Hackathon News

A brief implementation report on a few RFC6824bis features

Quentin De Coninck, Benjamin Hesmans, Fabien Duchêne,
Christoph Paasch, Olivier Bonaventure

Hackathon Agenda

- MPTCP socket API usage
- MPTCP experimental option (rfc6824bis, section 3.7)
- ADD_ADDR reliability
- MP_TCPRST option (section 3.6)
- Documenting Socket API
- New MP_CAPABLE option

Hackathon Agenda

- MPTCP socket API usage
- **MPTCP Experimental option (rfc6824bis, section 3.7)**
- ADD_ADDR reliability
- **MP_TCPRST option (section 3.6)**
- Documenting Socket API
- **New MP_CAPABLE option**

MPTCP Experimental Option

- Allow MPTCP hosts to exchange opaque options between them
- Partially implemented in Linux kernel (mptcp_trunk, 4.1)
  - Transmission and parsing of option done
- What about the “S”ynchronising bit?
  - Need to keep the sent data
  - How to ensure (in-order) reception of multiple options?

MP_TCPRST Option

- Provide additional feedback about subflow reset
- Implemented in Linux (mptcp_trunk, 4.1)
- Discussion
  - Should T bit taken into account for code != 0x00?
  - Option only sent when removing subflow
    - State is removed after
    - Remote could possibly not see the MP_TCPRST option

MP_TCPRST Option

- Discussion (continued)
  - Why not adding reason code for unknown MPTCP connection?
    - E.g, MP_JOIN SYN with unknown token
  - What about a subflow closed by a remote server due to inactivity timeout?
    - Should it have its own code too?
  - Specific code for response to remove id?
  - Force sending reset after N RTO
    - Code 0x05
  - Still didn’t use codes 0x03 and 0x04 yet

New MP_CAPABLE option

- Connection keys determined by application
- Implemented in Linux
- Seem functional :-)