CTSRD

CRASH-WORTHY TRUSTWORTHY SYSTEMS RESEARCH AND DEVELOPMENT

(The First?) CHERI Microkernel Workshop

Robert N. M. Watson and **Simon W. Moore**
University of Cambridge

Peter G. Neumann
SRI International

CHERI Microkernel Workshop – 23 April 2016

Approved for public release; distribution is unlimited. This research is sponsored by the Defense Advanced Research Projects Agency (DARPA) and the Air Force Research Laboratory (AFRL), under contracts FA8750-10-C-0237 ('CTSRD') and FA8750-11-C-0249 ('MRC2'). The views, opinions, and/or findings contained in this article/presentation are those of the author(s)/presenter(s) and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

SRI International

UNIVERSITY OF CAMBRIDGE

Welcome to the Computer Laboratory

- When the building catches fire, you want to go **down the stairs and out the front door**
- We are in room FW11
- Tea/coffee/food is in room FW09
- Room FW08 is also available as a breakout room
- Toilets are on the ground floor opposite the café

The (working) plan

- 10:30 start
- 12:30(ish) lunch
- Afternoon(ish) finish (15:00? 17:00?)
- Dinner if you would like to join us (tell us soon)

- Talks with discussion
- Followed by discussion, white boarding, and general schmoozing

THE WORKSHOP

Why are we here?

- CHERI is strongly influenced by historic microkernel thinking
  - E.g., HYDRA, PSOS, Mach, etc.
- "The Machine" NewOS workshop in ETH Zurich:

  How could architectural capabilities affect microkernel design?

- CHERI brings several architectural features to the table:
  - Fine-grained **memory protection** within address spaces
  - Scalable software **compartmentalization** with efficient memory sharing
  - Strong **compatibility** with MMU-based VM / C-language model
- But CHERI microkernels mean different things to different people

Hybrid architecture → hybrid OS?

- **Multi-address-space OS**
  - + CHERI memory protection within tasks – and kernel?
  - + CHERI for kernel bypass on capability operations
  - + CHERI to compartmentalize within microkernel
- **Single-address-space OS**
  - + CHERI compartmentalization model – some MMU use
  - + CHERI compartmentalization model – MMU for full-system virtualization only
  - + CHERI compartmentalization model – no MMU use
- And what about convergence with **language runtimes**?

Goals

- Help all understand the CHERI baseline
- Identify common interests across a broad set of microkernel research communities
- Explore models for architectural capability use in microkernel and microkernel-like designs
- Discuss potential collaborations to explore those
- Ponder whether there are architectural gaps in CHERI that could/should be filled
- Think more fundamentally about capabilities

INTRODUCTIONS

Current talklet list

CHERI introduction + architecture	Robert Watson
CHERI models	Simon Moore
CHERI C + compiler	David Chisnall
CheriBSD OS model	Robert Watson
CheriABI process model	Brooks Davis
CheriBSD compartmentalization model	Robert Watson
CheriOS microkernel	Hadrien Barral
CHERI models and proofs	Peter Sewell
CHERI and The Machine	Dejan Milojicic
L4 capability model	Adam Lackorzynski
SeL4 capability model	Matthew Grosvenor
Barrelfish capability model	Timothy Roscoe
Language capability models	Ben Laurie