ULBS
Universitatea "Lucian Blaga" din Sibiu

FANTOME CATASTROFICE ȘI CATASTROFE FANTOMATICE ÎN MICROPROCESOARELE ACTUALE

Promoțiile 4 C & TI, 4 ISM

"Ultimul" (micro-)curs, 21 mai, 2018

Profesor univ. dr. ing. Lucian VINȚAN

Membru titular al Academiei de Științe Tehnice din România

Universitatea "*Lucian Blaga*" din Sibiu, România

Computer Engineering Department, "Lucian Blaga" University of Sibiu, Romania
http://csac.ulbsibiu.ro/

acaps research center

Introducere

- Noiembrie 2017: *Brian Krzanich*, *Chief Executive Officer* ***INTEL***, a vândut cca. 80% din acțiunile sale (?!) **...**
- 3 ianuarie 2018: **vulnerabilități grave de securitate în microprocesoarele actuale** (Grup internațional cercetători, *Google*)
- **Procesarea *Out of Order*** *(Meltdown)* respectiv **procesarea speculativă** (*Spectre*)
- ***INTEL, ARM, AMD…***

  miliarde de dispozitive!

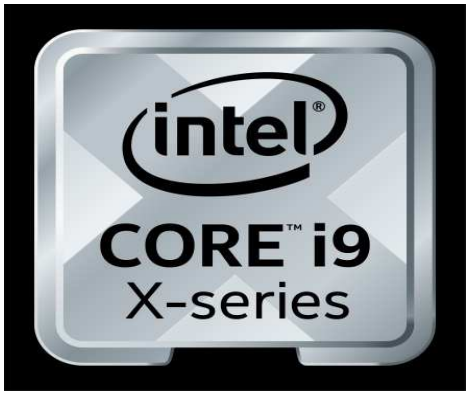
- *"[..] ideas to use neural branch prediction [Vintan,…] have been picked up and integrated into CPU architectures."* Meltdown Paper

Obiectivele prezentării:

- **O idee asupra cauzelor** acestor breșe de securitate
- **Extragerea unor concluzii / probleme deschise…**

Recapitulare necesară înțelegerii (I)

Schema bloc a unui procesor superscalar *OoO*



LOAD R1, (R2)204;
SUB R4, R1, R5; *Fwd_CDB!*

PIPELINE:

I-Fetch → I-Decode & Dispatch → Issue → ALU → D-Mem → Wr-Back → Commit

Recapitulare necesară înțelegerii (II)

Noțiuni fundamentale

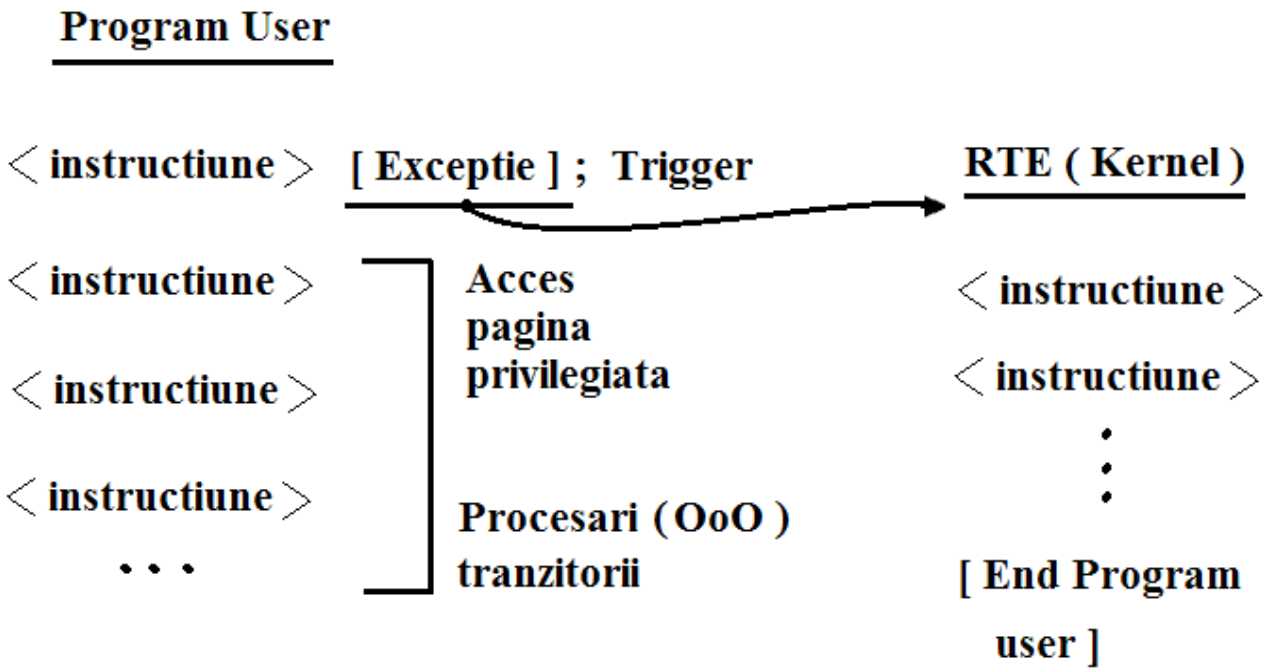
- **Procesarea *Out of Order*** *(TOMASULO)*

```
LD   R5,   (R9)offset;  D-Cache miss!
ADD  R1,   R2,   #26;          OoO!
SUB  R12,  R17,  R19;          OoO!
```

