LISA: A Linked Slow-Start Algorithm for MPTCP
draft-barik-mptcp-lisa-00

Runa Barik (UiO), Simone Ferlin (SRL), Michael Welzl (UiO)

Multipath TCP @94th IETF Meeting
Yokohama, Japan
2nd November 2015
**Problem Statement**

**Performance Issue to MPTCP:**

- MPTCP is aggressive during slow start (SS): subflows are uncoupled.
- What will a Datacenter face when 8 subflows per MPTCP is recommended [3]?

![Diagram showing the sum of cwnd (in pkts) over time (in terms of RTTs) for TCP, MPTCP with 2 subflows, and MPTCP with 8 subflows.](image-url)
LISA tries to behave like one TCP during SS.

Basic approach in LISA:
- From subflows in SS, select subflow with maximum sending rate \( \text{max}_\text{subflow} \).
- From \( \text{max}_\text{subflow}'s \text{cwnd} \), take between 3 and IW packets as "packet credit" to give \( \text{new}_\text{subflow} \) as IW.
- \( \text{Max}_\text{subflow} \) ignores cwnd-increase for \( (\text{packets}_\text{inflight} - \text{cwnd}) \) ACKs.
- If no \( \text{max}_\text{subflow} \), set the IW of \( \text{new}_\text{subflow} \) as per RFC 6928.
Alternative approach:
- Assign smaller IW to each subflow.
  - Problem: we do not know how many subflows will be established.

Implementation:
- LISA is implemented as a patch to the Linux kernel 3.14.33+, within MPTCP’s v0.89.5.
- The algorithm is explained in more detail in the draft.
LISA results in CORE emulator

(a) Shared bottleneck: 2.5Mbps, 70ms, transfer-size: 300KB

Performance Evaluation metrics:
- Completion time $T$,
- Total retransmissions $R$, and retransmissions during SS, $R_{ss}$.

(b) Total cwnd
Shared Bottleneck

(a) Shared bottleneck: 5Mbps, 40ms

(b) Total retransmits (SS) $R_{ss}$ vs buffer size

(c) Mean completion time vs file size

(d) Retransmissions vs file size
Datacenter

(a) Datacenter: 50Mbps, 5ms

(b) $R_{ss}$ vs buffer size

(c) Mean compl. time vs file size

(d) $R$ vs file size
LISA performance in Nornet testbed

Multi-homed Client

Subflow@3G
India
U.S.A.

Server
Subflow@WLAN
Lab network

3G
WLAN

U.K.
Germany

U.S.A.
India

Non-shared bottleneck

Download

Background Traffic

Shared bottleneck

Runa Barik
LISA for MPTCP
Results

(a) Shared Bottleneck

(b) Non-Shared Bottleneck

MPTCP and MPTCP-LISA: Completion Time
Conclusion

- We identified the adverse effect of MPTCP’s uncoupled slow-start on the performance of MPTCP itself and on concurrent TCP traffic.
- LISA was implemented as a patch to the Linux kernel 3.14.33+, within MPTCP’s v0.89.5.
- We analysed the performance of LISA in shared, non-shared bottleneck experiments in both emulation and real testbed, and in a datacenter topology in emulation.

Acknowledgements:
- This work was part-funded by the European Community under its Seventh Framework Programme through the Reducing Internet Transport Latency (RITE) project (ICT-317700).


Thank you!