Paper Reading Notes

Jerry

Junyi Yang[1]

PhD. Student at the Institute of Computing Technology, Chinese Academy of Sciences

Website

LinkedIn

Github

yangjunyi22s@ict.ac.cn

[1] *Jerryy959*

Contents

PART

I

介绍

Section 1

大纲

这份笔记主要记录自己阅读的 paper，与对应的总结。分类原则是按照类别进行划分，主要包括体系结构与大模型。

PART

II

文章

Section 2

计算机体系结构

Subsection 2.1

SpecLFB: Eliminating Cache Side Channels in Speculative Executions[1]

Cache side-channel attacks based on speculative executions are powerful and difficult to mitigate. Existing hardware defense schemes often require additional hardware data structures, data movement operations and/or complex logical computations, resulting in excessive overhead of both processor performance and hardware resources. To this end, this paper proposes SpecLFB, which utilizes the microarchitecture component, Line-Fill-Buffer, integrated with a proposed mechanism for load security check to prevent the establishment of cache side channels in speculative executions. To ensure the correctness and immediacy of load security check, a structure called ROB unsafe mask is designed for SpecLFB to track instruction state. To further reduce processor performance overhead, SpecLFB narrows down the protection scope of unsafe speculative loads and determines the time at which they can be deprotected as early as possible. SpecLFB has been implemented in the open-source RISC-V core, Sonic- BOOM, as well as in Gem5. For the enhanced SonicBOOM, its register-transfer-level (RTL) code is generated, and an FPGA hardware prototype burned with the core and running a Linux-kernel-based operating system is developed. Based on the evaluations in terms of security guarantee, performance overhead, and hardware resource overhead through RTL simulation, FPGA prototype experiment, and Gem5 simulation, it shows that SpecLFB effectively defends against attacks. It leads to a hardware resource overhead of only 0.6% and the performance overhead of only 1.85% and 3.20% in the FPGA prototype experiment and Gem5 simulation, respectively.

Remark **翻译**：缓存侧信道攻击[1]基于推测执行非常强大且难以缓解。现有的硬件防御方案通常需要额外的硬件数据结构、数据移动操作和/或复杂的逻辑计算，导致处理器性能和硬件资源的过度开销。为此，本文提出了SpecLFB，它利用微架构组件，即行填充缓冲

[1] 利用 *Cache* 数据推测信息，见常见的侧信道攻击方法，也可参考如下文章 *[2]*

区（Line-Fill-Buffer），并集成了用于加载安全检查的提出机制，以防止在推测执行中建立缓存侧信道。为确保加载安全检查的正确性和即时性，为 SpecLFB 设计了一种称为 ROB 不安全掩码的结构，用于跟踪指令状态。为了进一步减少处理器性能开销，SpecLFB 缩小了不安全推测加载的保护范围，并尽可能早地确定它们可以解除保护的时间。SpecLFB 已在开源 RISC-V 核心 Sonic-BOOM 以及 Gem5 中实现。对于增强的 SonicBOOM，生成了其寄存器传输级（RTL）代码，并开发了一个烧录了核心并运行基于 Linux 内核操作系统的 FPGA 硬件原型。基于通过 RTL 模拟、FPGA 原型实验和 Gem5 模拟在安全保证、性能开销和硬件资源开销方面的评估，它表明 SpecLFB 有效地防御了攻击。在 FPGA 原型实验和 Gem5 模拟中，它导致硬件资源开销仅为 0.6%，性能开销分别为 1.85% 和 3.20%。

2.1.1 复现

```
$ pwd
/home/user
$ ls
file1.txt  file2.txt  script.sh
```

2.1.2 分析

根据摘要，作如下分析与预测：

1. 本文目标是为了解决缓存测信道攻击的问题。
2. 本文的方法利用了 LFB 技术。
3. 设计了一种称为 ROB 不安全掩码的结构，用于跟踪指令状态。
4. 在 Sonic-BOOM 核中进行了实现，结果是实现了 FPGA 硬件并启动了 Linux。
5. 做了数据评估。

Subsection 2.2

FLUSH+RELOAD: a High Resolution, Low Noise, L3 Cache Side-Channel Attack[2]

[2]*Sharing memory pages between non-trusting processes is a common method of reducing the memory footprint of multi-tenanted systems. In this paper we demonstrate that, due to a weakness in the Intel X86 processors, page sharing exposes processes to information leaks. We present FLUSH+RELOAD, a cache side-channel attack technique that exploits this weakness to monitor access to memory lines in shared pages. Unlike previous cache side-channel attacks, FLUSH+RELOAD targets the Last- Level Cache (i.e. L3 on processors with three cache levels). Consequently, the attack program and the victim do not need to share the execution core.*

We demonstrate the efficacy of the FLUSH+RELOAD attack by using it to extract the private encryption keys from a victim program running GnuPG 1.4.13. We tested the attack both between two unrelated processes in a single operating system and between processes running in separate virtual machines. On average, the attack is able to recover 96.7% of the bits of the secret key by observing a single signature or decryption round.

[2]*Notes:* 选择这篇文章是因为*2.1*中介绍了有关缓存侧信道攻击的问题背景，因此找到这篇文章作为入门的该问题的文献。

Remark **翻译**：在非信任进程之间共享内存页面是减少多租户系统内存占用的一种常见方法。在本文中，我们演示了由于英特尔 X86 处理器的一个弱点，页面共享会使进程面临信息泄露的风险。我们提出了 FLUSH+RELOAD，这是一种利用该弱点的缓存侧信道攻击技术，用于监控共享页面中内存行的访问。与之前的缓存侧信道攻击不同，FLUSH+RELOAD 针对的是最后一级缓存（即具有三个缓存级别的处理器上的 L3）。因此，攻击程序和受害者不需要共享执行核心。

我们通过使用 FLUSH+RELOAD 攻击从运行 GnuPG 1.4.13 的受害者程序中提取私有加密密钥来证明 FLUSH+RELOAD 攻击的有效性。我们在单个操作系统中的两个无关进程之间以及在不同虚拟机中运行的进程之间测试了该攻击。平均而言，攻击能够通过观察单个签名或解密轮次恢复 96.7% 的秘密密钥位。

2.2.1 复现

2.2.2 解析

Section 3

人工智能与大模型

References

[1] Xiaoyu Cheng, Fei Tong, Hongyu Wang, Zhe Zhou, Fang Jiang, and Yuxing Mao. SpecLFB: Eliminating cache side channels in speculative executions. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 631–646, Philadelphia, PA, August 2024. USENIX Association.

[2] Yuval Yarom and Katrina Falkner. FLUSH+RELOAD: A high resolution, low noise, l3 cache Side-Channel attack. In *23rd USENIX Security Symposium (USENIX Security 14)*, pages 719–732, San Diego, CA, August 2014. USENIX Association.