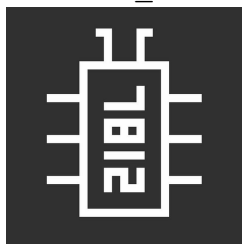
# Introduction to software-based microarchitectural side-channel attacks

Abc Xyz
@dura_lex




DCG#7812
2018

# Agenda

# Agenda

Introduction
- Side-channel attacks
- Microarchitectural attacks

# Agenda

Introduction

- Side-channel attacks

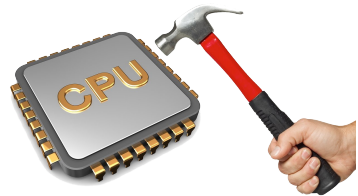
# Side-channel attacks


Example of target for side-channel attack

# Agenda

# Microarchitectural attacks

```
code1a:
  mov (X), %eax
  mov (Y), %ebx
  clflush (X)
  clflush (Y)
  jmp code1a
```




The DRAM cells get permanently damaged if hammered for a long time

# Agenda

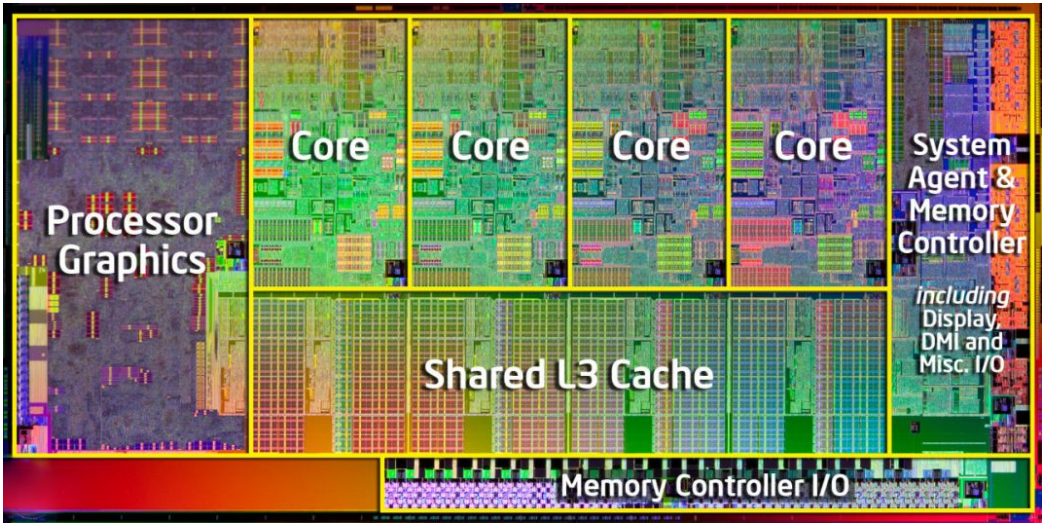
Theory
- CPU
- Cache
- DRAM

# Agenda

Theory

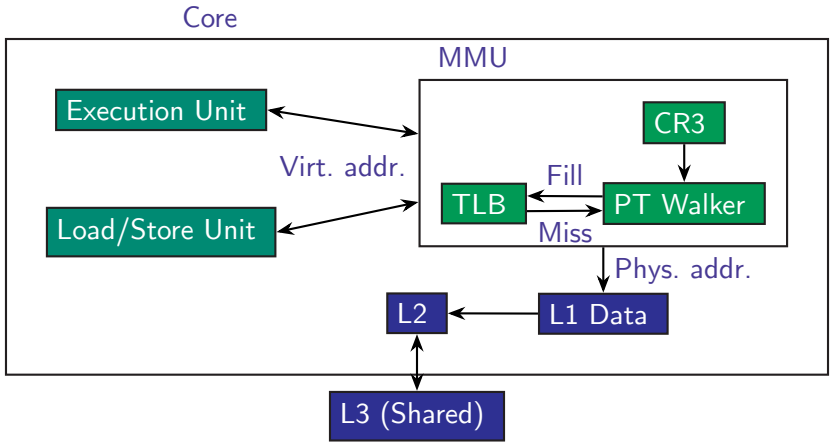
- CPU
  - Pipelining
  - Branch Prediction and Speculation
  - Multicore