- ***Buffer* de reordonare (*ROB*, multiport)**
- **Stațiile de rezervare (SR)**
- **Ierarhia de memorii *cache***
- ***Dynamic Branch Prediction*** → **procesare speculativă** & ***I.L.P.***
- **Memoria virtuală, două funcții:**
  - **Mapare** dinamică spațiu virtual → memoria principală (DRAM…)
  - **Protecție** a accesului la memorie; izolarea spațiilor de memorie inter-*task*-uri... niveluri de privilegiu, drepturi de acces…

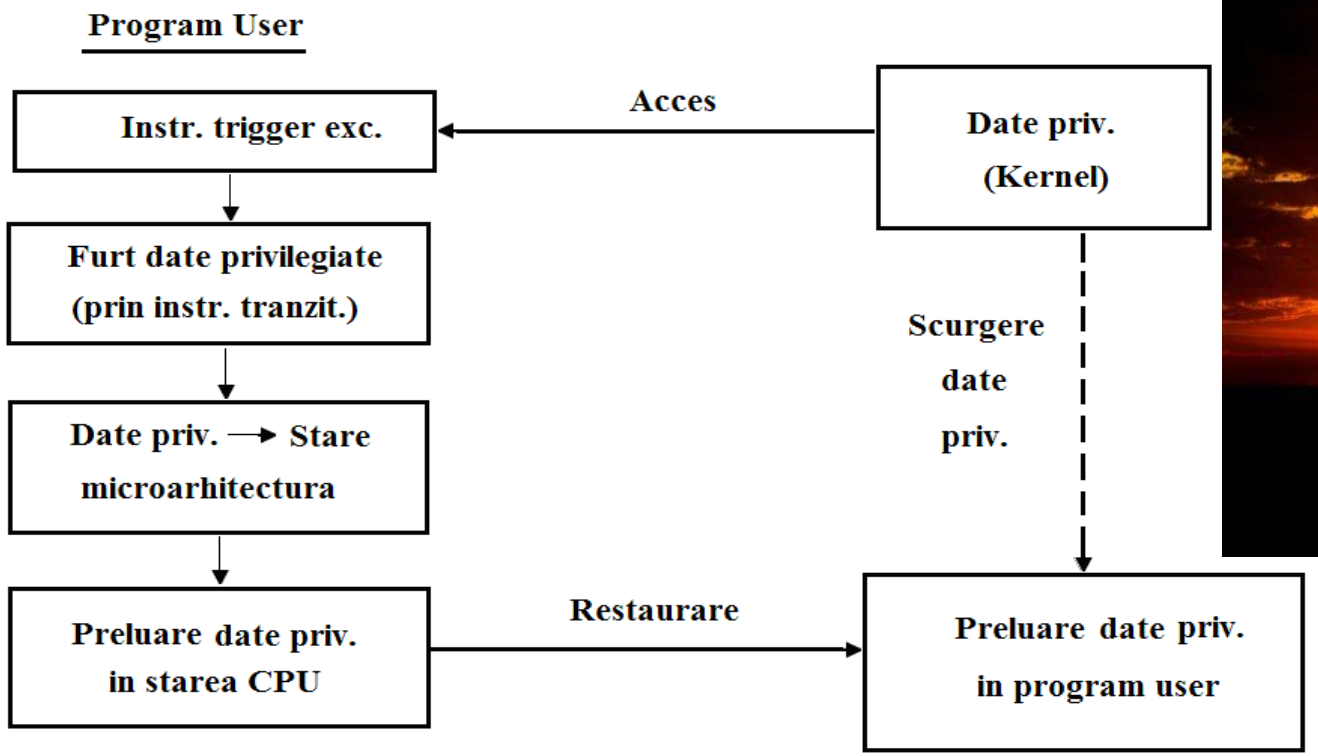
Breșa de securitate *Meltdown* („Catastrofa”)

