Meltdown

Security and Privacy Assignment 1

Jack Cassidy - Student Number: 1432 0816

February 17, 2018

1 Out of Order Execution and Speculative Execution

Most modern CPU's today come equipped with an optimisation technique called 'Out of Order Execution' (OOE). OOE enables latent sections of the CPU pipeline to do meaningful work by reordering instructions so as to minimize the three types of data hazards; Read after Write (RaW), Write after Read (WaR) and Write after Write (WaW).

An extension to OOE is Speculative Execution (SE). SE enables the CPU to pre-emptively execute entire sections of code that are deemed likely to be executed in the future. If the code is executed in the future, then program execution was saved a number of wasted clock cycles, otherwise it is not a huge issues to roll-back any changes. An example of SE is branch prediction. These instructions are referred to as *transient instructions*.

2 Meltdown

Meltdown was the name dubbed to the security breach discovered to be in the vast majority of modern CPU's (post 2010) by [2] at the start of this year, a member of the Spectre family of exploits. It exploits SE to launch a cache side-channel attack to access kernel memory (or *any* physical memory) in a system. A side channel attack is any attack based on information that can be gathered from the physical implementation of a computer system [1].

When a page of memory is read into the CPU cache by a transient instruction, although the instruction may be discarded, the data will remain in the cache. In code, an array access into a byte of arbitrary kernel memory is constructed so that it is executed out of order. The resulting data read is used to *index* into a user-accessible array, which is filled with some junk data and cached. The array access into kernel memory will raise an exception and the transient instruction is not committed. So although we can't access the kernel memory contents directly, we can deduce it by iterating over all of the user array elements and timing the read times. The quickest read time is the array element we indexed into using the kernel memory value, so the index is equal to the kernel byte value. Rinse and repeat this process, known as a Flush+Reload attack [3], and the entire kernel memory can be dumped at a reported speed of 512 kB/s [2].

3 Results of Meltdown

The reason for the importance of Meltdown is that it is baked into the design of most modern CPU's. It is not a software leak that can be effectively patched in an update, it requires a complete redesign of how OOE and SE works as a conclusive fix. A dump of a users kernel memory can reveal passwords, encryption keys etc. However possibly more frightening is that this attack can be applied to cloud computing systems, where thousands of customers sensitive information is susceptible to attack. Chrome and Mozilla have reported that website Javascript can be leveraged to execute such an attack. Intel's official response was to ”replace CPU hardware.”

References

[1] Wikipedia contributors. *Side-channel attack — Wikipedia, The Free Encyclopedia*. [Online; accessed 17-February-2018]. 2018. URL: `https://en.wikipedia.org/w/index.php?title=Side-channel_attack&oldid=822962347`.

[2] Moritz Lipp et al. “Meltdown”. In: *ArXiv e-prints* (Jan. 2018). arXiv: `1801.01207`.

[3] 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)*. San Diego, CA: USENIX Association, 2014, pp. 719–732. ISBN: 978-1-931971-15-7.