DSbD Programme Cohort 2 – Pytilia Project Plan

Applicability of DSbD to performance-sensitive networking applications

Table of Contents

[Introduction 1](#_Toc115350599)

[October 2022 1](#_Toc115350600)

[November 2022 1](#_Toc115350601)

[December 2022 2](#_Toc115350602)

[January 2023 2](#_Toc115350603)

[February 2023 2](#_Toc115350604)

[March 2023 2](#_Toc115350605)

## Introduction

This document describes the draft plan for Pytilia’s DSbD cohort 2 project: “*Applicability of DSbD to performance-sensitive networking applications*”

The outline given below assumes a 6-month project commencing at the start of October 2022. The milestones listed reflect what appears (at the outset of the project) to be the most natural order of progression towards the project’s goals. However we aim to preserve maximum flexibility to re-order tasks if this makes sense as the project progresses.

As with any similar research project, effort estimates are approximate and subject to revision on the basis of findings obtained in the course of our investigations.

## October 2022

The goal for October was familiarisation with Morello, CheriBSD and CHERI concepts

Relevant material will include:

* Technical reports from the University of Cambridge with particular emphasis on the C/C++ Programming Guide
* Security Analysis of CHERI (Microsoft paper)
* ARMs description of the Morello architecture
* DSbD “getting started” guide

## November 2022

Successfully boot, install and configure CheriBSD on the supplied Morello board

Using the theory gleaned in the course of the previous month, initialise and bring up the Morello board and successfully load CheriBSD on that board.

Verify the ability to build and execute ‘Pure Cap’ and standard (Compat) sample applications on the Morello board. Note that the build itself need not be performed on the Morello board, cross-compilation is also acceptable.

## December 2022

Develop a single-process CHERI-enabled simple packet processing application with DPDK. The application should support bounds checking, permissions enforcement and memory compartmentalisation for incoming packet buffers – packet consumers and the DPDK library itself should be co-located in the same address space.

## January 2023

Develop a simple non-CHERI packet processing application using DPDK. Each packet consumer should run as a separate process in its own address space and the DPDK process should transfer packets to the consumer processes using a standard inter-process communications (IPC) mechanism.

## February 2023

Define test packet streams (including a range of packet sizes and frequencies) and transmit those streams to both varieties of the packet processing application (CHERI-enabled and IPC-based)

Measure and analyse key performance characteristics including packet processing latency and CPU utilisation for both varieties

## March 2023

Identify key findings from the development and performance measurement activities and write these up in a final project report with conclusions and supporting evidence.

Clearly state whether the CHERI-enabled application architecture yields any performance benefits when compared with the traditional IPC-based architecture. If any performance benefits are present, list the test scenarios in which they become apparent.

Enumerate any areas in which further research may be worthwhile.

## Background Reading

* [University of Cambridge CHERI](https://www.cl.cam.ac.uk/research/security/ctsrd/cheri/)
* <https://www.dsbd.tech/>
* <https://www.arm.com/architecture/cpu/morello>
* [Security Analysis of CHERI (Microsoft)](https://github.com/microsoft/MSRC-Security-Research/raw/master/papers/2020/Security%20analysis%20of%20CHERI%20ISA.pdf)