**Hardware Vulnerability Research**

**CYBR3020**

**Vulnerabilities and Exploits**

Arr Domingo

Student ID: 200458099

Instructor: Clayton Amelia

Table of Contents

[**Introduction** 1](#_Toc211507336)

[**What is Meltdown** (Report describes the Vulnerability) 1](#_Toc211507337)

[**How meltdown works** (Report explains how the exploit works) 2](#_Toc211507338)

[**What is speculative execution** 2](#_Toc211507339)

[**What is a kernel** 3](#_Toc211507340)

[**What is/was vulnerable** (Report describes what was/is vulnerable) 3](#_Toc211507341)

[**How to Prevent Exploitation** (Report describes how to prevent exploitation) 5](#_Toc211507342)

[References 1](#_Toc211507343)

# **Introduction**

Hardware Vulnerability refer to weaknesses or flaws in the physical components of computing devices, such as processors, memory, and firmware. It has emerged as a critical concern in the rapidly evolving cybersecurity landscape. Unlike software vulnerabilities, which can often be patched or updated, hardware vulnerabilities are more challenging to address because they are literally part of the physical machine. One of the most common hardware vulnerabilities is Meltdown.

# **What is Meltdown** (Report describes the Vulnerability)

With CVE-2017-5754, Meltdown is a vulnerability that leverages the *speculative or out-of-order execution* capabilities of modern Intel CPUs, also known as Rogue Data Cache Load (RDCL) or variant 3 of the CPU speculative execution flaws. This hardware vulnerability works on personal computers, mobile devices, and in the cloud. Every Intel processor which implements out-of-order execution is potentially affected by Meltdown. This vulnerability is the result of a serious design flaw in the affected chips, and the discovery of this issue has led to a redesigning of Windows, Mac, and Linux operating system to mitigate vulnerability and prevent attackers from exploiting it.

On January 3, 2018, Meltdown was publicly disclosed by researchers at Google’s Project Zero which is a team that’s dedicated to finding security flaws before they can be exploited by attackers. As a result of this discovery, security teams at major tech companies like Apple, Intel, and Microsoft, as well as open-source Linux developers are now dedicating heavy resources to try and ensure that their processors and operating systems are secured ahead of any malicious exploits.

# **How meltdown works** (Report explains how the exploit works)

Meltdown is a vulnerability created in the execution of a special low-level code called “*kernel code*”, which runs specifically during a process known as *speculative execution*.

## **What is speculative execution**

As an analogy, imagine a hiker lost in the woods who comes across a fork in the trail creating two roughly parallel paths; one path will get the hiker home, the other will not. Rather than waste time waiting for another hiker to give her directions, she chooses the path she believes is most likely to get her home. At some point on the hike, she comes across a trail marker, if that trail marker informs her that she’s on the right path, then she continues down that path and gets home. If the trail marker tells her she is on the wrong path, she quickly backtracks and hops over to the alternate trail, which leaves her no worse off than if she was still at the base of the trail hoping for directions.

Many modern processors perform a similar technique called speculative execution, where the CPU tries to guess what code needs to be executed next, and then performs that code before being required to do so. If the executed code turns out not to be needed, the changes are reverted. This is meant to save time and speed up performance.

Reports on the Meltdown vulnerability are suggesting that Intel CPUs may be performing speculative execution of code without requiring important security checks. It may be possible to write software designed to check if the processor has completed an instruction that would normally be blocked by these security checks. This mishandling of speculative execution creates a CPU vulnerability which an attacker can exploit to access very sensitive data in kernel memory such as passwords, encryption keys, personal photographs, emails, etc.

## **What is a kernel**

A kernel is the program at the core of a computer’s operating system. It has complete control over the operating system and administers everything from start-up to the handing of memory. The kernel is also responsible for sending data-processing instructions to the CPU (Central Processing Unit). Most CPUs are constantly shifting back and forth between kernel mode and user mode.

In kernel mode, the CPU is executing code that has unrestrained access to the computer’s hardware and memory. This mode is generally reserved for the lowest-level and most trusted operations. Crashes that occur while the CPU is in kernel mode are potentially catastrophic; they can crash the entire Operating System.

# **What is/was vulnerable** (Report describes what was/is vulnerable)

According to Google, every device with an Intel processor chip made after 1995 is affected by Meltdown. Desktop, laptop, and cloud computers which implements out-of-order execution or speculative execution is potentially at risk. ARM chips (e.g. Cortex-A75) were partially affected while AMD CPUs were mostly immune as their memory permission checks happen before speculative execution touches protected memory. Ideally, the issue is the property of CPU microarchitecture, not of a single OS or application.

