# SCTP Send buffer Advertising

## CS4099 Project
## Final Evaluation

Arpan Kapoor, Deepak Sirone J, K Prasad Krishnan
Guided By:
Dr. Vinod Pathari
Mr. V Anil Kumar, Principal Scientist, CSIR, Bengaluru

April 25, 2016

# Outline

# Introduction

- Stream Control Transmission Protocol (SCTP):
  - Supports multiple logical channels called streams
  - Multi-homing
- Send buffer Advertising:
  - specialized chunks will carry the amount of backlogged data present in the sender's buffer.

# Problem Statement

- To propose a scheme to
  - advertise send buffer occupancy information in SCTP
  - implement it in the Linux kernel and
  - study the performance implications of the same.

# Previous Design

- New chunk type with Chunk Type value between 128 to 190.
- Highest order 2 bits determine action to be taken if Chunk Type is unknown.
- This ensures that unmodified hosts won't send a Unrecognized Chunk Type Error chunk upon reception.

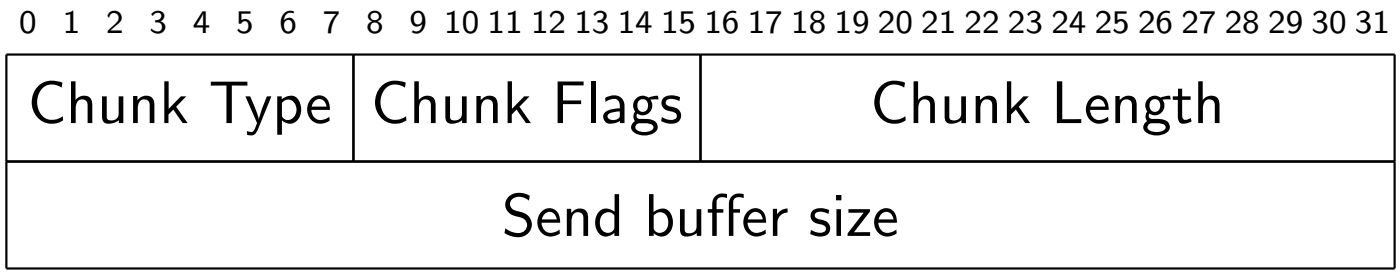



Figure: Proposed Chunk for send buffer advertisement

# Current Design

- Every SCTP packet having a DATA chunk contains the percentage send buffer occupancy chunk as the first chunk.
- Traffic controller classification required each packet to have the send buffer occupancy information.

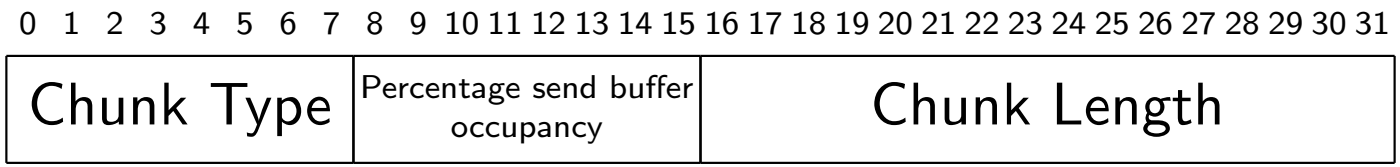



Figure: Proposed Chunk for send buffer advertisement

# Test bed Design

A dumbbell shaped network topology was created with two routers in the center, and multiple hosts connected to one end of each router via a switch. This ensures that we have a bottleneck link in all flows between end hosts on either side.