# CPU



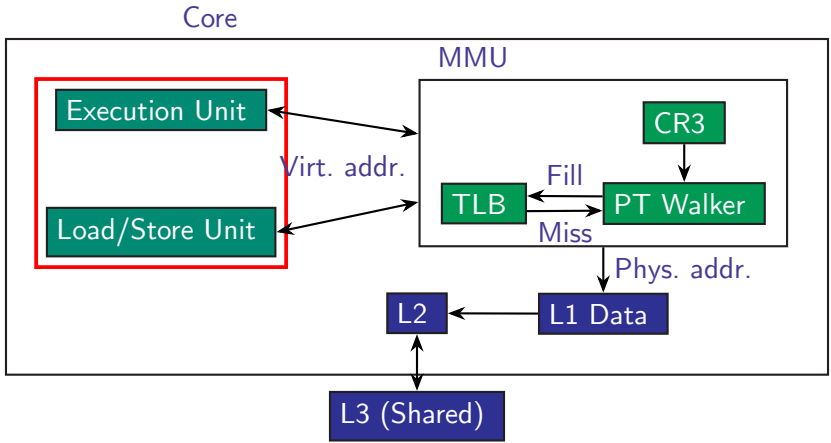

Architecture of multicore CPU

# CPU



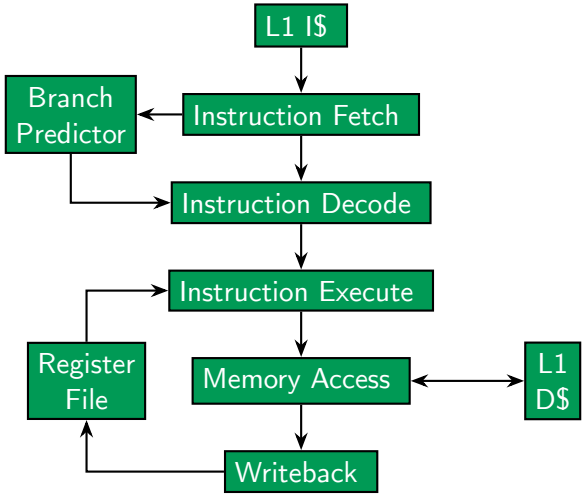

Abstract architecture of core and memory organization

# CPU



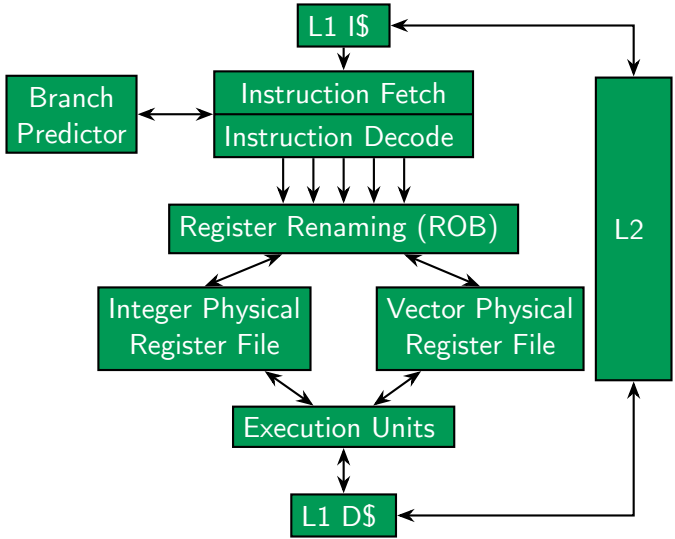

Abstract architecture of core and memory organization

# Pipelining. In-Order



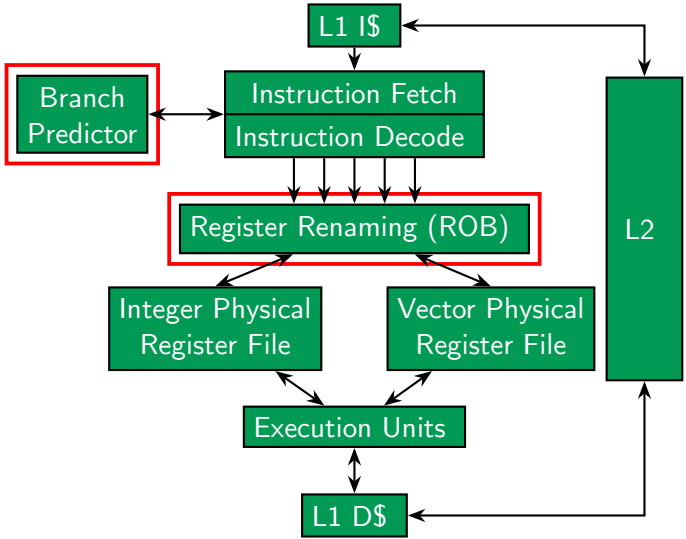

Elements of a modern in-order core

# Pipelining. Out-of-Order



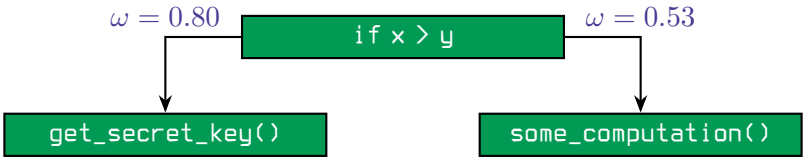

Elements of a modern out-of-order core

# Pipelining. Out-of-Order



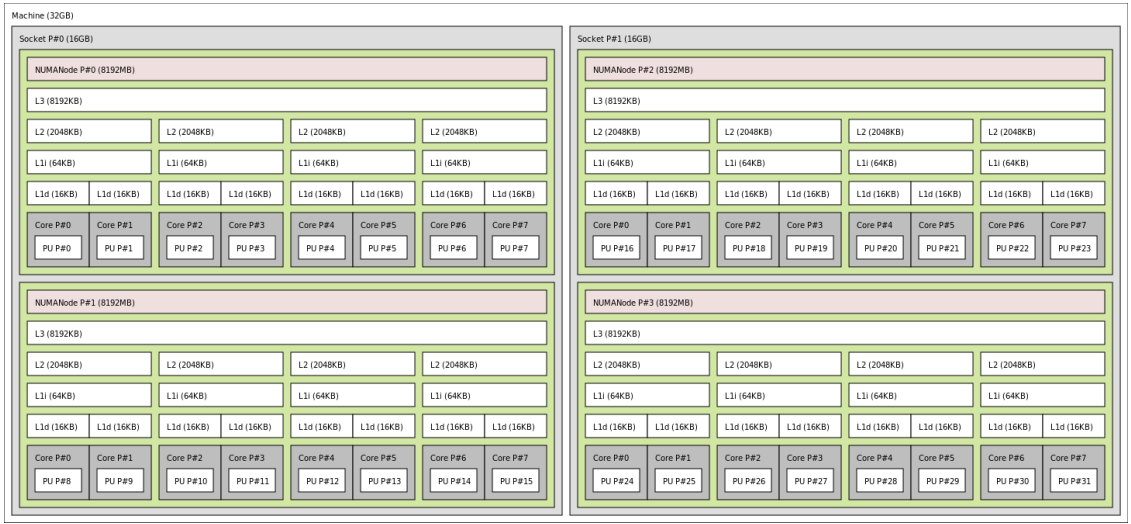

Elements of a modern out-of-order core

# Branch Prediction and Speculation




`get_secret_key()` can be executed speculatively

