QoS NSLP
draft-ietf-nsis-qos-nslp-12

Jukka Manner,
Georgios Karagiannis,
Andrew McDonald

Recent changes from v.11

- Proxy mode added
- Knowledge of QoS NSLP unaware nodes on the path from QSPEC to NSLP
- Confirmation for tear operation added
- IANA section re-written
- Various editorial things (see the diff on tools.ietf.org)

Changes between 10 and 11

- Some restructuring
- Removed p2p Ack mechanism
- Req-reply on reduced refresh operation added
- INFO_SPEC fixes

TODO

- Make reduced refreshes the default behavior (currently need to acknowledge that)
- Some clarifications needed for proxy mode, object order, tear operation, response/notify
- Summary refresh messages: one NSLP message could refresh several sessions?
- “Stacking”: where do it?

Summary refreshes

- Add a new SUMMARY_REFRESH object carrying SID+RSN pairs
- All identified sessions are refreshed
- A QNE must know the GIST SII_HANDLE (need to have received a message from the peer earlier)
- If a SID is not understood, a RESPONSE is sent back, carrying a SUMMARY_REFRESH

What is “stacking”?

- Effectively it is needed for allowing a domain to
  1. Add QoS objects in addition to those set by the QNI
  2. Have reduced state and reduced operation QNEs inside the domain (edges still do full operation)

Stacking options

1. In NSLP: API to RMF provides some sort of push/pop/replace operation for adding and removing QSPECs in NSLP messages
2. In QSPEC: along the e2e QoS objects, a domain can add local objects used only within the domain

Pros/cons

- In NSLP:
  - Con: (1) adds bytes in message because of two QSPECs, (2) NSLP must understand the concept of stacks, and different operation in QNEs, (3) changes may happen in any QSPEC (need to give all to RMF)
  - Pro: some syntax checking done by QoS NSLP
- In QSPEC:
  - Pro: (1) simpler NSLP, (2) clean layering, (3) less messaging overhead
  - Con: same (larger) QSPEC always provided to RMF