

Unikernel Linux (UKL)

Isaiah Stapleton
Software Engineer Intern
Research Team

What we'll discuss today

- ▶ Brief Introduction
- ▶ Background
- ▶ Project Description
- ▶ Main Task
- ▶ Why this Work Matters
- ▶ Current Progress
- ▶ Side Task
- ▶ Problems & Solutions
- ▶ Next Steps
- ▶ Related Work
- ▶ Things I Learned
- ▶ Conclusion
- ▶ Q & A

Brief Introduction



- ▶ Born in Dover, NH
- ▶ Moved to Charleston, SC ~12 years ago
- ▶ Going into senior year at the College of Charleston
 - Computer Science Major & Data Science Minor
- ▶ Passionate about Linguistics (Language & Language Acquisition)
 - Currently working on one year research project designing a mobile language learning app utilizing comprehensible input

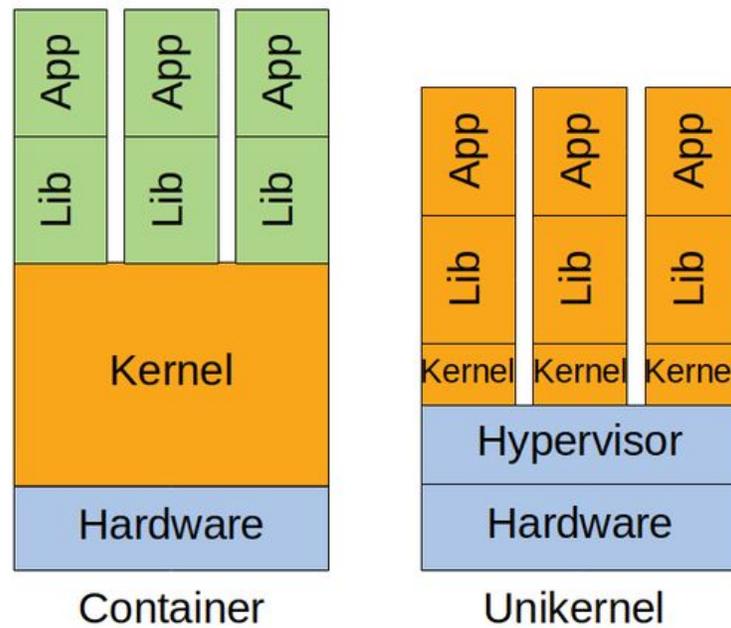
Background

What is a Unikernel?

A highly specialized OS that is designed to run a single privileged application that is linked with a kernel and executes in a single address space

- ▶ Only contains application code as well as all necessary operating system functions required by that application

Unikernel Architecture



What is the point of a Unikernel? What problem do they aim to solve?

- ▶ Virtual machines are often used to run only a single application belonging to a single user
- ▶ No need to support:
 - Isolation between different users
 - Thousands of drivers
 - Isolation between different processes
- ▶ Benefits:
 - Fast boot times
 - Increased performance
 - Increased security

Project Description

Unikernel Linux (UKL)

- ▶ Small patch to Linux and glibc which allows one to build many programs as Unikernels
- ▶ A system for applying Unikernel optimization techniques to the Linux Kernel
- ▶ Current Goals: Offer a selection of applications optimized using Unikernel techniques and working towards upstream submission of the UKL patch set

Why did I choose this
project?

Main Task

Main Task

Port Nginx over to UKL and perform benchmark testing in order to compare results to the Lupine benchmark tests



Why this Work Matters

Why?

In order to get our changes to the Linux kernel pushed upstream we have to show that our changes to these applications can make a significant difference in their performance

Current Progress

Current Progress

Build Nginx from source

Figure out what dependencies it has (and their dependencies, all the way back to glibc)

Make sure we can build entire dependency tree and Nginx in a UKL compatible way

Current Progress

Build Nginx in a UKL compatible way

Boot Nginx as a UKL kernel

Perform benchmark tests on UKL-Nginx

Benchmark

UKL-Nginx

Lupine-Nginx

```
Document Path:      /index.html
Document Length:    612 bytes

Concurrency Level:   100
Time taken for tests: 6.568 seconds
Complete requests:  100000
Failed requests:     0
Total transferred:  84500000 bytes
HTML transferred:   61200000 bytes
Requests per second: 15225.91 [#/sec] (mean)
Time per request:   6.568 [ms] (mean)
Time per request:   0.066 [ms] (mean, across all concurrent requests)
Transfer rate:      12564.35 [Kbytes/sec] received
```