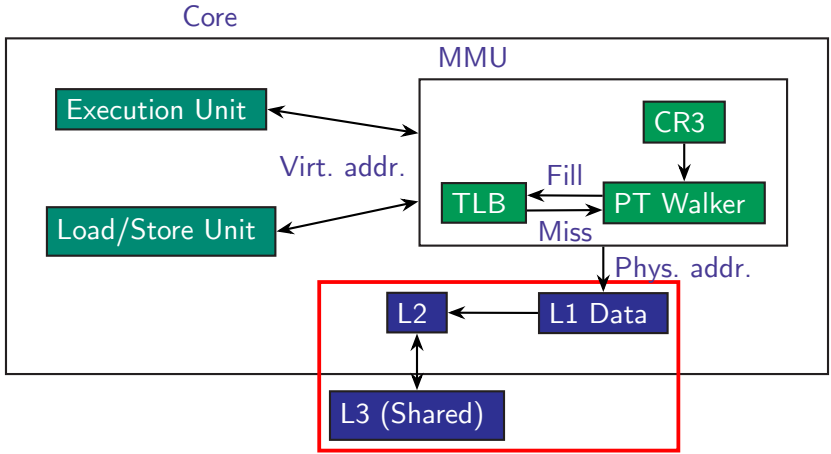
# Multicore




Architecture of multicore CPU AMD Bulldozer

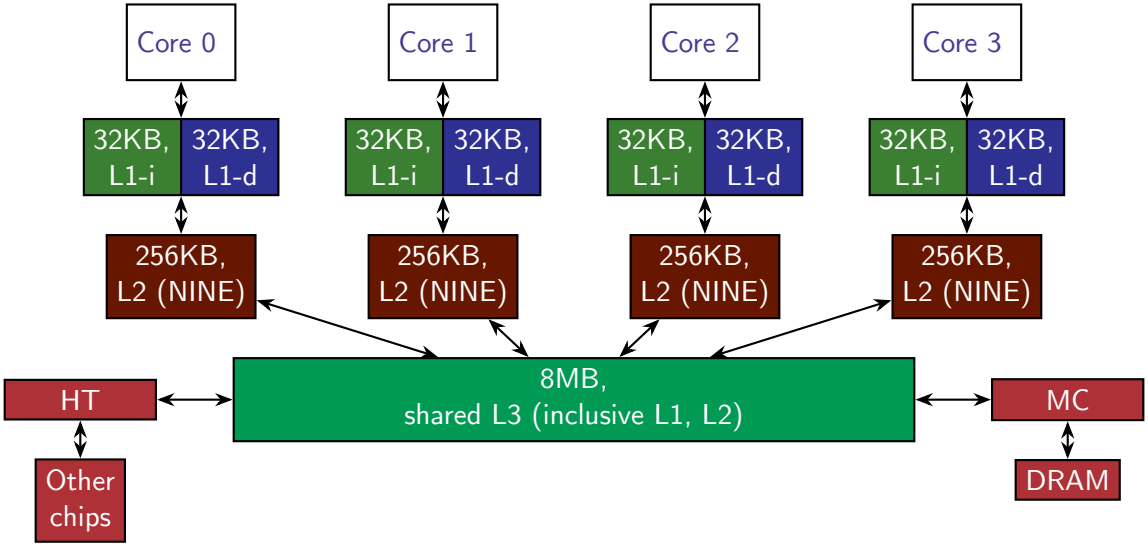
# Agenda

# Cache



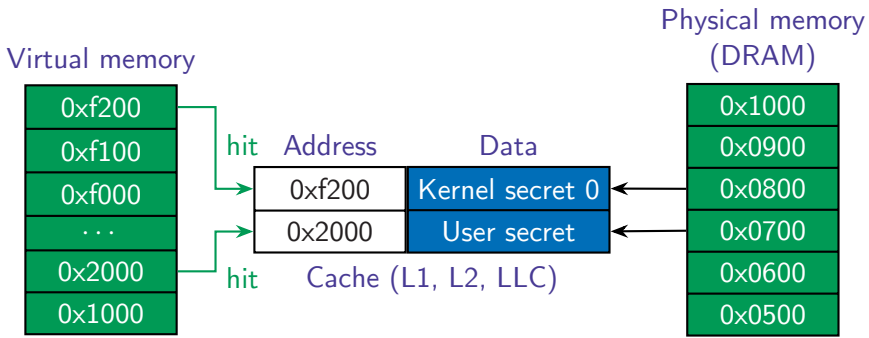

Abstract architecture of core and memory organization

# Cache



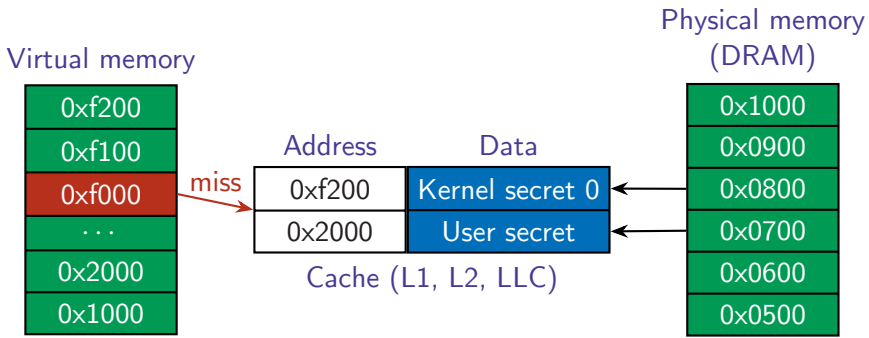

The flow of data through a modern platform

# Cache



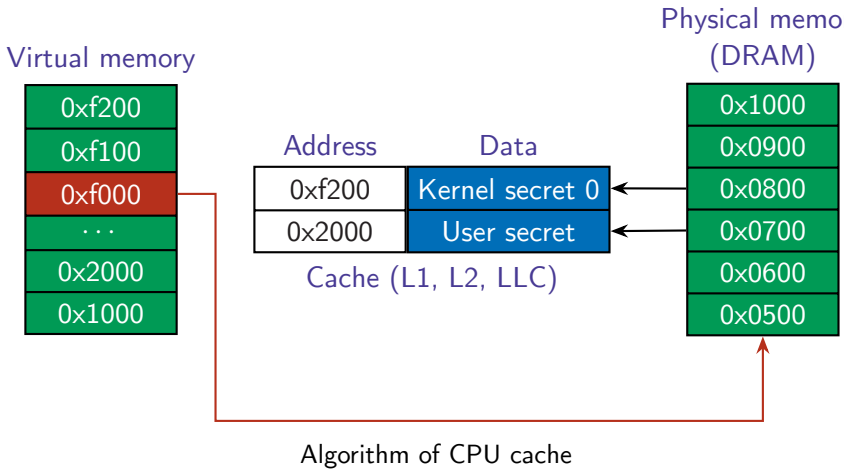

Algorithm of CPU cache

# Cache

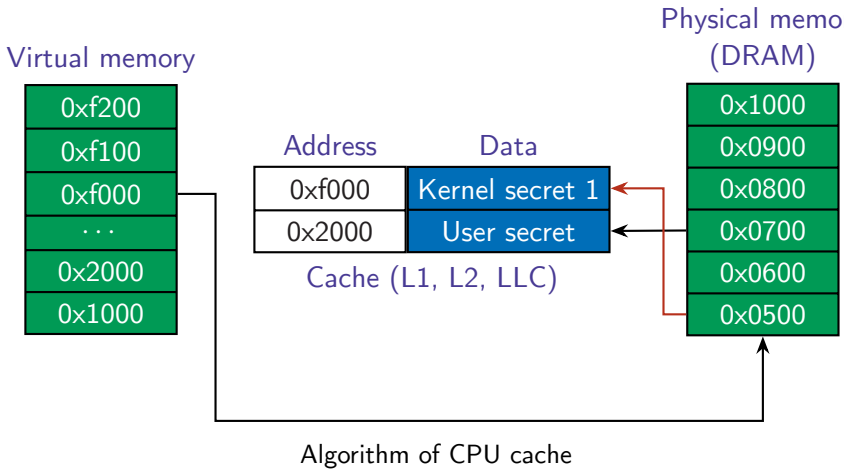



Algorithm of CPU cache

# Cache


Virtual memory
0xf200
0xf100
0xf000
...
0x2000
0x1000
Address
Data
0xf200
Kernel secret 0
0x2000
User secret
Cache (L1, L2, LLC)
Physical memory
(DRAM)
0x1000
0x0900
0x0800
0x0700
0x0600
0x0500
Algorithm of CPU cache