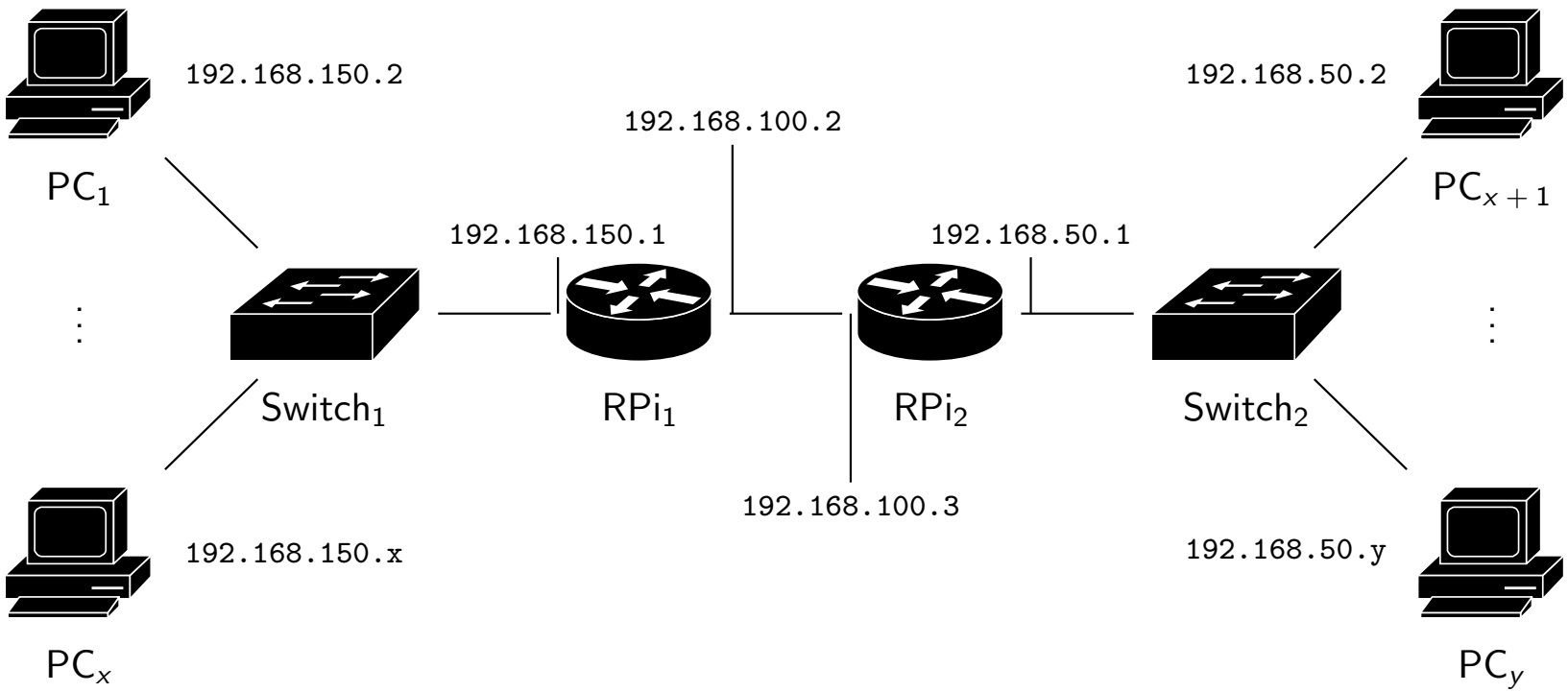



Figure: Test bed implementation

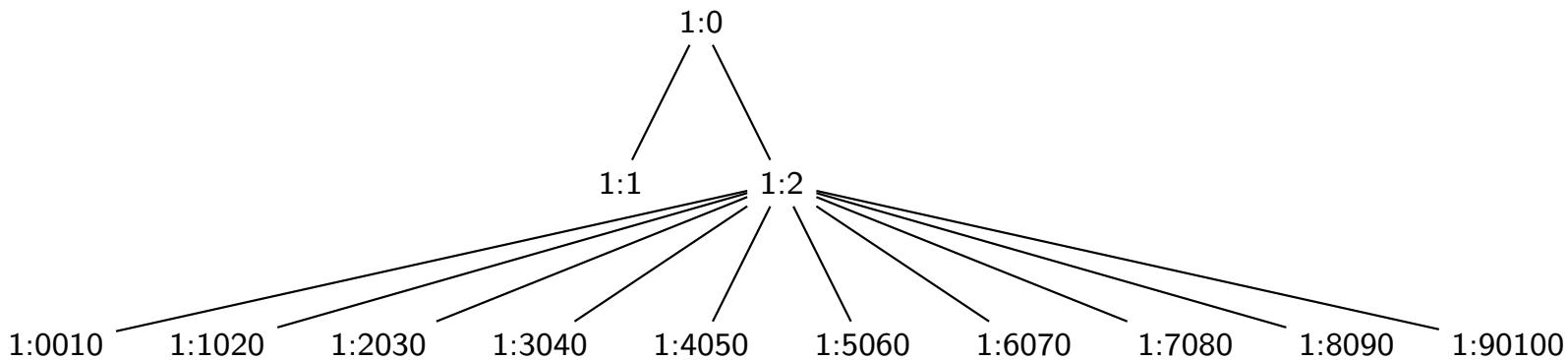
# Work Done

- Modified kernel module `sctp_probe` to measure send buffer.
- Explored Linux kernel SCTP implementation.
- Identified parameter to be advertised.
- A patch implementing the SCTP send buffer advertisement was created for Linux kernel `v4.6-rc4`.
- The send buffer advertisement chunk type value was set to 150.
- Previous design: A state table was created for this chunk, specifying the states in which the send buffer advertisement chunk should be generated and sent.

# Tests conducted

- Queueing discipline - Classless & Classful.
- Classless - TBF, SFQ, etc.
- Classful - HTB, CBQ, etc.

AIM - create a classification hierarchy which works better than the existing qdiscs.



