Manual Of The Microarchitectural Attack

Hardware Security

For questions contact:
Mathé @ m.c.hertogh@vu.nl
or
Max @ m.barger2@vu.nl

***** WEEK #2 *****

Goal: Leak the transient result of a Floating-Point operation via Floating-Point Value Injection (FPVI)

This week, instead of only transiently leaking sensitive data (e.g., from the LFB, Staging buffer), you are going to inject controlled data into subsequent instructions. To understand better how, you have to read the Machine Clear (MC) paper, understand what a MC is, what are the different types of MCs, specifically, what is a denormal number, how it is used to trigger a floating-point machine clear, and how a FPMC leads to a transient execution attack called Floating-Point Value Injection (FPVI).

Steps:

1. Read the Machine Clear paper, specifically all the parts concerning the Floating-Point machine clear and the Floating-Point Value Injection attack.
2. Implement `FPVI`, a program that:
   a. Generates two random numbers **X** and **Y** (i.e. via `rdrand`),
   b. Transforms **X** and **Y** to obtain their corresponding denormal versions **dX** and **dY**
   c. Performs the division between **dX** and **dY**
   d. Leak the transient result of the division

For Step 2(b), simply zero out the top 12 bits.

You can find under **`/tmp`** (on all cluster nodes) a program called **`test_operands`** that, when given two floating-point numbers, returns the transient and architectural results of the division, which you can use to compare your leaked transient result against.

(Intermediate) Deadline: Fri, December 5th @ 23:59

Upload all of your code (i.e. the latest version) to Canvas. Even if your code is not fully functional yet, upload what you have on friday. You can still improve your code in the later weeks, but try to keep this to a minimum. Both the version of your code on the intermediate deadline (friday) and your final submission will be taken into account for grading: staying on schedule is preferred. Functionality, performance, and readability of all code submitted will be taken into account during grading.