Advanced Operating Systems: Lab 1 – Getting started with kernel tracing

Lecturelet 1

Prof. Robert N. M. Watson

2022-2023

Lab 1 objectives

In Labs 2 and 3, you will:

- Utilise systems research methodology and practice
- Explore real-world systems artefacts through performance and functional evaluation/analysis
- Develop scientific writing skills **(L41 only)**

The purpose of Lab 1 is to teach skills needed in later labs:

- Using DTrace for some basic OS instrumentation
- Managing the data coming out of DTrace using Python
- Plotting using matplotlib and flame graphs

This lab contributes less to your overall mark, and has a different format than later labs

Documents on the module/unit websites

- We have provided several documents you will need from the module website, which apply to all labs:
  - Lab Setup Guide (Everyone)
  - DTrace Quick Start (Everyone)

- These documents are specific to Lab 1:
  - Lab 1 – Kernel Tracing – Assignment (Everyone)
  - Lab 1 – JupyterLab notebook (Everyone)

- NB: In Labs 2-3, materials for Part II and L41 will differ.

Lab 1 – Getting started with kernel tracing

- This lab represents only a small number of assessed marks – it is intended as a learning exercise before we hit more complex topics:
  - Introduce our RPi4/FreeBSD/JupyterLab environment
  - Drive a simple benchmark using a JupyterLab notebook
  - Instrument the kernel using DTrace
  - Trace and profile various kernel activities
  - Do a little data analysis and plotting
  - Measure and report on the probe effect
- The format is “short answer” and all students (Part II, L41) will submit PDF output using Moodle.
  - Generate this PDF by using the “Print to file…” mechanism in your web browser
  - Try this out early – if you have trouble, get in touch.

The workload: dd(1)

- Not so much of a benchmark as a standard UNIX I/O tool:

DD(1) FreeBSD General Commands Manual DD(1)

NAME
dd – convert and copy a file

SYNOPSIS
dd [operands ...]

DESCRIPTION
The dd utility copies the standard input to the standard output. Input data is read and written in 512-byte blocks. If input reads are short, input from multiple reads are aggregated to form the output block. When finished, dd displays the number of complete and partial input and output blocks and truncated input records to the standard error output.
…

The workload: Command-line arguments

- We will use dd(1) to copy blocks from `/dev/zero` (source of all zeroes) to `/dev/null` (endless sink for all data):

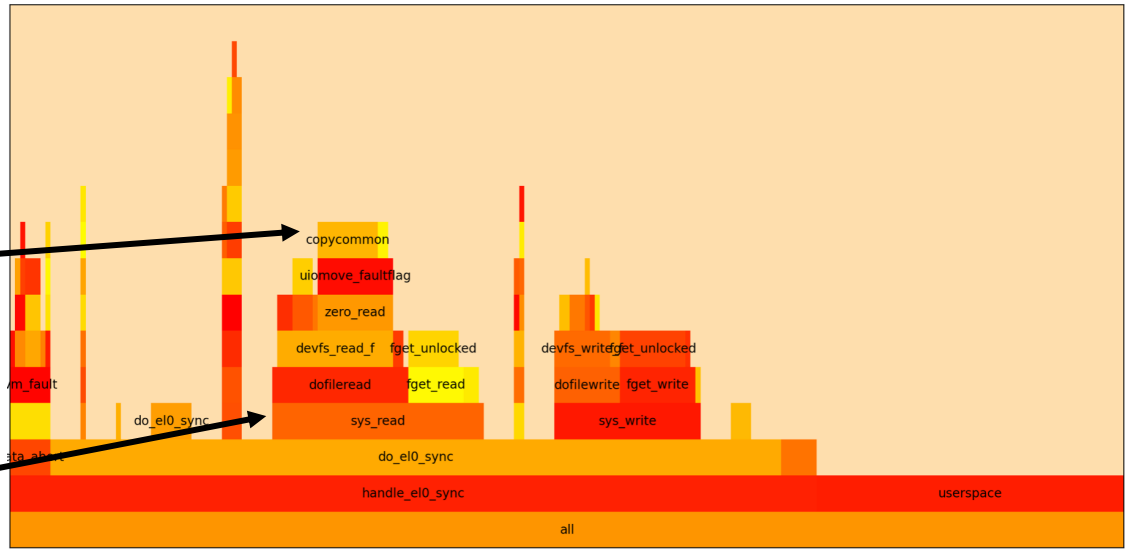
```
dd if=/dev/zero of=/dev/null bs=1K count=10000
```

- The block size is 1,024 bytes, and we copy 10,000 blocks – i.e., a bit less than 10MiB
- This is a kernel-centric benchmark that does nothing productive, but does quite a lot of work to do that
- We will explore where the CPU cycles go!

Flame graphs

- You will already be familiar with most of the plot types we use in this course – scatter plots, histograms, etc.
- But we will also use **flame graphs**, used to render collections of stack traces captured via profiling [etc]
  - X axis represents (unordered) execution time (**or another unit**)
  - Y axis captures call chain – "top" entries are on CPU

In this example, the copycommon() routine is a significant user of CPU time when called (indirectly) from sys_read()



Y axis: Call graph

X axis: Proportion of overall execution time

- See Gregg, CACM 59(6) / ACMQ April 2016

How to get started

- Acquire login credentials to your RPi4 (see email)
- SSH into it setting up TCP port 8080 forwarding as described in the lab setup guide and Lecture 2
- Following the Lab 1 assignment material:
  - (Optionally under tmux) launch JupyterLab
  - Open the first lab
  - Step through cell by cell, trying out and experimenting with the example we've given
  - When you get to the exercises, complete them
  - Print to PDF and submit via Moodle by the deadline
- Come to the first lab session and we can offer help and advice as you work through the lab
- The course instructor cannot extend assignment deadlines; this must be done via the undergraduate or graduate education offices / your college tutor

How to contact us

- Preferred: Course slack
  - advopsys.slack.com
- Also possible: Email to the lecturer
  - robert.watson@cl.cam.ac.uk