Notes for literature review

alisonc

2015-04-23

Lots of design decisions for MPTCP were made with the goal of making it compatible with Plain Old TCP and the myriad of middleboxes in the today's internet. This is not something we need to be concerned about. [1]

MPTCP: the same data can be sent on multiple subflows for resilience. The first copy that arrives should be taken as authoritative and delivered to the application; further are ignored. [1, p. 25]

Retransmissions are "clearly suboptimal". All retransmissions on a different subflow first require retransmission on the original subflow. Why? Because compatibility with legacy middleboxes. [1, § 3.3.6] Also, much about subflows depends on local policy. [1, § 3.3.8]

"Multiple retransmissions are triggers that will indicate that a subflow performs badly and could lead to a host resetting the subflow with a RST. However, additional research is required to understand the heuristics of how and when to reset underperforming subflows. For example, a highly asymmetric path may be misdiagnosed as underperforming." [1, p. 33]

Subflow priority is binary – either main or backup. This is just a suggestion and not binding though.

Address adding and removing via `ADD_ADDR` and `REMOVE_ADDR` depends on the hosts view of its network connections, not the topology. [1, § 3.4].

MPTCP has some shortcomings:

- No network level view of topology. can't pick the k shortest paths
- Most applicable to multihomed systems
- data redundancy is implement badly
- All the design compromises to be compatible with legacy middleboxen

References

[1] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure. TCP extensions for multipath operation with multiple addresses, January 2013. RFC 6824.