



UNIVERSITY OF STAVANGER

BACHELOR THESIS

DATBAC

Make the Internet Faster!
Improving Alternative Backoff with ECN in Linux

Students

Dan Erik RAMSNES

Erlend Moen AL-KASIM

Supervisor

Naeem KHADEMI

January 31, 2020

Abstract

Lorem ipsum sit amet, consectetur adipiscing elit. Sed mollis dolor risus, a pulvinar quam pretium et. Suspendisse ut dolor arcu. Nulla facilisi. Nullam finibus vestibulum nulla, ac sollicitudin neque pretium vitae. Donec in est pretium, elementum velit et, pulvinar enim. Vestibulum consectetur et lectus ut fringilla. Donec malesuada, ante quis ultricies feugiat, leo tortor egestas nunc, non sagittis nisl lorem ac felis. Donec venenatis eget tortor et aliquam. Integer a sapien ultricies, dapibus erat sit amet, luctus ex.

Donec vitae metus pretium, tempus sapien eget, maximus quam. Nam dictum aliquam mi, ut fringilla nunc vehicula a. Morbi vitae dictum eros. Phasellus non urna felis. Etiam eu lectus justo. Etiam et ex ultrices, elementum erat laoreet, hendrerit lorem. Vivamus imperdiet consectetur dictum. Sed ac orci placerat quam rutrum pellentesque. Proin sollicitudin diam et erat feugiat feugiat. Quisque tempus velit viverra sem aliquet feugiat vitae et augue. Donec at odio viverra, posuere tortor eu, volutpat quam. Proin id rutrum metus, id mollis ipsum.

Sed pulvinar tristique nibh eu convallis. Integer luctus pretium massa sit amet placerat. Donec tincidunt consectetur efficitur. Nam tincidunt libero ut nisi lacinia, rhoncus cursus elit lobortis. Interdum et malesuada fames ac ante ipsum primis in faucibus. Duis vel semper lectus. Donec et ullamcorper turpis.

Contents

Abstract	i		
1 Introduction	1		
1.1 Background and Motivation	1		
1.2 Goals and Research Questions . . .	1		
1.3 Research Methodology	1		
1.4 Thesis Structure	1		
2 Theory	2		
2.1 Network Delay and Bufferbloat . .	2		
2.2 Transmission Control Protocol . . .	2		
2.2.1 Congestion Control	2		
2.3 Active Queue Management	2		
2.3.1 Explicit Congestion Notifi- cation	2		
2.4 Alternative Backoff with ECN . . .	2		
3 Methodology	3		
3.1 Network Topology	3		
3.1.1 Raspberry Pi 4 Cluster . . .	3		
		3.2 TCP Experimentation with TEACUP	3
		3.2.1 Exposing TCP State with web10g	3
		3.3 Achieving Low Latency with ABE	3
		3.4 Improving ABE by Adapting Its Reduction Factor β	3
		4 Results	4
		5 Conclusion	5
		A The PI4-Cluster Testbed	6
		A.1 Setting Up Dual Boot	6
		A.2 Compiling Mainline Kernel 5.5 for Raspberry Pi 4	7
		A.3 Patching web10g on Mainline Ker- nel 5.5	7
		Terms	8
		References	9

Introduction

This chapter aims at giving an introduction and overview of the thesis. It starts with a brief explanation of why Internet today still feels slow despite major advances in technology, followed up by establishing the goals and research questions for the entire thesis. To address the research questions, a small view into the research methodology is presented, with a brief overview of the thesis structure in the final section.

Donec vitae metus pretium [GMS93], tempus sapien eget, maximus [Knu]. Nam dictum aliquam mi, ut fringilla nunc vehicula a. Morbi vitae dictum eros.

1.1 Background and Motivation

Maecenas pulvinar quis quam eu convallis. Fusce vulputate sodales suscipit. Nullam porttitor hendrerit lacinia. Cras tincidunt ultrices lobortis. Sed ac sem efficitur, faucibus nunc vehicula, elementum arcu. Aenean pellentesque augue ut massa ullamcorper congue. Phasellus varius at erat at lobortis. Ut nunc metus, consequat nec blandit in, laoreet nec ligula. Integer interdum, massa eu accumsan eleifend, nunc nunc ultrices augue, at commodo risus ligula varius ex. Praesent blandit risus sit amet tellus semper, a suscipit libero malesuada. Mauris nec lacus ipsum. Nullam at ipsum rhoncus, mattis nibh vitae, placerat odio. Sed nec leo id dui vulputate accumsan. Pellentesque placerat congue arcu id pharetra. Maecenas imperdiet ex sed vehicula euismod.