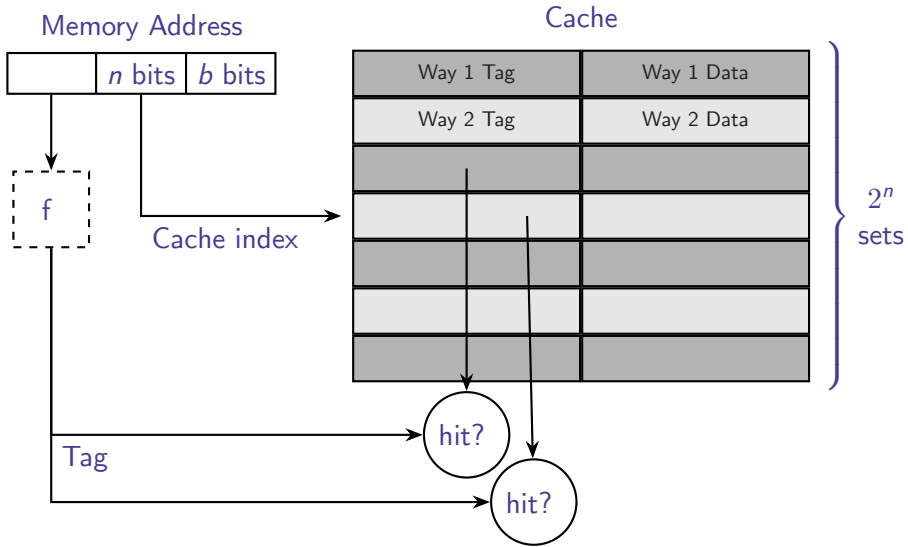
# Cache


Virtual memory
0xf200
0xf100
0xf000
...
0x2000
0x1000
Address
Data
0xf000
Kernel secret 1
0x2000
User secret
Cache (L1, L2, LLC)
Physical memory
(DRAM)
0x1000
0x0900
0x0800
0x0700
0x0600
0x0500
Algorithm of CPU cache

# Types of cache

- Direct-mapped cache
- Fully-associative cache
- 2/4/8/12-way set associative cache

# Two-way set associative cache



# Cache replacement policies

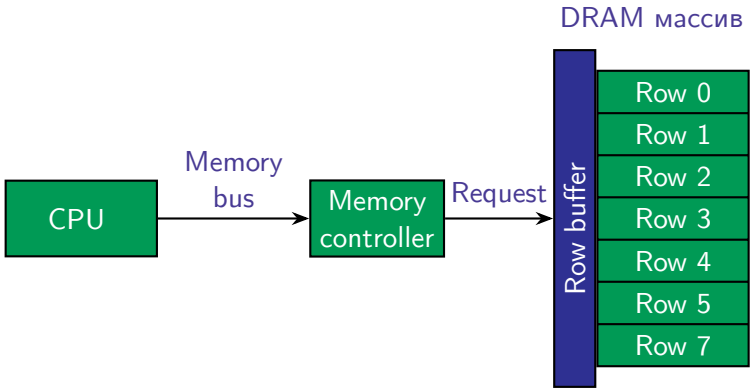
- FIFO
- LIFO
- least recently used, LRU
- time aware least recently used, TLRU
- most recently used, MRU
- pseudo-LRU, PLRU
- random replacement, RR
- segment LRU, SLRU
- least frequently used, LFU
- least frequent recently used, LFRU
- LFU with dynamic aging, LFUDA
- low inter–reference recency set, LIRS
- adaptive replacement cache, ARC
- clock with adaptive replacement, CAR
- multi queue, MQ
- and etc.

# Addressing modes

- Virtually indexed, virtually tagged (VIVT)
- Physically indexed, virtually tagged (PIVT)
- Virtually indexed, physically tagged (VIPT)
- Physically indexed, physically tagged (PIPT)

