第九讲：进程和线程

第 9 节：进程地址空间与熔断 (meltdown) 漏洞

向勇、陈渝

清华大学计算机系

xyong,yuchen@tsinghua.edu.cn

2020 年 4 月 12 日

提纲

1. 第 9 节：进程地址空间与熔断 (meltdown) 漏洞
   - 背景知识回顾
   - 熔断漏洞
   - 进程用户态和内核态的隔离

侧信道攻击

假设有 abc 三个变量，其中 a 没有访问权限，但是 b 和 c 可以访问，此时执行下面这个条件表达式：

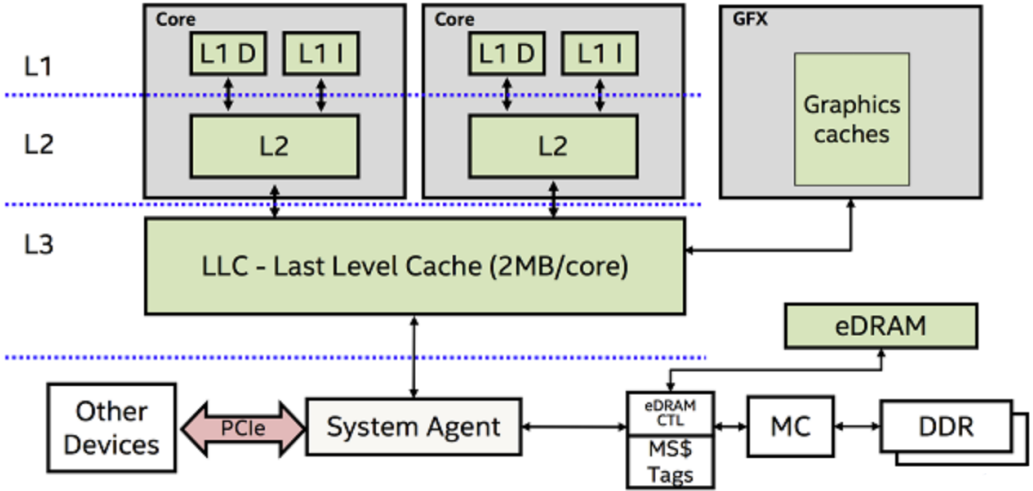
```
1    x = a ? b:c
```

- a 无法访问，系统会直接报错！
- 采用多流水线的 CPU 在检查 a 的访问权限时，继续往下执行。
- a 的权限检查完成时，CPU 已依据 a 的值完成了 b 或者 c 的读取，只是还没有赋值给 x。

侧信道攻击的影响

- 虽然表达式报错，但再次访问变量的速度会变快；
- 后续访问 b 和 c 时，依据访问时间长短，可猜到 a 的值；
- 这个问题导致了 2018 年元旦前后熔断漏洞 (CVE-2017-5754: meltdown)；
- 需要操作系统来补救这个 CPU 设计问题：KAISER
- 类似问题不止这一个……

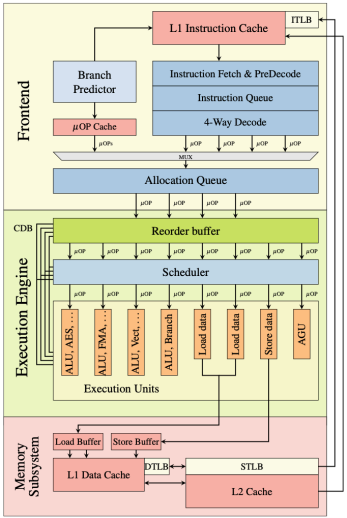
CPU 高速缓存结构 (Intel Skylake)



各级存储结构的访问延迟

访问类型	延迟
L1 cache 命中	约 4 个时钟周期
L2 cache 命中	约 10 个时钟周期
L3 cache 命中	约 40 个时钟周期
访问本地 DDR	约 60 纳秒
访问远端内存节点 DDR	约 100 纳秒

