# Software Fault Isolation

# Software Fault Isolation [Whabe et al., 1993]

**Goal**: confine apps running in same address space

- Codec code should not interfere with media player
- Device drivers should not corrupt kernel

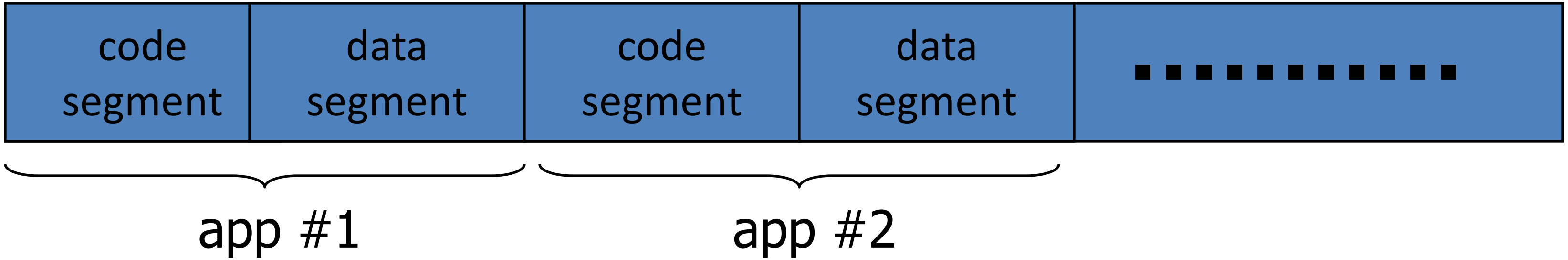
Simple solution: runs apps in separate address spaces

- Problem: slow if apps communicate frequently
  - requires context switch per message

# Software Fault Isolation

SFI approach:

- Partition process memory into segments



- Locate unsafe instructions: **jmp, load, store**
  - At compile time, add guards before unsafe instructions
  - When loading code, ensure all guards are present

# Segment matching technique

- Designed for I
- **dr1, dr2**: d
  - compiler p
  - **dr2** contains segm
- Indirect load instruct **R12 ← [R34]** becomes:

Guard ensures code does not load data from another segment

```
dr1 ← R34
scratch-reg ← (dr1 >> 20)        : get segment ID
compare scratch-reg  and  dr2    : validate seg. ID
trap if not equal
R12 ← [dr1]                      : do load
```

# Address sandboxing technique

- **dr2**: holds segment ID
- Indirect load instruction **R12 ← [R34]** becomes:

```
dr1 ← R34  &  segment-mask      : zero out seg bits
dr1 ← dr1  |  dr2               : set valid seg ID
R12 ← [dr1]                     : do load
```

- Fewer instructions than segment matching
  … but does not catch offending instructions
- Similar guards places on all unsafe instructions

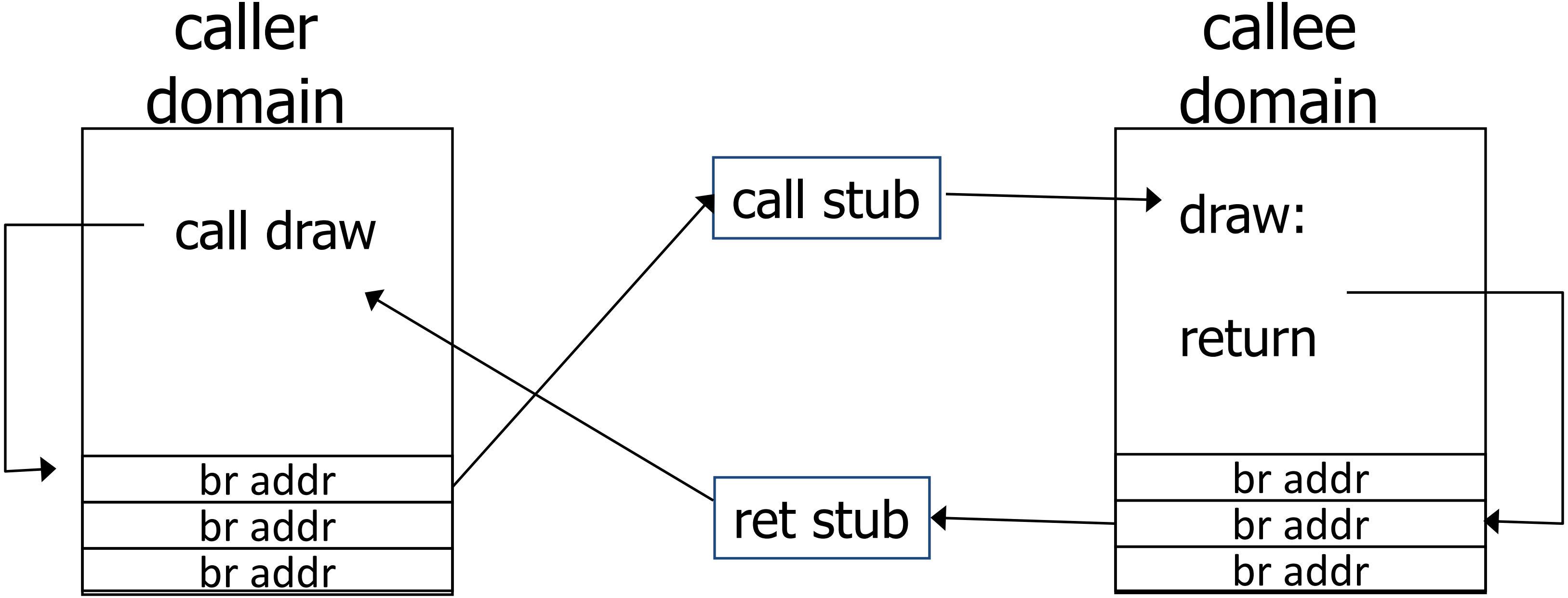
**Problem**: what if **jmp [addr]** jumps directly into indirect load?

**(bypassing guard)**

**Solution:**

**jmp** guard must ensure **[addr]** does not bypass load guard

# Cross domain calls



- Only stubs allowed to make cross-domain jumps
- Jump table contains allowed exit points
  - Addresses are hard coded, read-only segment

# SFI Summary

- Shared memory: use virtual memory hardware
  - map same physical page to two segments in addr space

- Performance
  - Usually good: mpeg_play, 4% slowdown

- Limitations of SFI: harder to implement on x86 :
  - variable length instructions: unclear where to put guards
  - few registers: can't dedicate three to SFI
  - many instructions affect memory: more guards needed