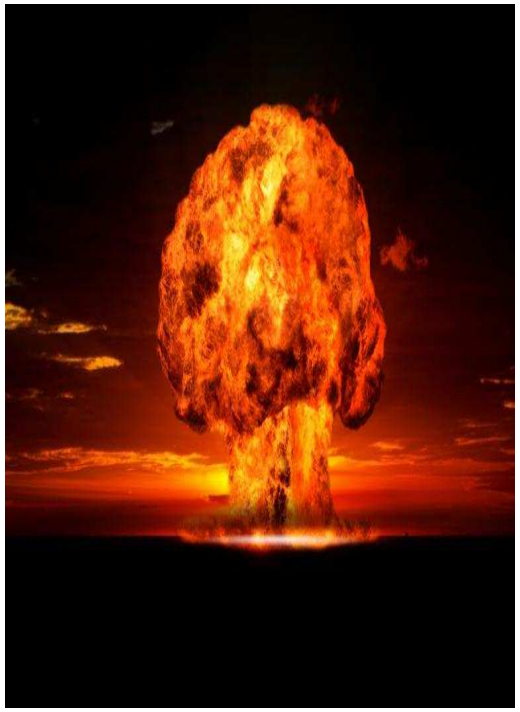
Principiul evidențierii atacului



- **Exploatează execuția *Out of Order***, între acces ilegal al instr. *trigger* - ex. violare de privilegiu (!) - și *RTE → End_User_Program.*
- **Instrucțiunile tranzitorii → transferă data secretă (s) în *Cache***

Meltdown
Schema logică a preluării datelor secrete



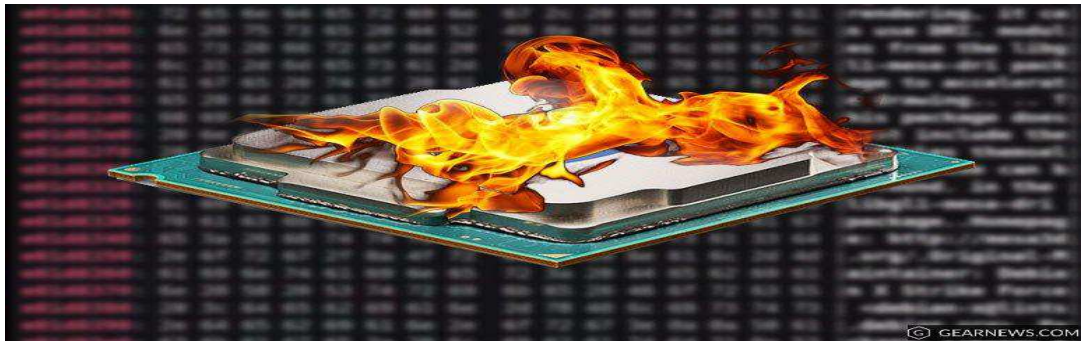


Atacatorul, printr-o aplicație *user* → *fork*:

- **Un fir „copil”** – instrucțiunile tranzitorii accesează date privilegiate (parole, chei criptare etc.) → data secretă (s) → resurse micro-arhitecturale (*Cache*)
- **Un fir „părinte”** – va transfera data secretă (s) din resursele micro-arhitecturii, în cele logice, ale arhitecturii.