# Results I

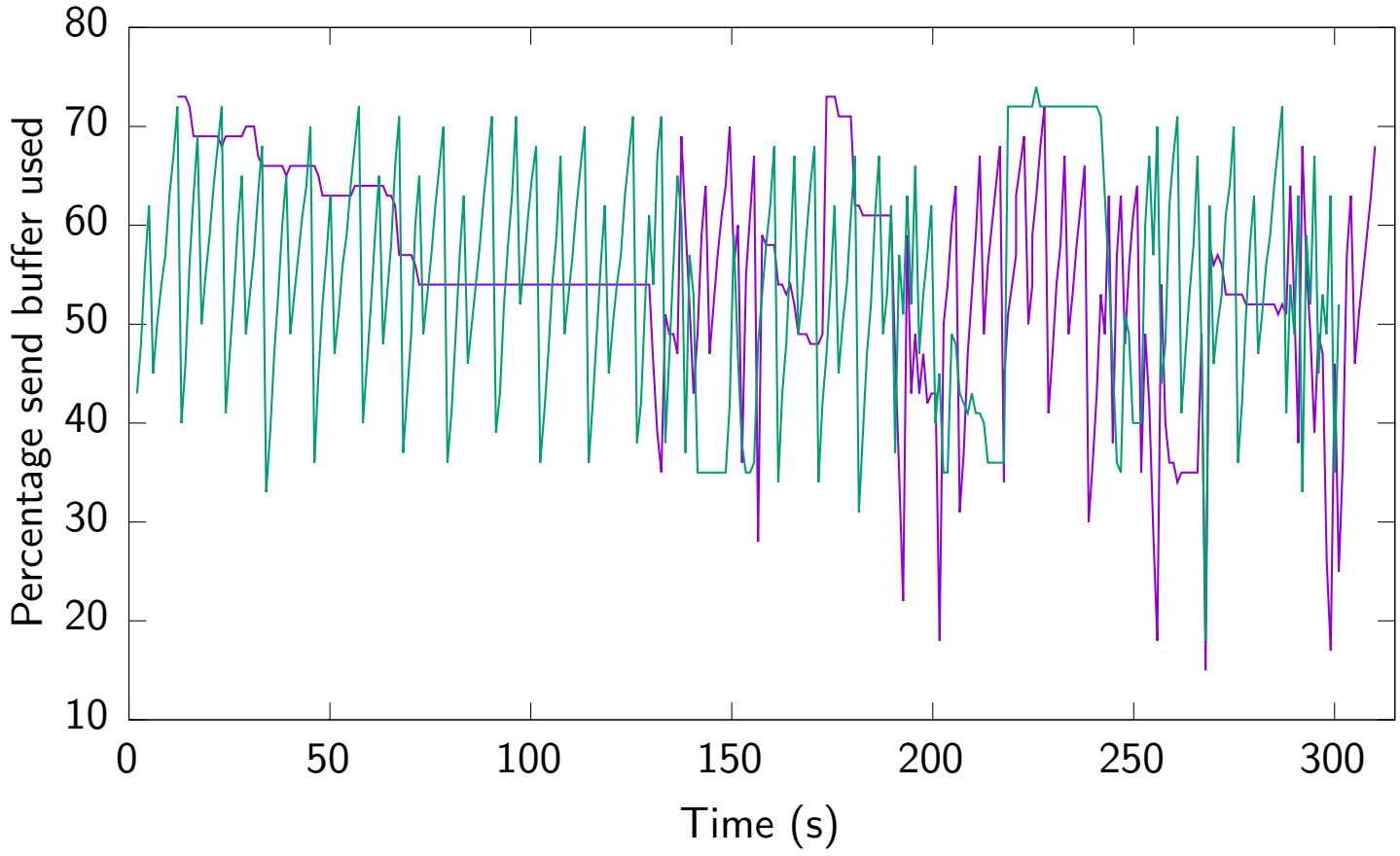



Figure: Percentage send buffer variation with 2 flows each having duration of 300 seconds using the Token Bucket Filter qdisc with rate limited to 1mbit/sec