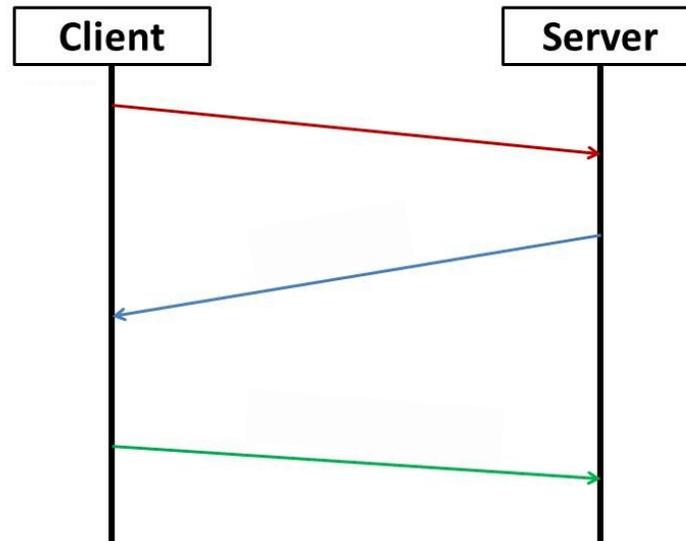
```
Document Path:      /index.html
Document Length:    612 bytes

Concurrency Level:   100
Time taken for tests: 9.119 seconds
Complete requests:  100000
Failed requests:     0
Total transferred:  84500000 bytes
HTML transferred:   61200000 bytes
Requests per second: 10965.75 [#/sec] (mean)
Time per request:   9.119 [ms] (mean)
Time per request:   0.091 [ms] (mean, across all concurrent requests)
Transfer rate:      9048.89 [Kbytes/sec] received
```

Side Task

Side Task

Implement TCP shortcuts in order to improve the performance of Nginx



Current Progress

Implement send shortcut

Implement receive shortcut

Problems & Solutions

Task: Main Task

Problem	Root Cause	Solution
Open /dev/null/ failed, no such file or directory	/dev not populating when building UKL-Nginx	Richard Jones added changes to UKL to change the way applications are built

Task: Main Task

Problem	Root Cause	Solution
Socketpair() failed while spawning worker process	A function in the Linux kernel (putuser) that writes values from kernel space into user space. If destination was found to be outside of user space the process fails / terminates	Modified Linux kernel to bypass this check

Task: Side Task

Problem	Root Cause	Solution
Undefined reference to shortcut_tcp_send msg	Nginx failing to build, causing the entire build process to halt	Editing the Nginx Makefile to ignore failure to build Nginx in order to continue the build process

Task: Side Task

Problem	Root Cause	Solution
Kernel panic during benchmark testing	???	???

Next Steps

Next Steps

Get Lupine benchmark tests running on same hardware ✓

Compare UKL-Nginx benchmark tests to the Lupine-Nginx benchmark tests ✓

Further optimize UKL-Nginx ✓

Tools I Need to Learn

- ▶ Cscope
- ▶ GDB
- ▶ Vim
- ▶ C

Related Work

Related Work

- ▶ [Linux Kernel Development Workshop](#)
 - Jonathan Cameron
- ▶ [Linux Kernel Development 3rd Edition by Robert Love](#)

Things I learned

Things I Learned

- ▶ Kernel Development
- ▶ Open Source Development
- ▶ Git
- ▶ Presentation skills
- ▶ How and when to ask for help
- ▶ How to fight imposter syndrome

Conclusion

Thank you!

Technical Mentors: Eric Munson & Larry Woodman

Early Talent Mentor: Shamair Faison

Manager: Heidi Dempsey

Ali Raza, Richard Jones, and the rest of the UKL &
research team

Questions?

Connect with me on LinkedIn