Meltdown

Cele 3 faze succesive:



1. Se accesează și se încearcă **încărcarea, într-o locație a ROB și în SR din cadrul CPU, a unui octet (*s*) din memoria privilegiată** → Excepție
2. O instrucțiune tranzitorie accesează pagina nr. ***s*** dintr-un T. P. din memoria *user* (***s***, pe post de *pointer* de memorie) → **alocare data din acea pagină, în *cache*!**
3. Atacatorul declanșează o metodă pentru a **afla linia accesată din *cache* și, implicit, secretul (*s*) memorat** la locația privilegiată.

Meltdown. Exemplu de atac *Intel x86*

rcx = o adresă de memorie situată într-o pagină privilegiată

rbx = adresa de bază a unui "*Tablou de Probă*" (T. P.), situat în memoria utilizator. Inițial, T. P. complet nealocat în *cache*!

1. `mov al, byte [rcx];` — În reg. al_{ROB} ←"octet secret" ***s***
   Violare de privilegiu → Excepție...

2. `shl rax, 0xC;` *Out of Order (OoO)* — Reg. rax_{ROB} (*Fwd_CDB*) va conține:
   s$\cdot 2^{12}$ = ***s***$\cdot$1000H = ***s***$\cdot$4096

3. `mov rbx, qword [rbx + rax];` *OoO* — **Alocă în *cache* un singur bloc de date, din pagina nr. *s* a T. P.**

PG. 0 (4 KO)	PG. 1 (4 KO)	...	PG. s (4 KO)	...	PG. 255 (4 KO)

***"Tabloul de Probă"* (T. P.)**

Aflarea valorii secrete (*s*)

- Alt *thread* atacator **va citi secvențial, primul bloc din fiecare pagină a T. P.** → *[rbx + k*1000H], k=0, 1, 2, ..., 255*.
- **Măsurare timp de acces pentru fiecare bloc.** Citirea din pagina nr. **s** a T. P. → timp minim acces!
- **S-a aflat s!** Iterare secvența anterioară pentru diferite adrese de pagini privilegiate *[rcx]* → ***Memory Dumping* pentru orice *task*!**