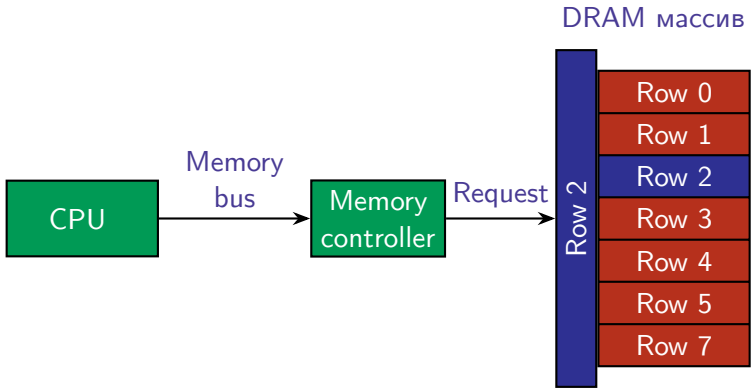
# Agenda

# How DRAM works




A very simple computer system, with a single DRAM array

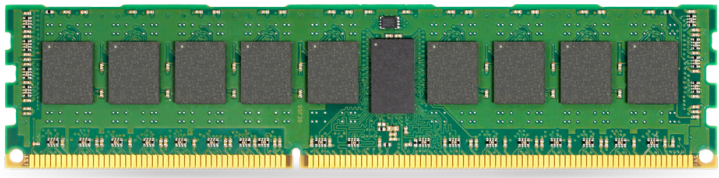
# How DRAM works



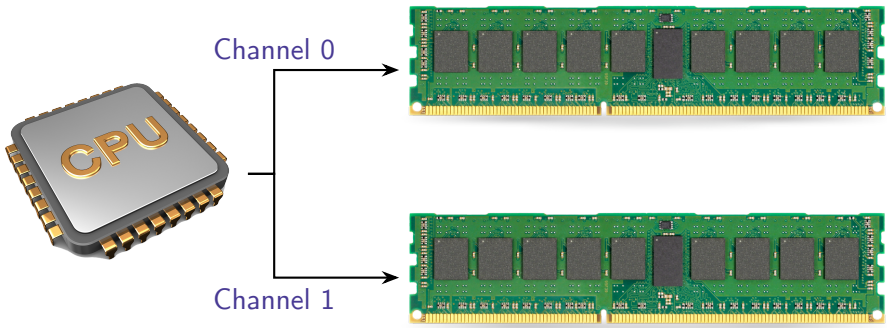

A very simple computer system, with a single DRAM array

# DRAM organization

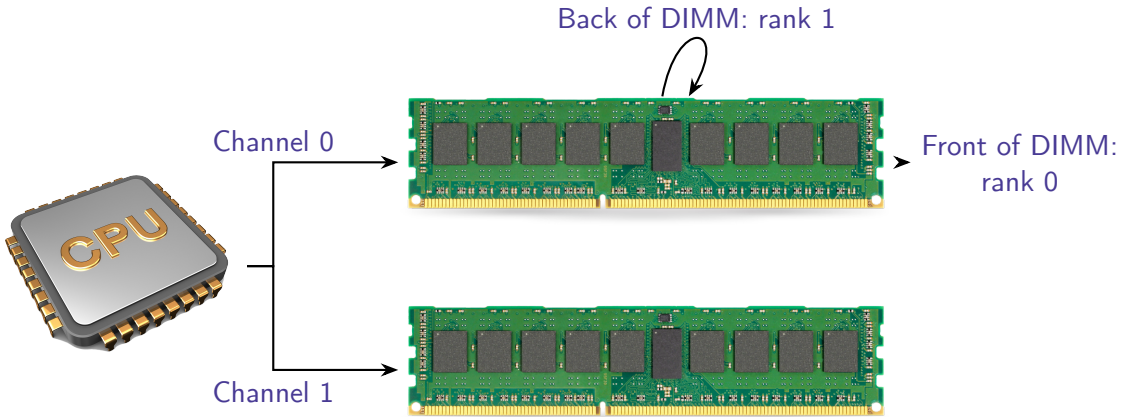
DIMM



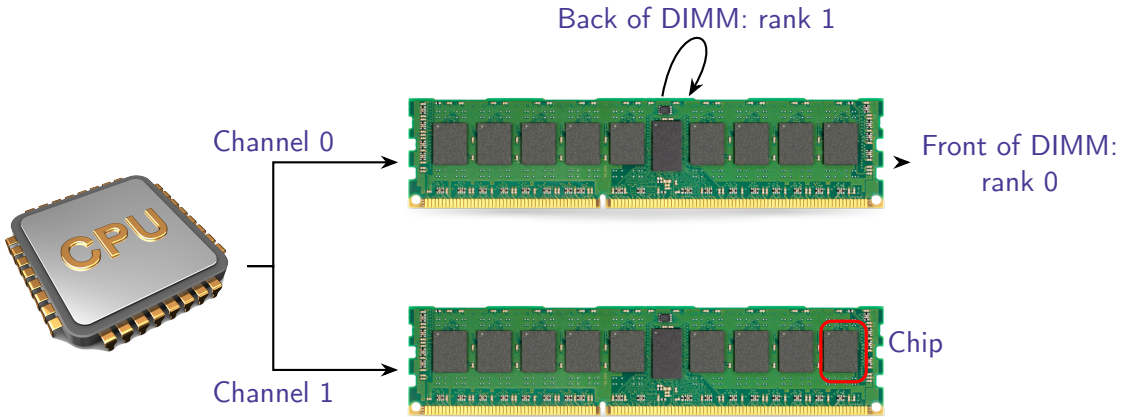
# DRAM organization



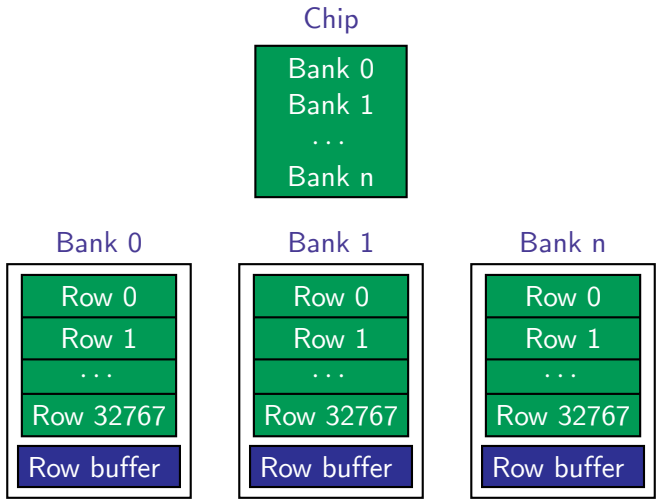
# DRAM organization



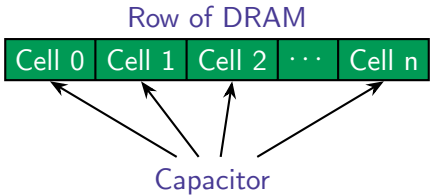
# DRAM organization



# DRAM organization



# DRAM organization



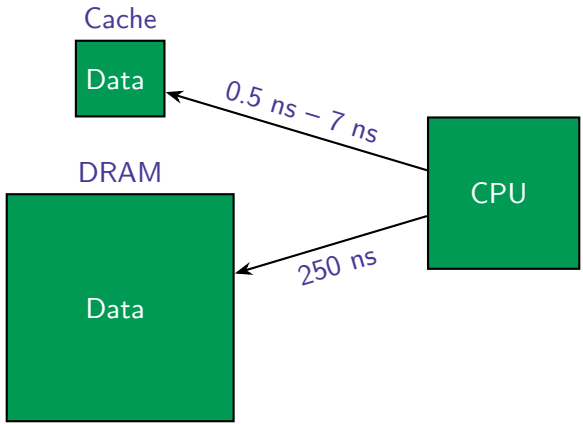
# Agenda

Basic attacks

- Cache attacks
- Branch-prediction attacks
- TLB-based attacks
- Exception-based attacks
- DRAM-based attacks
- Covert channels

# Agenda

# Cache attacks



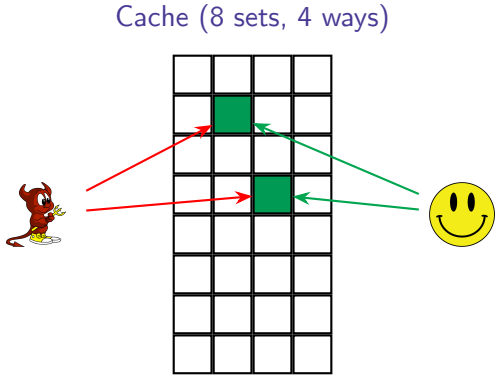

Timing attack — attack exploiting differences in the execution time of an algorithm

# Flush + Reload

1. Map binary (e.g., shared object) into address space
2. Flush a cache line (code or data) from the cache
3. Schedule the victim's program
4. Check if corresponding cache line from step 2 has been loaded by the victim's program

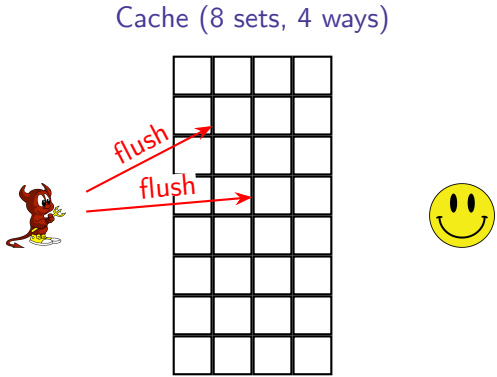
# Flush + Reload

Map binary (e.g., shared object) into address space



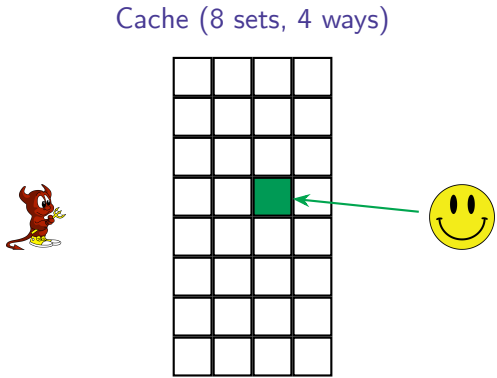
# Flush + Reload

Flush a cache line (code or data) from the cache



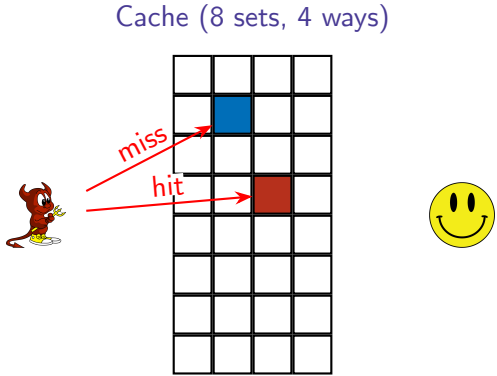
# Flush + Reload

Schedule the victim's program

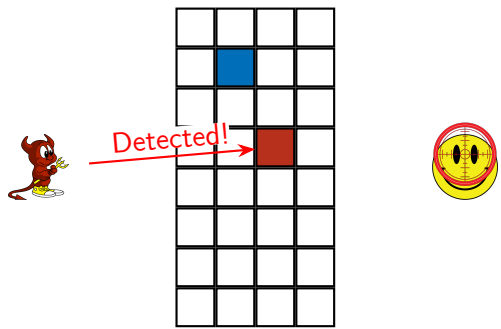


# Flush + Reload

Check if corresponding cache line from step 2 has been loaded by the victim's program



