Contribution ID: **15** Type: **not specified**

How I started chasing speculative type confusion bugs in the kernel and ended up with 'real' ones

Tuesday, September 13, 2022 10:00 AM (45 minutes)

This talk will illustrate my journey in kernel development as a PhD student in Computer Systems Security. I've started with Kasper, a tool I have co-designed and implemented, that finds speculative vulnerabilities in the Linux kernel. With the help of compilers Kasper emulates speculative execution to apply sanitizers on the speculative path.
Building a generic vulnerability scanner allows finding gadgets that were previously undiscovered by pattern matching with static analysis. Spectre is not limited to a bounds check bypass! Kasper tries to find speculative gadgets and present them in a web UI for developers to analyse. I will also discuss ongoing efforts to improve the precision of the analysis and reason over practical exploitability.

After we found a speculative type confusion within the list iterator macros, I posted a patch set with a suggested mitigation strategy. By looking at different uses of the list iterator variable after the loop, I entered territory of actual type confusions. I will also discuss ongoing efforts in building an automatic tool for the Linux kernel to detect invalid downcasts with container_of since they otherwise stay completely undetected. We would also gladly like to open a discussion with the audience on the interest and welcome feedback from the community.

Primary author: KOSCHEL, Jakob (VUSec Amsterdam)

Presenter: KOSCHEL, Jakob (VUSec Amsterdam)

Session Classification: LPC Refereed Track

Track Classification: LPC Refereed Track