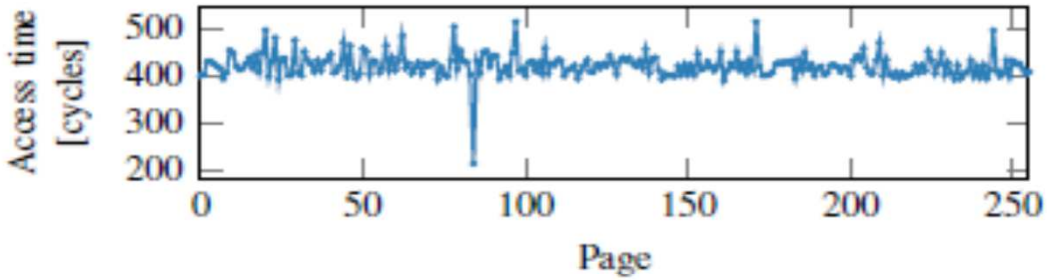
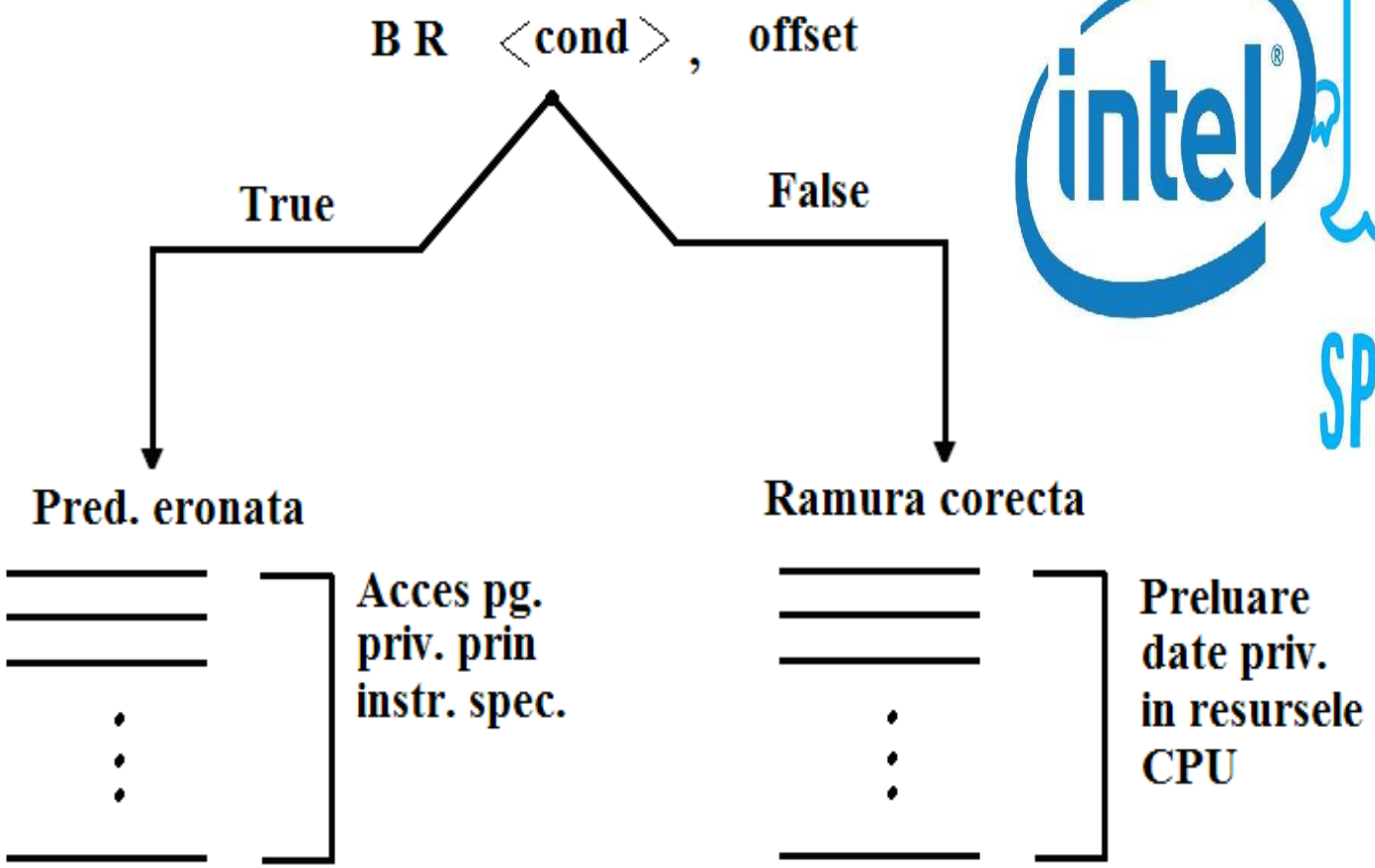


Figure 4: Even if a memory location is only accessed during out-of-order execution, it remains cached. Iterating over the 256 pages of `probe_array` shows one cache hit, exactly on the page that was accessed during the out-of-order execution.

Breșa de securitate *Spectre* („Fantoma”)

Exception Suppression...



Spectre. Alt exemplu de atac

Atacatorul: apel funcție bibliotecă de nivel privilegiat (ex. *syscall…*)

Predictor *branch*-uri va prezice eronat ramura *True (x < array1_size)*

if (x < array1_size)

y = array2[array1[x] * 256]; *x > array1_size!*

- *array1[x]* = data secretă **s**
- *array2* nealocat în *cache*
- → **(1)** Se citește speculativ, cu *hit* în *cache*, data secretă **s** (*array1[x]*).
- → **(2)** Se citește speculativ (*miss*) locația nr. **s** din tabloul *array2*, **doar pentru ca data respectivă să se aloce în *cache*!**
- **(3)** Un alt fir de control (apel) va **citi cele 256 locații *array2[n*256]***, *n = 0, 1, …, 255.* Doar o citire va fi rapidă, cea aferentă **locației nr. *s*!**

Concluzii (I)