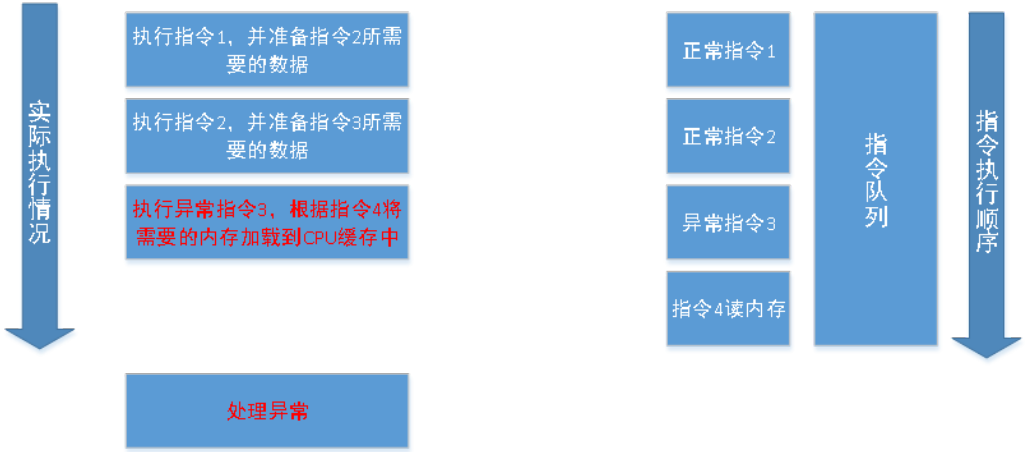
指令执行的乱序优化 (Intel Skylake)



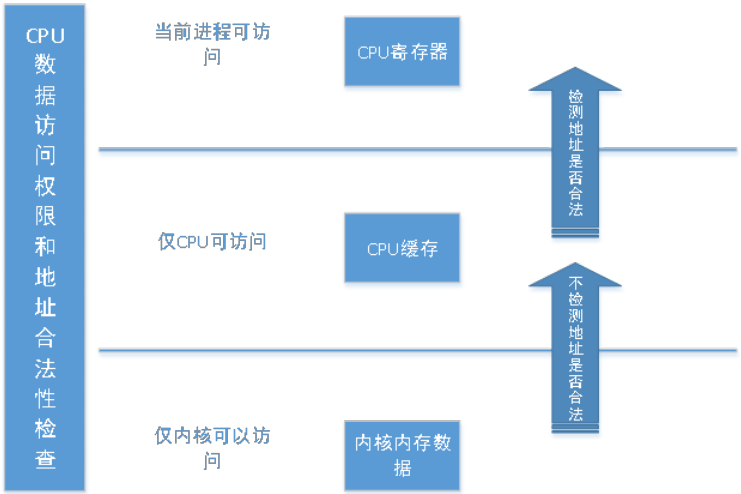
乱序执行过程

- 获取指令和解码：保放到执行缓冲区
- 乱序执行指令：保存在结果序列中
- 退休期 Retired Circle：重新排列结果序列及安全检查（如地址访问的权限检查），提交结果到寄存器

CPU 异常指令执行


实际执行情况
执行指令1，并准备指令2所需要的数据
执行指令2，并准备指令3所需要的数据
执行异常指令3，根据指令4将需要的内存加载到CPU缓存中
处理异常
正常指令1
正常指令2
异常指令3
指令4读内存
指令队列
指令执行顺序

