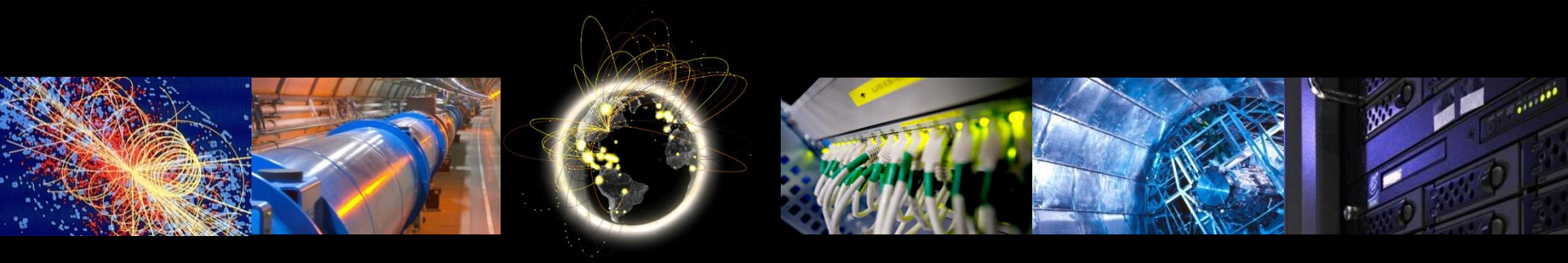
Speculative Execution Vulnerabilities

Spectre/Meltdown: what? How to mitigate?

Vincent BRILLAULT, CERN/EGI-CSIRT

GDB, CERN, Feb 2018



WLCG
Worldwide LHC Computing Grid

CERN

Speculative Execution Vulnerabilities

- Multiple names/naming conventions:
  - Google Project Zero: Variant 1/2/3
  - Press releases: Meltdown & Spectre
  - Official registry: CVE-2017-{5753, 5715, 5754}

- Conventions for this talk:
  - Spectre Variant 1 (CVE-2017-5753)
  - Spectre Variant 2 (CVE-2017-5715)
  - Meltdown (CVE-2017-5754/Variant 3)

Speculative Execution Vulnerabilities II

- Shared principle:
  - Use speculative branches to bypass protections
  - Execution is reverted, but traces remain in CPU caches
- Technical details:
  - Too complex for this talk, focus on mitigations
  - Good talks available, e.g. FOSDEM closing keynote: https://fosdem.org/2018/schedule/event/closing_keynote/
- No full solution without hardware changes
  - But mitigations possible!

Mitigating: Spectre Variant 1

- Bounds Check Bypass
  - Bypass untrusted code execution restrictions
  - Kernel: eBPF JIT compiler
  - Browsers: JS engines
- Vulnerable: Intel, AMD, latest ARM
- Mitigations: add ‘LFENCE’ opcode
  - Kernel: fixed in most distributions (update & reboot required)
  - Browsers: updates to limit attack efficiency
  - No substantial performance impact expected

Mitigating: Meltdown

- Rogue data cache load
  - Speculatively read kernel (protected) memory from userland
- Vulnerable: Intel
- Mitigation: Isolate Userland/Kernel Page Tables (KTPI)
  - Fixed in most distributions (update & reboot required)
  - Potential performance impact, depends on CPU (e.g. PCIDs)
- Straight forward abuse (lots of PoCs)
  - No public weaponized version (yet?)