- **Alocarea date privilegiate în *cache*, prin instrucțiuni tranzitorii** - *OoO*, speculative -, datorată unor **erori de proiectare *hardware***
- **Posibile soluții, în opinia mea: citirile *in order* (1; reduc *IPC*)** sau *branch recovery* → **invalidare date alocate speculativ în *cache* (2).** Tardiv? Interzicere alocări în *cache* prin instr. tranz.
- **Scrierea în pagini privilegiate?** Teoretic imposibilă; scrierile *in order*!
- Accesarea ***cache* cu mapare directă** → instrucțiuni subsecvente speculative. **Ar putea fi oare și acestea utilizate abuziv?**
- Soluțiile software (*KAISER…*) pot ameliora problemele, dar nu le pot elimina complet. **Re-proiectare *hardware* a CPU → Timp!**
- **Ingeniozitate, inspirație, talent, intuiție adâncă... de admirat!**

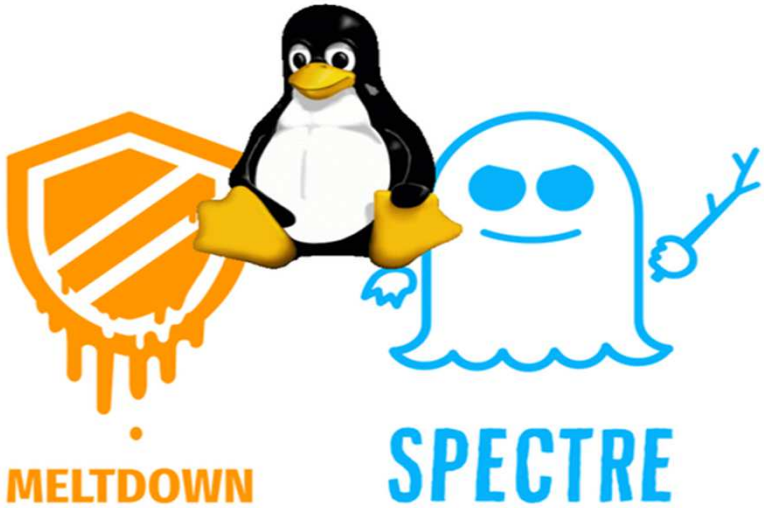
Concluzii (II)

- **Performanță, energie, complexitate... vs. securitate**
- **CONCEPTE ESENȚIALE: vecinătate temporală, spațială respectiv a valorilor (1), predicție și speculație (2), concurență și paralelism (3), virtualizare (4)** etc. Erorile de proiectare *hardware* au apărut tocmai **la interfața acțiunilor subtile dintre aceste concepte fundamentale.**
- **Cum s-ar putea formaliza acești piloni arhitecturali în vederea dezvoltării unei științe autentice (deductiv-axiomatice) a sistemelor de calcul?**
- **Acest "micro-curs" → doar un imbold spre studiu și aprofundare a acestor probleme, la nivel individual**
- **Cum mai putem înțelege și stăpâni complexitatea actuală?**

Și la final, câteva gânduri

- **→ Necesar un <u>fundament științific și cultural</u> solid. (Empirism!)**
- **→ Gândire <u>critică</u>! <u>Gândiți</u> permanent, cu propria minte!**
- **<u>"Faceți"</u> (*Homo faber*) dar și <u>înțelegeți</u>... sau măcar <u>încercați</u> să înțelegeți...**
- **Ce <u>repere umane</u> mi-a oferit studiul, știința, cultura?** *"Știința fără conștiință este ruina sufletului!"* (*Rabelais*)
- **Educația → cultivarea minții și a sufletului → <u>complexitatea</u>, <u>frumusețea</u> și <u>transcendența</u> acestei lumi**
- **Dincolo de "înșurubarea în iluzii" bune...**
- **→ Căutați-vă, fiecare, un <u>sens superior al vieții</u>, prin care <u>să lăsați o "urmă" bună în această "*trecere-petrecere*"</u>...**
- **Onorarea statutului de ființă umană**

Vă mulțumesc! Succes în profesiune, succes în viață!



Promoțiile 4-C & TI, 4-ISM
Ultimul "curs", 21 mai 2018