CPU 数据访问权限和地址合法性检查

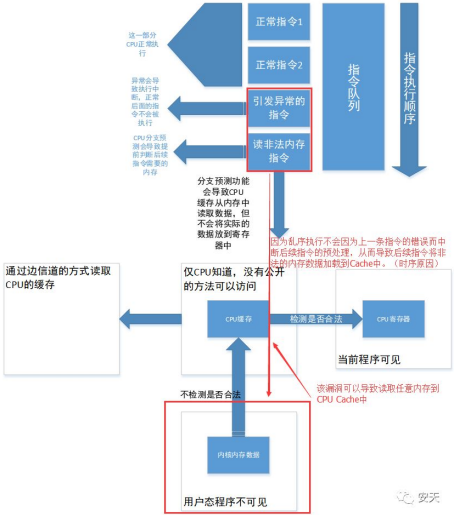


熔断漏洞 (CVE-2017-5754)：核心攻击代码

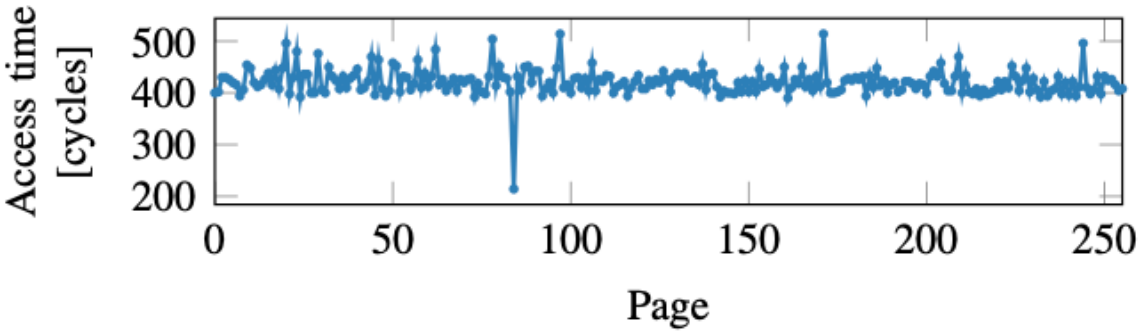
```
1 ; rcx = kernel address
2 ; rbx = probe array
3 retry:
4 mov al, byte [rcx]
5 shl rax, 0xc
6 jz retry
7 mov rbx, qword [rbx + rax]
```

- 假定已分配一块 $2^8 = 256$ 个 4KB 页面大小的内存区域 $(256 * 4096)$ 作为探测数组 (probe array)，并保证该内存块未被缓存；要非法访问的内存地址在 rcx；
- 通过 mov 指令读取一个字节到 rax；该指令会产生异常，但在异常产生前，CPU 已部分完成读取操作；
- 假定读到的值是 i，则 CPU 会继续访问探测数组的第 $i * 4096$ 个元素，导致 CPU 缓存该元素；
- 测量所有 256 个页面内存的访问时间，就可估计出 i 的值。

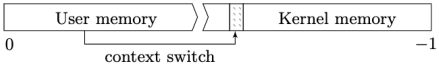
熔断漏洞：攻击过程示意图



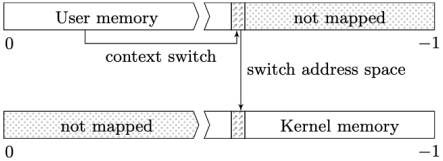
熔断漏洞：在用户态读取内核数据



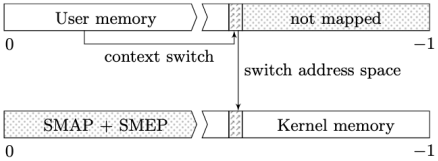
KPTI: Kernel page-table isolation



(a) Regular OS



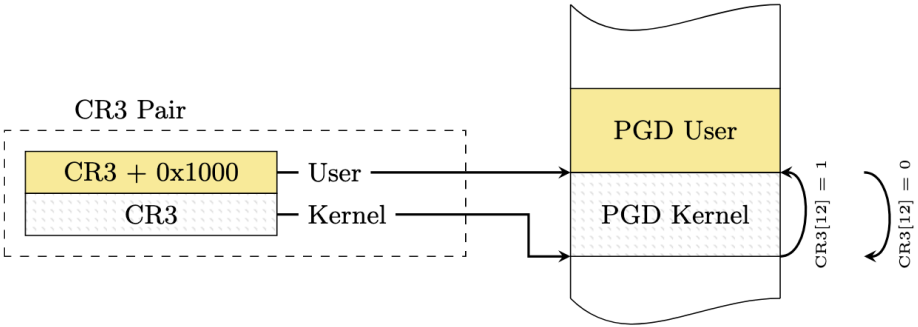
(b) Stronger kernel isolation



(c) KAISER

(a) Kernel is mapped into the address space
(b) Stronger kernel isolation: only interrupt handling code is mapped
(c) KAISER: prevent invalid references (SMAP) and prevent execution (SMEP)

Shadow address space in KAISER



- KAISER: Kernel Address Isolation to have Side channels Efficiently Removed
- PML4 of user address space and kernel addressspace are placed next to each other in physical memory.

“骑士”漏洞（CVE-2019-11157）

- 动态电源管理模块 DVFS（Dynamic Voltage and Frequency Scaling）允许多核处理器根据负载信息采用相应的频率和电压运行，以降低处理器的功耗。
- 当一个核出现电压和频率不太匹配（如电压偏低无法满足较高频率运行需求）时，系统就会出现短暂“故障”。
- 故障对系统行为结果的干扰会泄露出的系统行为信息。