# Flush + Reload



# Cache attacks

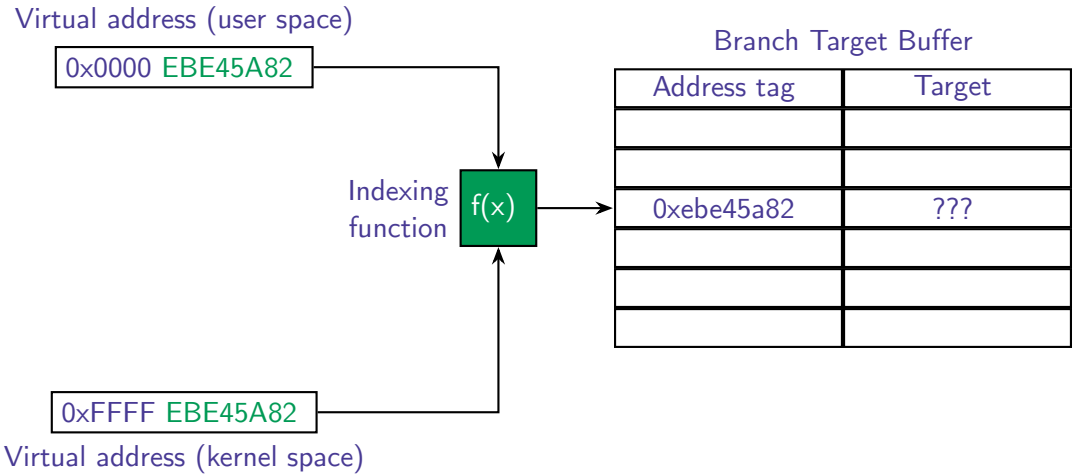
- Evict + Time
- Prime + Probe
- Prime + Abort
- Flush + Flush
- Evict + Reload
- AnC (ASLR $\oplus$ Cache)
- and etc.

# Agenda

Basic attacks

- Branch-prediction attacks

# Branch-prediction attacks



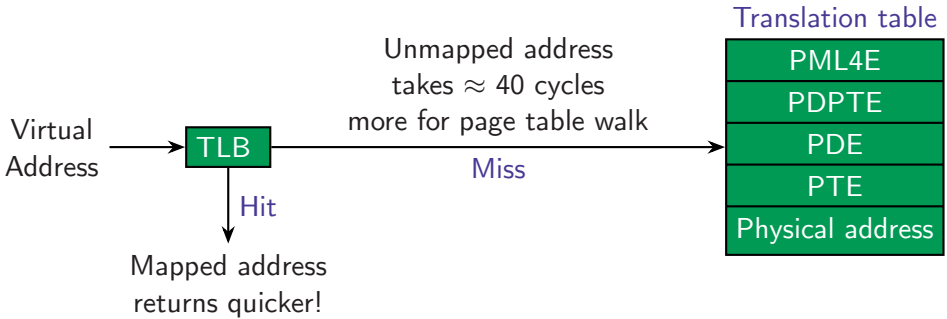

Branch Target Buffer addressing scheme in Haswell processor

# Agenda

Basic attacks

- TLB-based attacks

# TLB-based attacks




A translation lookaside buffer (TLB) is a memory cache that is used to reduce the time taken to access a user memory location

# Agenda

Basic attacks

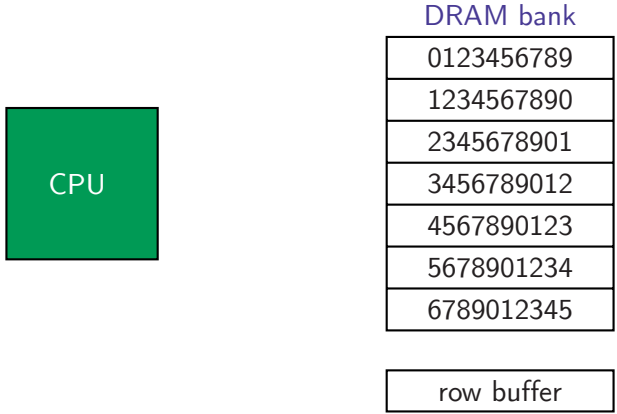
- Exception-based attacks

# Exception-based attacks

- Scheduler interrupts
- Instruction aborts
- Page faults
- Behavioral differences (e.g, error code)

