mar 2021

Joe Touch
strayalpha.com
-09 updates

• Changes (see 11/25/20 post)
  – Added UNSAFE
  – Revised FRAG (integrates prev. FRAG+LITE)

• Clarifications
  – Typos, section numbering
  – OCS pseudoheader and zero issue
  – ACS **not** dropped by default
  – ACS to CRC32c
  – ACS and AE cover payload only
New UNSAFE

- **UNSAFE**
  - Introduces options that modify UDP user data
  - Indicates “user data is unsafe if specific unsafe option is unknown”
  - No options modify other options
  - Halts further option processing if specific option is not supported
    - Including FRAG
    - Indicates user data is NOT OK
    - BUT packet is still passed to app layer by default (to emulate legacy)
    - Apps can override to ask UDP to “default drop”

- **OCS, ACS, AE are different**
  - OCS halts option processing too, but user data remains OK
  - ACS, AE say user data is suspect, but option processing continues even if they fail
Revised FRAG

• Placement matters
  – If first, it supports RDMA-like avoidance of user data copy (as before)
  – “Not first” should should only happen with UNSAFE options, which means data needs to be copied anyway

• When FRAG fails, packet processing continues
  – as with any other failed but supported options (i.e., options after FRAG still happen and the data - if any - would be delivered, or a zero-length packet)
ACS and AE

• Failed ACS / AE is NOT silently dropped by default
  – Default behavior emulates legacy receivers
    • “Failed ACS” flag allows app to decide
  – Receivers can override to silently drop if desired
• ACS and AE cover only the UDP payload
Option processing principles

• Individually ignored if not supported
• Flagged as failing if supported but computes incorrect checksum, etc.
  – The RECEIVER decides what to drop
  – The default is NOT to drop (legacy behavior) but CAN be overridden
• All options to be ignored if any one FAILS due to format / parsing or OCS failure
• BUT NO options can prevent UDP data from going to the app by default
  – Apps that care can override that default
  – Options that should “share fate” with UDP data must be designed as UNSAFE options
  – There are NO currently defined UNSAFE options, FWIW
New issue: MSS

• Originally imported from TCP
  – Hint for path MTU
• UDP has two "maximums"
  – Max fragment size
    • Soft hint for path MTU (as with TCP)
  – Max reassembly size
    • Hard upper bound, similar to MSS_R
• Should we have two MSS options?
  – MaxFrag
  – MaxReassembly
10 pending changes

- Remove “updates ROHC/3095”
  - Add a note that ROHC does not prohibit opts; it runs uncompressed when lengths differ

- Address MSS issue in prev slide
  - Either leave as-is or create separate MSS options