Mitigating: Spectre Variant 2

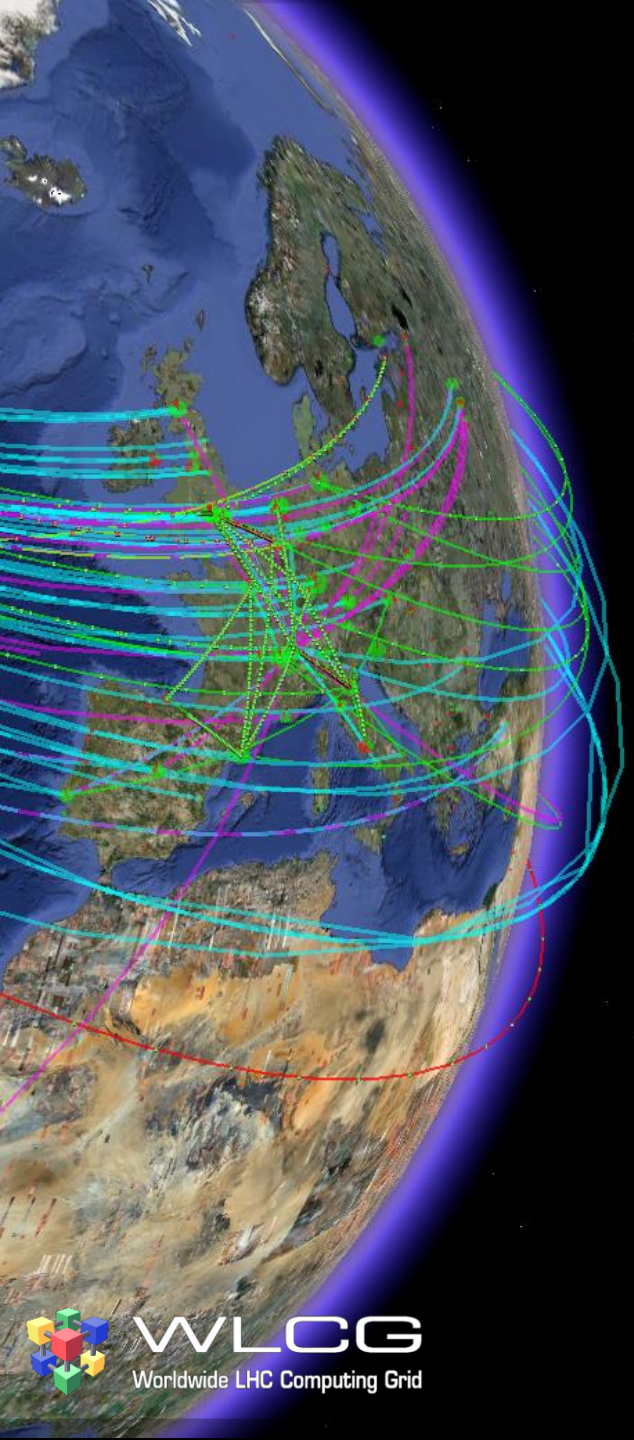
- Branch Target Injection:
  - Trick CPU to speculatively execute your code
  - Kernel, any userland program, **hypervisors**
- Vulnerable: Intel, (AMD)
- Mitigations:
  - IBRS: restrict branch prediction (Intel/Redhat Jan 2018)
    - Kernel & **microcode*** update required
    - Userland protected if ibrs_enabled set to 2
    - Non negligible performance impact (new slow instructions via MSRs)
  - Retpoline: special construct to avoid vulnerability (Upstream)
    - New compiler option (GCC/LLVM)
    - Kernel/userland: enable new options & recompile
    - More issues with specific processor versions (Skylake/Kabylake underflow)

Mitigating: Spectre Variant 2: Microcode

- Jan 3rd: RedHat released 3 microcodes with its patch
- Jan 8th: Intel releases 8 microcodes
  - 2 correspond to RedHat's, Redhat's 3rd not released
- Jan 11th: Intel recognize reboot & instabilities issues
- Jan 16th: RedHat revert microcode updates
  - Recommends getting them from "hardware vendors"
- Jan 22nd: Intel: reboot root cause identified
  - Only for Broadwell & Haswell
- Feb 7th: Intel announces progress on some microcodes
  - Several Skylake-based products
  - Released to hardware vendors

Recommendations

- Update Kernel & reboot
  - EGI deadline was last week, followed up by EGI-CSIRT
- Microcode:
  - Do not update microcode
    - In case of instabilities, downgrade (update package & reboot)
  - Follow your hardware vendor recommendations
  - Follow intel advisory updates: INTEL-SA-00088
    - In particular the "affected products" link (may change)
- Be prepared for further updates, listen to security broadcasts



WLCG
Worldwide LHC Computing Grid