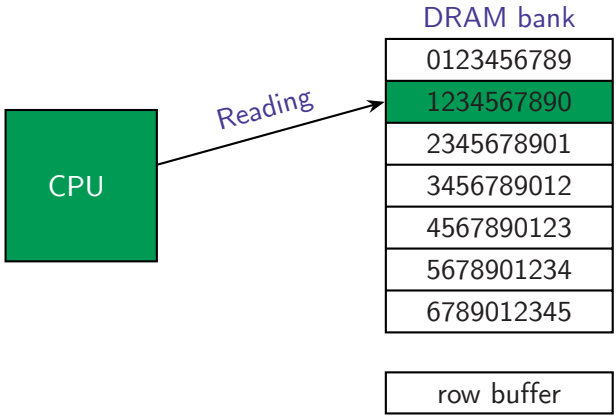
# Agenda

# Reading from DRAM



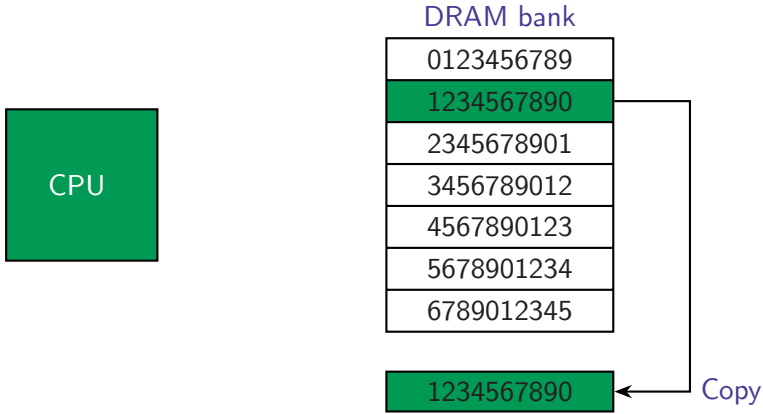

Reading from DRAM

# Reading from DRAM

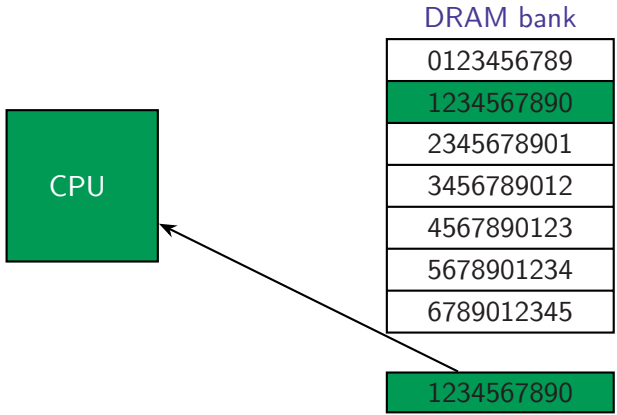



CPU reads row 1, row buffer empty

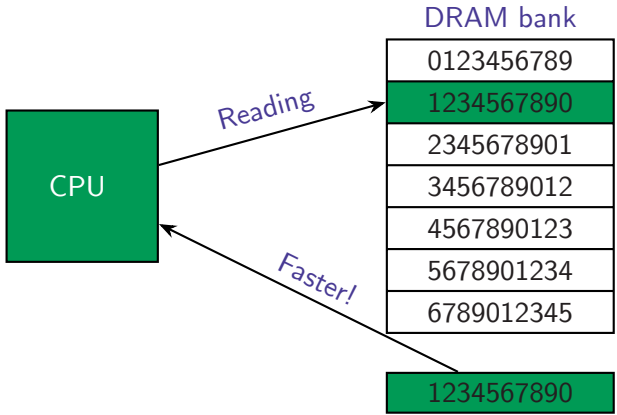
# Reading from DRAM



# Reading from DRAM



# Reading from DRAM




CPU reads row 1, row buffer now full

# Complex DRAM-based attacks

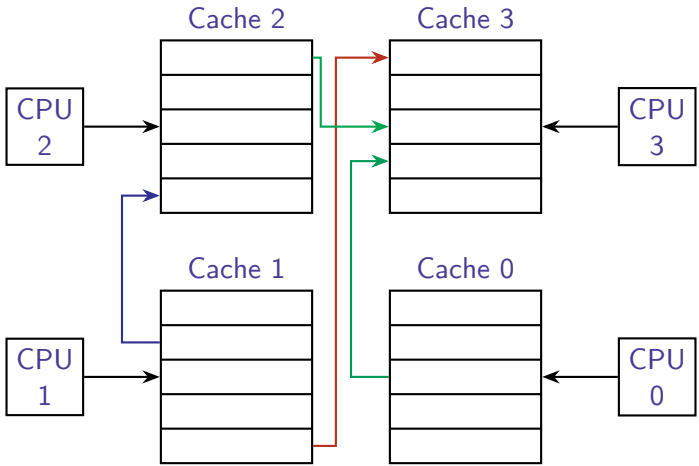
- DRAMA
- Row hit (Flush + Reload)
- Row miss (Prime + Probe)
- and etc.

# Agenda

Basic attacks

- Covert channels

# Covert channels




Cross-core covert channels

# Covert channels

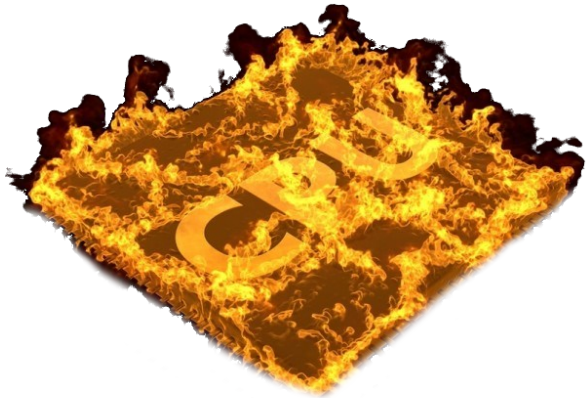
- Cache-based covert channels (shared libraries)
- Row miss attack (DRAM)
- Thermal covert channels
- Radio covert channels

# Agenda

Software-based Microarchitectural Fault Attacks

- Rowhammer

# Software-based Microarchitectural Fault Attacks



Software-based microarchitectural fault attacks do not require physical access, but instead only some form of code execution on the target system

# Agenda

# Rowhammer. Exploitation primitives

- Fast uncached memory access
- Physical memory massaging
- Physical memory addressing

# Variations of Rowhammer

- Flip Feng Shui — targeted Rowhammer
- Throwhammer — remote Rowhammer
- Nethammer — better remote Rowhammer
- Drammer, RAMpage — exploitation ARM-based hardware
- Glitch — better exploitation ARM-based hardware

# Agenda

Meltdown & Spectre

- Derived attacks and not only
- Abstract example of exploitation

# Agenda

Meltdown & Spectre

Derived attacks and not only

# Derived attacks and not only

Spectre-NG

- MeltdownPrime & SpectrePrime
- SgxPectre
- SMM Speculative Execution Attacks
- BranchScope
- LazyFP
- ...

# Derived attacks and not only

- Spectre 1.1, 1.2 (Speculative Buffer Overflows)
- SpectreRSB
- NetSpectre
- L1TF (Foreshadow)
- and etc.

TotalMeltdown? and other patches...

# Agenda

Meltdown & Spectre

- Abstract example of exploitation

# Abstract example of exploitation

The four components of speculation techniques

1. Speculation primitive

# Abstract example of exploitation

The four components of speculation techniques

1. Speculation primitive

- Bypass out of bounds checks
- Training of branch predictor
- Speculatively read an earlier value of the data
- Pending exceptions
- Exploit branch history table
- Exploit the Return Stack Buffer
- Speculatively write to register (buffer overflow)

} Microarchitecture — ?

# Abstract example of exploitation


Foundation of tower
*speculative-based attack*

# Abstract example of exploitation

The four components of speculation techniques

1. Speculation primitive
2. Windowing gadget

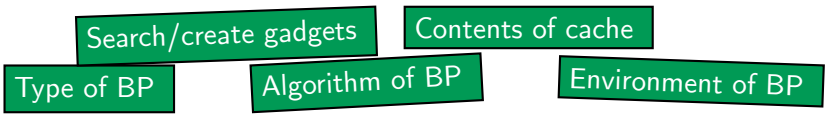
# Abstract example of exploitation

The four components of speculation techniques

1. Speculation primitive
2. Windowing gadget

- Non-cached loads
- Dependency chain of loads
- Dependency chain of integer ALU operations

# Abstract example of exploitation




Tower
*speculative-based attack*

# Abstract example of exploitation

The four components of speculation techniques

1. Speculation primitive
2. Windowing gadget
3. Disclosure gadget

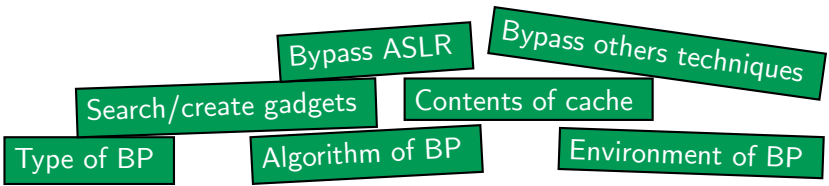
# Abstract example of exploitation

The four components of speculation techniques

1. Speculation primitive
2. Windowing gadget
3. Disclosure gadget

- ASLR
- CFI
- SMAP
- DEP/NX
- retpoline
- and others.

