1 Summary

InvisiSpec: Making Speculative Execution Invisible in the Cache Hierarchy (2018): The authors detail InvisiSpec, their strategy for defending against hardware speculation attacks in multiprocessors, including Spectre-like attacks and possible future speculative attacks. InvisiSpec works by making the unsafe speculative loads "invisible" in the data cache hierarchy, which is done by reading speculative load data into a new speculative buffer (as opposed to the local caches), waiting until it is safe (or not), then reissuing the safe load to the memory system and loads data into the cache ("make it visible"). Compared to fences, a different Spectre defense, InvisiSpec performs better with less overhead, only slowing down execution by 21% when defending against Spectre, and 72% when defending against futuristic speculative attacks.

2 Strengths

- The authors do a broad case analysis of what their micro-architecture does in different situations in regard to the load queues and speculative buffers (Sec. 6A2).

3 Weaknesses

- It is not clear why validation or exposure transactions are allowed to overlap when designed for Spectre attacks, but exposure transactions cannot overlap when designed for Futuristic attacks (Sec. 5B/D).
- Details regarding the "Futuristic" speculative attacks were vague and they were obscure in explaining how they tested InvisiSpec against Futuristic attacks.

4 Rating: 4

5 Comments

The authors were very thorough in explaining the mechanisms of InvisiSpec and the characterizing the situations in which it would be applicable. In their approach, they not only demonstrated that InvisiSpec would work against Spectre even better than using fences, but they also considered its utility against other speculative execution not yet well characterized. However, in their evaluation, there are tradeoff considerations in implementing InvisiSpec between performance and better security guarantees against *only* Spectre and Spectre-like attacks. The effects of increased execution times and network traffic over their control are likely some of the first limiting factors to using newer strategies against these covert and side-channel attacks, and the broader array of security risks in general. But the authors make it evident here that their architectures are increasingly minimizing that gap so the actual hardware may not be so far off.