9/30/22

(Referee) Communication vs. Common Service (Glue)

granularity of sharing
policy for controlled sharing

same interface

Topic of Today : Isolation

→ What do we want to isolate? and from whom?

① Process vs Process

② Process vs OS (kernel)

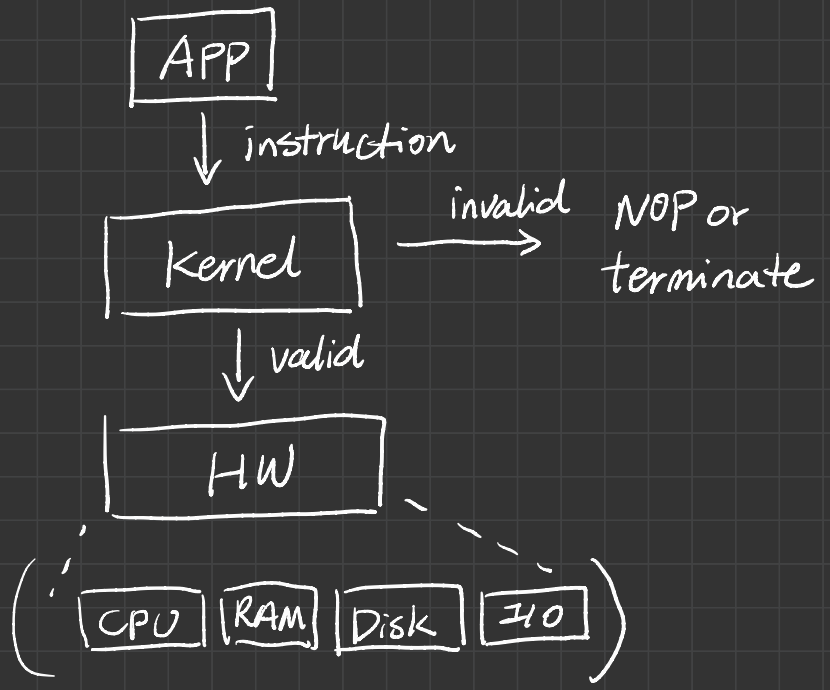
→ Mechanism to provide isolation

☆ Implement this in real life is too slow!

Approach 1 (Hypothetical)

Simulation

- What instr. are invalid?
  - halt
  - instr. that set up memory translation for processes
  - disable interrupts



☆ Common Approach to improve performance

→ Speed up the Common case

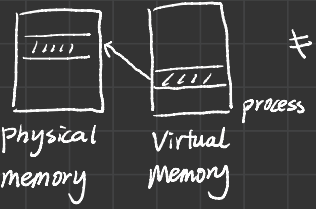
Hardware support!

Valid instr.

Class Question: memory translation?

0x20000 in process A

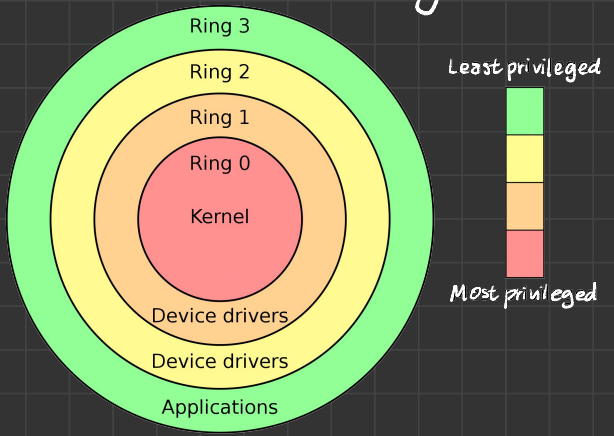
≠ 0x20000 in process B



Hardware Support For User Kernel Isolation

→ Priviledge Levels

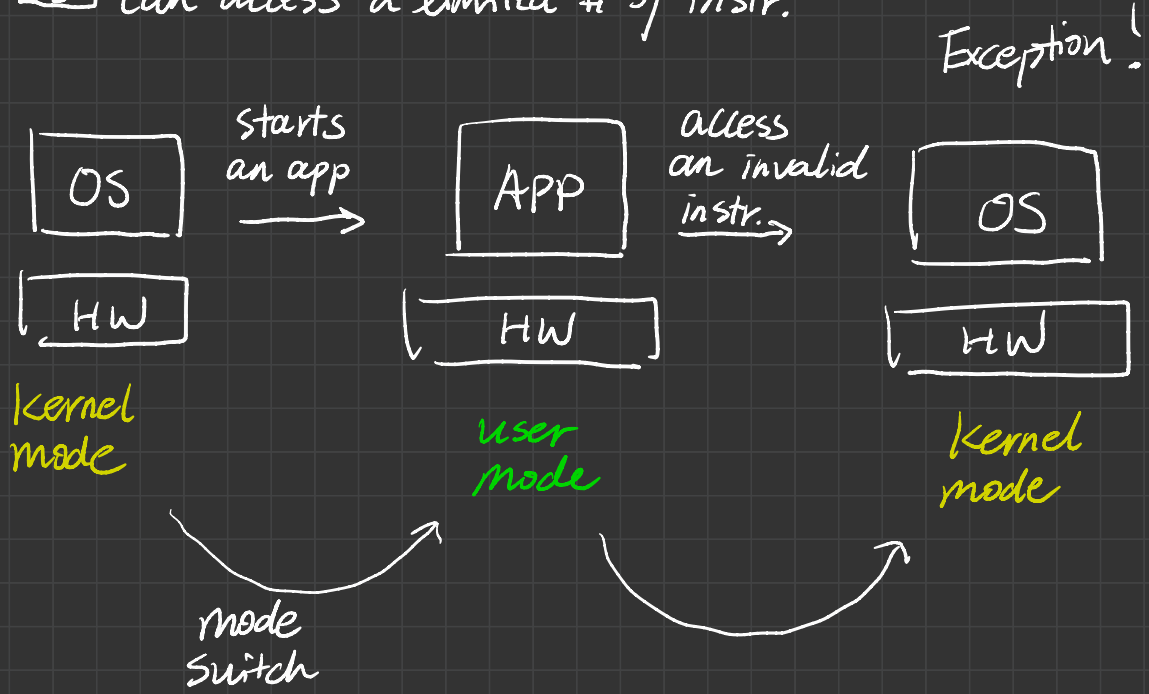
→ CPU register CS (16 bit, lower 2 bits = current priviledge level)



- Ring 0 = Supervisor mode / system / kernel
- Ring 3 = User mode, applications run here!

Dual Mode Execution

- 0 can access any instr.
- 3 can access a limited # of instr.



Exception :

- mismatching privilege level (terminate)
- Null ptr, divide by zero (report to application)

☆ Synchronous (triggered by the instruction)

→ What actually happens?

atomic step by the HW {
- mode switch to the kernel
- save the current process state (PC, register, stack)