JumpSwitches: Restoring the Performance of Indirect Branches In the Era of Spectre

Nadav Amit, Fred Jacobs, Michael Wei

July 2019
Spectre: Speculative Execution Vulnerabilities
Speculative Execution CPU Vulnerabilities

CALL *R10

branch predictor

OS kernel
Speculative Execution CPU Vulnerabilities

CALL *R10

R9 = [uptr + R8]
R8 *= 64
R8=*secret

branch predictor

OS kernel

CPU cache

This Photo by Unknown Author is licensed under CC BY-SA
Spectre v2 – Unrestricted Indirect Branch Speculation

- prediction
  - do_good()
- misprediction
  - do_no_evil()
  - leak_data()
Current Solution: Retpolines

retpoline(func_ptr) → do_good() → do_no_evil()

misprediction

every indirect branch is mispredicted
JumpSwitches

Dynamic indirect branch promotion

Mechanisms to reduce Retpoline overheads by:

- **Learning targets** on the fly
- **Binary rewriting** the targets
- Supporting **multiple** hot targets
- and **per-context** targets
Macro-Benchmarks on Linux

Normalized Performance

sysbench  dbench  nginx  redis

unprotected  retpoline  JumpSwitch
Security #1: Kernel

Today at 5:10, Track II