



TCP User Timeout Option (UTO)

draft-ietf-tcpm-tcp-uto-04.txt

**Lars Eggert (NEC) &
Fernando Gont (UTN/FRH)**

67th IETF Meeting, San Diego, California, USA
November 5-10, 2006



Overview

- Peers exchange local, per-connection user timeouts through advisory TCP option and adapt local user timeout accordingly
- Motivation:
 - longer UTO: tolerate longer disconnections
 - shorter UTO: less TCP state at busy servers
- TCP mod, not policy for picking user timeouts
- Adopted as WG item at the 61st IETF Meeting (Washington, DC, USA).

Main changes introduced in -03

- Thorough review by Gorry Fairhurst
- Rearranged the Introduction
 - It is now clear from the beginning what the document is about
- Removed discussion of `SO_RCVTIMEO` and `SO_SNDTIMEO`
 - these parameters refer to reading/writing from/to socket buffers, rather than waiting for data to be acked, etc.
- Clarified that the `UTO` option is disabled by default.
 - Could be enabled on a per-connection basis by a socket option, or on a system-wide basis by a toggle (e.g., `sysctl`)
- Enforce lower limit of one RTO
 - Having a `USER TIMEOUT` of less than one RTO could be problematic

Main changes introduced in -04

- Added advise that an UTO SHOULD be sent in the first segment sent after the SYN segment that initiated the 3WHS. (as suggested by Caitlin Bestler)
 - Particularly useful if the end that performed the passive OPEN does not record all the information included in the initial SYN (e.g., SYN cookies).
- Clarified the impact on interoperability of not negotiating the option during the connection establishment phase (as suggested by Jamshid Mahdavi)
 - Potential of 3% of failures (i.e., it is okay in the vast majority of cases)
 - Those failures result from violating the requirement that TCP MUST ignore unknown options



Changes to be introduced in -05

- Editorial tweaks suggested by Mark Allman
- No major changes



Moving forward

- We think the draft is ready for WGLC.
- Any questions/comments?