Session Timer Glare Handling

IETF#103

Bangkok, Thailand

Christer.Holmberg@ericsson.com

draft-ietf-sipcore-sessiontimer-race

https://github.com/cdh4u/draft-sessiontimer-race

(2) WHAT IS IT ABOUT?

- RFC is unclear on simultaneously session timer negotiations taking place
  - UPDATE request sent while INVITE transaction is still active
  - Both UPDATE and INVITE contain session timer parameters
  - If parameters do not match, which parameters are applied?
- Creating problems in deployed networks

(3) SHOULD I READ THE DRAFT?

- The current version (-02) of the draft is NOT aligned with the recent discussions and suggestions
- A new version of the draft will be submitted once we have agreed on an approach (at least on a high-level) on how to solve the glare issue
- Exactly how the modifications will be documented will be decided once we have a more clear picture of the amount of modifications needed

(4) WHAT HAS HAPPENEDSINCE IETF#102?

- Not much - until the last few weeks
  - E-mail discussions
  - GitHub issue tracker
  - Pre-103 phone call

(5) ANY WORKING ASSUMPTIONS?

- Focus on modifications for fixing the session timer glare issue.
- Non-related bugs can be fixed.
- We are NOT going to do modifications just because someone things (perhaps rightfully so) that it would improve the session timer mechanism in general.
- There is no solution that will make every implementation standards compliant without any modification.
- Approach is to specify procedures that will fix the problem for new/modified implementations, but also how to deal with old/non-modified implementations.

(6) UAC

- If a UA inserts S-E (Session-Expires header field) in an INVITE the UA must insert the same S-E in any UPDATE request that it sends while the INVITE transaction is ongoing

- If a UA has conflicting S-E information once the INVITE and UPDATE transactions have completed, it must send a new UPDATE with S-E, in order to ”sync” the S-E state among all entities

(7) PROXY

- **Request Handling**
  - Must rejects the S-E if the expiration value is too short
    - No matter if the request contains Supported:timer or not
  - If the proxy inserts/forwards/modifies S-E in an INVITE request the proxy must identically insert/fowards/modify the same S-E in any UPDATE request
    - Might insert another S-E if it knows that there is no active INVITE transaction
- **Response Handling**
  - If response contains S-E, the proxy must not modify it
  - If response does not contain S-E, the proxy may insert S-E if it remembers that the associated request indicated support of the session timer (Supported:timer)

(8) UAS

- If received request contains S-E:
  - The UAS copies the S-E of the request into the response
  - The UAS must not reduce the S-E expiration value in the response
    - If the UAS wants to change the S-E value, it later sends a request by its own
- If received request does not contain S-E
  - If the request contains Supported:timer, the UAS might include S-E in the response

(9) OPEN ISSUES

- **Non-offer UPDATE request glare situations**
  - Not covered by RFC 3311 (UPDATE method) or RFC 6141 (target-refresh handling)
  - UA might send non-offer UPDATE session refresh request when peer sends an UPDATE with an offer
- **If UAS does not support session timer, but the UAC does, the proxy can insert S-E only if UAC indicated refresher=uac**
  - Currently the text does not mention the refresher value
- **Meaning of ”initial session refresh request”**

- May not directly/uniquelly related to the session glare issue, but probably worth fixing/clarifying

(10) THE END

Thank You For Listening!