# Results II

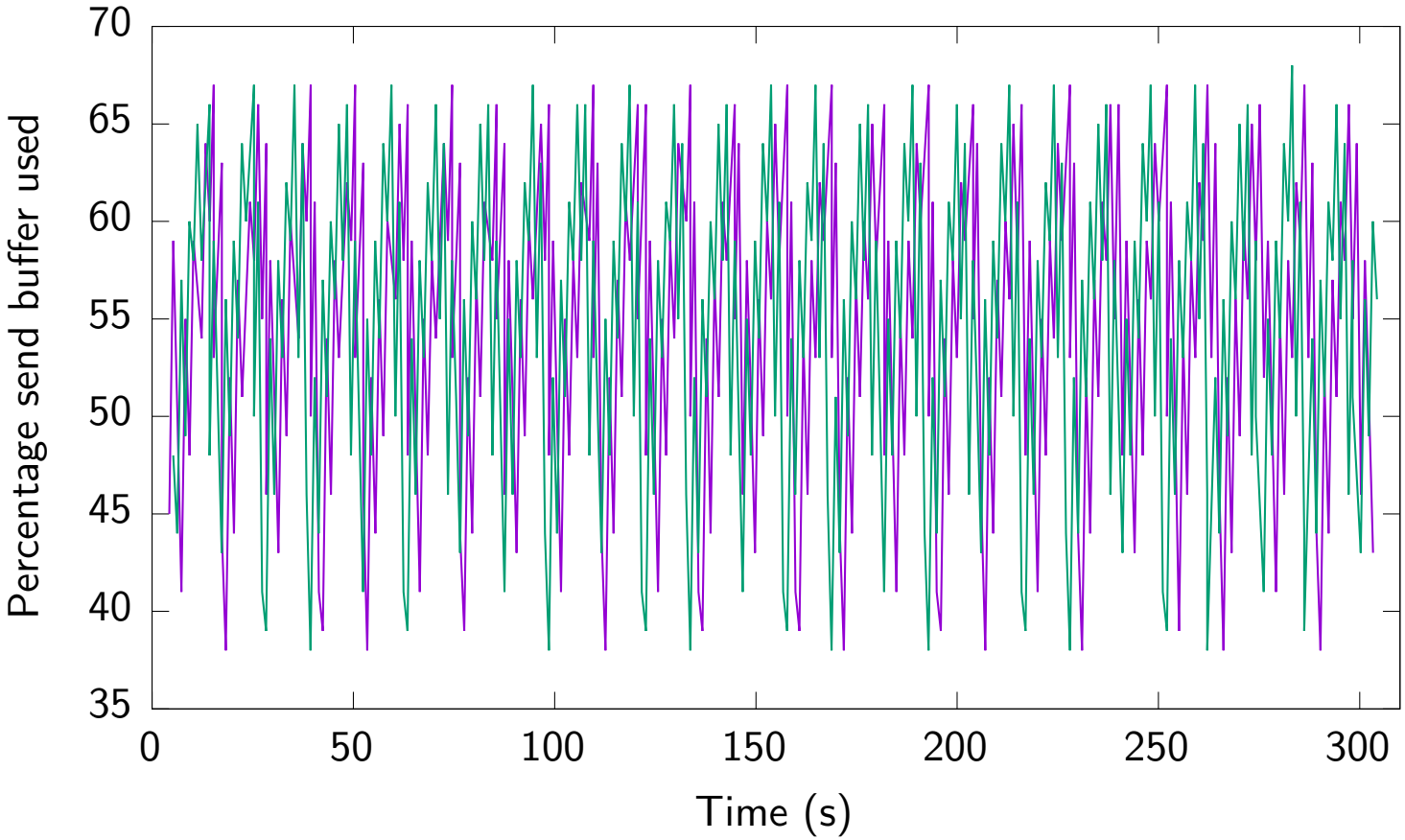



Figure: Percentage send buffer variation with 2 flows each having duration of 300 seconds using Stochastic Fair Queuing qdisc with rate limited to 1mbit/sec

# Work yet to be done

- Debugging the Linux kernel implementation of the current design.
- Testing the implementation with the proposed classification hierarchy.

# References I


[1] A. Agache and C. Raiciu. *TCP Sendbuffer Advertising*. Internet-Draft draft-agache-tcpm-sndbufadv-00.txt. IETF Secretariat, July 2015.

[2] Alexandru Agache and Costin Raiciu. "Oh Flow, Are Thou Happy? TCP Sendbuffer Advertising for Make Benefit of Clouds and Tenants". In: *7th USENIX Workshop on Hot Topics in Cloud Computing (HotCloud 15)*. Santa Clara, CA: USENIX Association, July 2015. URL: https://www.usenix.org/conference/hotcloud15/workshop-program/presentation/agache.

[3] Ian F. Akyildiz et al. "A roadmap for traffic engineering in SDN-OpenFlow networks". In: *Computer Networks* 71 (2014), pp. 1–30. ISSN: 1389-1286. DOI: http://dx.doi.org/10.1016/j.comnet.2014.06.002. URL: http://www.sciencedirect.com/science/article/pii/S1389128614002254.

