draft-ietf-mptcp-rfc6824bis-10

Alan Ford

Changes since -09 (July 2017)

- Handling MP_FASTCLOSE with RST (i.e. MP_*REALLY*FASTCLOSE)
- TCP Fast Open Considerations

MP_FASTCLOSE

- Original MP_FASTCLOSE design had reliability; all but one subflows could all be closed with TCP-level RST, but remaining one had ACKed MP_FASTCLOSE to ensure synchronisation
- Still too much state required in some circumstances
- New update specifies ability to do this on a RST packet too:
  - Send RST+MP_FASTCLOSE on all subflows and drop state immediately
  - Receiver does the same across all subflows if receiving RST+MP_FASTCLOSE on any subflow

TCP Fast Open (TFO)

- RFC7413 - goal to save 1xRTT before sending data
- This is done by exchanging cookies in TCP Options in initial SYNs, and then using these cookies in subsequent SYNs along with data in the SYN
- Naturally care must be taken not to exceed TCP options space
- We only care about TFO on initial subflow; later subflow setup has authentication delays so no TFO benefit

TCP Fast Open (TFO) Interactions

- TFO Cookie Request on MP_CAPABLE SYN
  - Enough space for TFO + MP_CAPABLE in SYN
  - Not so much space in SYN/ACK; max TFO cookie length would be 7 bytes if using 19 bytes on “typical” options
  - Recommendation on servers to consider which options are needed and choose whether cookie length is sufficient for needs
- Data Sequence Mapping and TFO
  - It would be possible for a middlebox to mess with initial TFO data, and since no mapping in SYN, no ability to discover this until it is too late
  - Therefore, explicitly exclude TFO data from Data Sequence Space
  - TFO data delivered to application as normal
  - Data Sequence Space starts at IDSN+1 after SYN/ACK exchange

Ready for WGLC?