1.2 Goals and Research Questions

Aliquam quis imperdiet orci, nec vestibulum risus. Curabitur vitae euismod mauris, ut iaculis erat. Ut interdum ex ac elit ultrices, at mattis magna ultricies. Duis ac suscipit nibh. Mauris sagittis consequat elit eget euismod. Mauris tincidunt dui ex, sit amet vulputate sem pretium non. Quisque at nulla lobortis, facilisis ligula eu, efficitur ante. Praesent bibendum dolor purus, non placerat felis pellentesque eu.

1.3 Research Methodology

1.4 Thesis Structure

Theory

2.1 Network Delay and Bufferbloat

2.2 Transmission Control Protocol

2.2.1 Congestion Control

2.3 Active Queue Management

2.3.1 Explicit Congestion Notification

2.4 Alternative Backoff with ECN

Methodology

3.1 Network Topology

3.1.1 Raspberry Pi 4 Cluster

3.2 TCP Experimentation with TEACUP

3.2.1 Exposing TCP State with web10g

3.3 Achieving Low Latency with ABE

3.4 Improving ABE by Adapting Its Reduction Factor β

Chapter 4

Results

Conclusion

The PI4-Cluster Testbed

A.1 Setting Up Dual Boot

First install Ubuntu. When asked for partitioning the disk, choose manual, select the disk and confirm creating a new empty partition with yes. Select the newly created empty partition followed by create a new partition and set a size for it. The type should be of primary, location at beginning and mounting point root. Finish off with done setting up the partition followed by finish partitioning and write changes to disk.

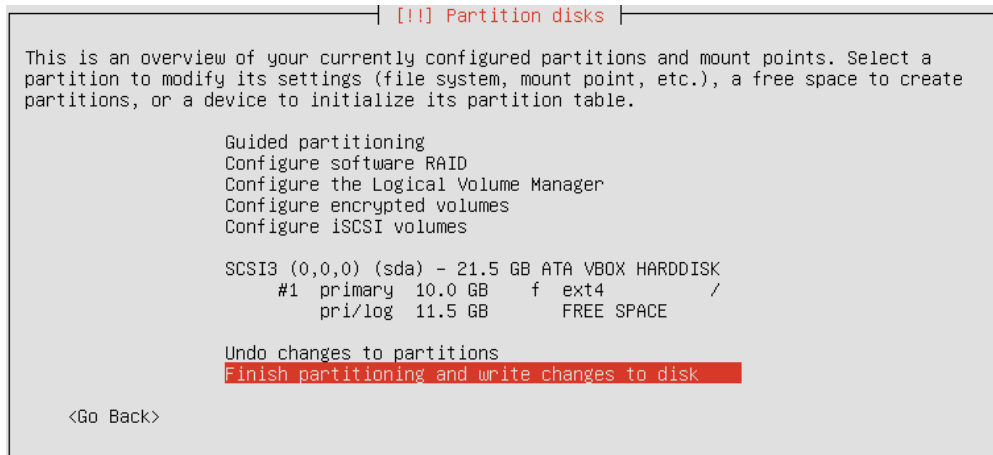


Figure A.1: The partition editor for Ubuntu.

Next, install FreeBSD. When asked for partitioning the disk, choose **auto** (UFS) followed by partition. Set a size, hit ok and finish.



Figure A.2: The partition editor for FreeBSD.

After installing both systems, only Ubuntu is presented in the **GRand Unified Bootloader (GRUB)**. To add FreeBSD as an option, run `sudo nano /etc/grub.d/40_custom` in Ubuntu, and add the following entry:

```

1 menuentry "FreeBSD" {
2     insmod ufs2
3     set root=(hd0,2)
4     kfreebsd /boot/loader
5 }

```

Then update **GRUB** with `sudo update-grub`. The FreeBSD option should now be available when rebooting. If the bootloader won't display, hold the RIGHT SHIFT key upon booting.

To enable a one-time reboot into FreeBSD from Ubuntu, run the command `grub-editenv /boot/grub/grubenv` set `next_entry="FreeBSD"` and reboot with `sudo reboot`.

A.2 Compiling Mainline Kernel 5.5 for Raspberry Pi 4

A.3 Patching web10g on Mainline Kernel 5.5

Terms

GRand Unified Bootloader (GRUB) A Multiboot boot loader. It was derived from GRUB, the GRand Unified Bootloader, which was originally designed and implemented by Erich Stefan Boleyn.

References

- [GMS93] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The L^AT_EX Companion*. Reading, Massachusetts: Addison-Wesley, 1993.
- [Knu] Donald Knuth. *Knuth: Computers and Typesetting*. URL: <https://www-cs-faculty.stanford.edu/~knuth/abcde.html>.