Kernel Memory Management in Verified Small Kernels

Dhammika Elkaduwe
dhammikae@cse.unsw.edu.au

Advisor: Kevin Elphinstone

seL4 Team:
Kevin Elphinstone
Philip Derrin

L4.Verified Team
Gerwin Klein
David Cock
Thomas Sewell
Formally assured microkernel for systems requiring strong security guarantees

- Formally assured
  - Abstract model
  - Abstract model facilitate reasoning
  - Kernel code must be rigid

- Deployable in variety of system
  - Diverse requirements
  - Example
    - Partitioning
    - Temporal guarantees
    - Share resources ...
  - Kernel should support and enforce the appropriate policy
Kernel Memory Management

• How to manage kernels physical memory?
  – Cache [EROS, Cache kernel] – No temporal predictability
  – Static allocation – Not suitable for dynamic systems
  – Quota – Underutilisation
  – Modifying the kernel – breaks refinement

• seL4 Model: Exports all memory allocation/deallocation decisions to user
  – No implicit allocations within the kernel
  – Kernel memory is represented as first class objects
    • Capabilities are used to confer authority
  – Inspired by early capability machines [Cap system]
  – Allocation takes place only on explicit user request
Advantages

- Supports diverse policies by modifying user-level code
- Supports co-existing policies
- Confinement of authority guarantees confinement of physical memory

Status:

- Formal proof of spatial partitioning
- Haskell prototype & C/C++ version of the kernel
- Performance evaluation/refinement – on going research
Thanks

- Travel fundings: **InfoSys Technologies Ltd.**

Thanks!