Also affected are cloud providers which use Intel CPUs and Xen PV as virtualization without having patches applied. Furthermore, cloud providers without real hardware virtualization, relying on containers that share one kernel, such as Docker, LXC, or OpenVZ are affected.

With Meltdown vulnerability, attackers could read any kernel memory that was mapped into a user process’s address space, including:

* Passwords stored in kernel memory
* Encryption keys
* Data from other processes
* Filesystem caches
* OS and driver data structures

Essentially, anything the kernel could see, user-space code could potentially read via Meltdown.

# **How to Prevent Exploitation** (Report describes how to prevent exploitation)

Besides replacing a PC’s processor, the only way to close the vulnerability is to patch the operating system. Apple quietly introduced a Meltdown patch to OSX in early December, while Microsoft released a Windows patch on January 3rd, and Linux developers are still scrambling to put together a patch.

**Kernel page-table isolation (KPTI)** (a.k.a. KAISER) — separates kernel memory from user processes so speculative accesses can’t reach kernel data; implemented in OS kernels (Linux, Windows, macOS) as patches.

**Microcode/firmware updates** and vendor guidance from CPU makers (Intel, AMD, ARM) and OS vendors.

Ongoing research and additional mitigations (hardware changes in newer CPU designs) for related speculative-execution attacks (Spectre family and later variants).

**What you should do now (practical)**

1. Keep your **operating system and firmware (BIOS/UEFI/microcode)** up to date. Vendors released patches in 2018 and later; manufacturers still publish guidance for newer related issues. [Microsoft Support+1](https://support.microsoft.com/en-us/topic/kb4073757-protect-windows-devices-against-silicon-based-microarchitectural-and-speculative-execution-side-channel-vulnerabilities-a0b9f66c-f426-d854-fdbb-0e6beaeeee87?utm_source=chatgpt.com)
2. For servers and cloud instances, follow your provider’s security advisories and apply recommended patches/firmware. [CISA](https://www.cisa.gov/news-events/alerts/2018/01/04/meltdown-and-spectre-side-channel-vulnerability-guidance?utm_source=chatgpt.com)
3. If you manage legacy/embedded devices, check vendor support — some old devices never received mitigations and may remain at risk.

# References

Constantin, L. (2024, July 15). *39 hardware vulnerabilities: A guide to the threats*. Retrieved from CSO: https://www.csoonline.com/article/567525/hardware-and-firmware-vulnerabilities-a-guide-to-the-threats.html

*Installing GitHub Desktop on Debian/Ubuntu [closed]*. (n.d.). Retrieved from StackOverflow: https://stackoverflow.com/questions/73980172/installing-github-desktop-on-debian-ubuntu

*isec-tugraz / meltdown*. (n.d.). Retrieved from Github: https://github.com/isec-tugraz/meltdown

*Meltdown - Bypass Intel's Hardware Barrier Between Applications And The Computer's Core Memory*. (n.d.). Retrieved from Exploit Databse: https://www.exploit-db.com/exploits/43425

*Understanding Hardware Vulnerabilities and Advanced Persistent Threats*. (2024, October 2). Retrieved from Linkedin: https://www.linkedin.com/pulse/understanding-hardware-vulnerabilities-advanced-persistent-giffe#:~:text=Hardware%20vulnerabilities%20refer%20to%20weaknesses,processors%2C%20memory%2C%20and%20firmware.

*What is Meltdown/Spectre?* (n.d.). Retrieved from Cloudflare: https://www.cloudflare.com/learning/security/threats/meltdown-spectre/

<https://www.cyber.gc.ca/en/alerts-advisories/meltdown-and-spectre-side-channel-vulnerabilities>

**Conclusion**

Hardware vulnerabilities pose significant risks to the security and functionality of systems. Addressing these vulnerabilities requires a comprehensive approach that includes secure design, rigorous testing, and robust supply chain management. By understanding and mitigating these risks, organizations can better protect their hardware from exploitation and ensure the integrity and security of their systems. <https://clocked-out.com/what-is-hardware-vulnerabilities/>

Both Meltdown and Spectre use side-channel to obtain the information from the accessed memory location, termed “Kernel-memory-leaking”. While Meltdown breaks the mechanism that keeps applications from accessing arbitrary system memory, Spectre tricks other applications into accessing arbitrary locations in their memory. Leaked information could include passwords stored in a password manager or browser, personal photos, emails, instant messages and documents