# References II


[4] Karthik Budigere. "Linux Implementation Study of Stream Control Transmission Protocol". In: *Proceedings of Seminar on Network Protocols in Operating Systems*, p. 22.

[5] M. Tim Jones. *Better networking with SCTP*. Feb. 28, 2006. URL: http://www.ibm.com/developerworks/library/l-sctp/.

[6] L. Ong and J. Yoakum. *An Introduction to the Stream Control Transmission Protocol (SCTP)*. RFC 3286. RFC Editor, May 2002, pp. 1–10. URL: http://www.rfc-editor.org/rfc/rfc3286.txt.

[7] Jon Postel. *Transmission Control Protocol*. RFC 793. RFC Editor, Sept. 1981, pp. 1–85. URL: http://www.rfc-editor.org/rfc/rfc793.txt.

[8] R. Stewart. *Stream Control Transmission Protocol*. RFC 4960. RFC Editor, Sept. 2007, pp. 1–152. URL: http://www.rfc-editor.org/rfc/rfc4960.txt.

# References III

[9] R. Stewart et al. *Sockets API Extensions for the Stream Control Transmission Protocol (SCTP)*. RFC 6458. RFC Editor, Dec. 2011, pp. 1–115. URL: http://www.rfc-editor.org/rfc/rfc4960.txt.