# Abstract example of exploitation




**Babel** tower
*speculative-based attack*

# Abstract example of exploitation

The four components of speculation techniques

1. Speculation primitive
2. Windowing gadget
3. Disclosure gadget
4. Disclosure primitive

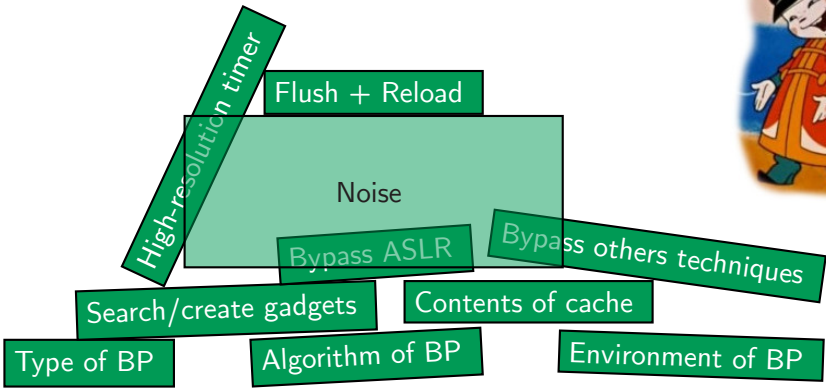
# Abstract example of exploitation

The four components of speculation techniques

1. Speculation primitive
2. Windowing gadget
3. Disclosure gadget
4. Disclosure primitive

- Architecture of cache
- Replacement policies
- Exclusive and inclusive
- Type of cache attack
- Noise
- High-resolution timer
- and etc.

# Abstract example of exploitation




**Babel** tower
*speculative-based attack*

# Agenda

Summary

# Summary

- Software-based microarchitectural attacks become **a very popular**

# Summary

- Software-based microarchitectural attacks become **a very popular**
- Requires **a lot of resources** to develop working exploit

# Summary

- Software-based microarchitectural attacks become **a very popular**
- Requires **a lot of resources** to develop working exploit
- Microarchitectural attacks may be **automated**

# Summary

- Software-based microarchitectural attacks become **a very popular**
- Requires **a lot of resources** to develop working exploit
- Microarchitectural attacks may be **automated**
- Many attacks have **not yet been published**

# Summary

- Software-based microarchitectural attacks become **a very popular**
- Requires **a lot of resources** to develop working exploit
- Microarchitectural attacks may be **automated**
- Many attacks have **not yet been published**
- Countermeasures come with a **performance impact**

# Questions?

# References I



Daniel Gruss
*Software-based Microarchitectural Attacks*.


Moritz Lipp, Daniel Gruss
*ARMageddon: Cache Attacks on Mobile Devices*.


D. Page
*MASCAB: a Micro-Architectural Side-Channel Attack Bibliography*.


Pessl P., Gruss D. and others
*DRAMA: Exploiting DRAM Addressing for Cross-CPU Attacks*.


Bos H., Fratantonio Y. and others
*Drammer: Determenistic Rowhammer Attacks on Mobile Platforms*.


Microsoft
*Mitigating speculative execution side channel hardware vulnerabilities*.

# References II



Google Project Zero
*Reading privileged memory with a side-channel*.


Daniel Gruss, Moritz Lipp
*KASLR is Dead: Long Live KASLR*.


Daniel Gruss, Clémentine Maurice and others
*Flush+Flush: A Fast and Stealthy Cache Attack*.


Fangfei Liu, Yuval Yarom and others
*Last-Level Cache Side-Channel Attacks are Practical*.


Caroline Trippel, Daniel Lustig, Margaret Martonosi
*MeltdownPrime and SpectrePrime: Automatically-Synthesized Attacks Exploiting Invalidation-Based Coherence Protocols*.

# References III



Michael Schwarz, Clémentine Maurice, Daniel Gruss, Stefan Mangard
*Fantastic Timers and Where to Find Them: High-Resolution Microarchitectural Attacks in JavaScript*.


Moritz Lipp, Misiker Tadesse Aga and others
*Nethammer: Inducing Rowhammer Faults through Network Requests*.


Andrei Tatar, Radhesh Krishnan and others
*Throwhammer: Rowhammer Attacks over the Network and Defenses*.


Giovanni Camurati, Sebastian Poeplau and others
*Screaming Channels: When Electromagnetic Side Channels Meet Radio Transceivers*.


Julian Stecklina, Thomas Prescher
*LazyFP: Leaking FPU Register State using Microarchitectural Side-Channels*.


Mordechai Guri, Assaf Kachlon and others
*GSMem: Data Exfiltration from Air-Gapped Computers over GSM Frequencies*.

# References IV


- Dean Sullivan, Orlando Arias, Travis Meade, Yier Jin
  *Microarchitectural Minefields: 4K-Aliasing Covert Channel and Multi-Tenant Detection in IaaS Clouds*.
- B. Gras, K. Razavi, E. Bosman, H. Bos, C. Giuffrida
  *ASLR on the Line: Practical Cache Attacks on the MMU*.
- van Schaik, S. Giuffrida, C. Bos, H. Razavi, K.
  *Malicious Management Unit: Why Stopping Cache Attacks in Software is Harder Than You Think*.
- Daniel Gruss, Anders Fogh and others
  *Prefetch Side-Channel Attacks: Bypassing SMAP and Kernel ASLR*.
- Esmaeil Mohammadian Koruyeh, Khaled N. Khasawneh and others
  *Spectre Returns! Speculation Attacks using the Return Stack Buffer*.

# References V


- Giorgi Maisuradze, Christian Rossow
  *ret2spec: Speculative Execution Using Return Stack Buffers*.
- Guoxing Chen, Sanchuan Chen and others
  *SgxPectre Attacks: Leaking Enclave Secrets via Speculative Execution*.
- Moritz Lipp, Michael Schwarz and others
  *Meltdown*.
- Paul Kocher, Daniel Genkin and others
  *Spectre Attacks: Exploiting Speculative Execution*.
- ARM Whitepaper
  *Cache Speculation Side-channels*.
- Michael Schwarz, Martin Schwarzl, Moritz Lipp, Daniel Gruss
  *NetSpectre: Read Arbitrary Memory over Network*.

# References VI


Sophia D'Antoine
*Out-of-Order Execution and Its Applications*.

Vladimir Kiriansky, Carl Waldspurger
*Speculative Buffer Overflows: Attacks and Defenses*.

Gras, B. Razavi, K. Bos, H. Giuffrida, C.
*Translation Leak-aside Buffer: Defeating Cache Side-channel Protections with TLB Attacks*.

Craig Disselkoen, David Kohlbrenner, Leo Porter, Dean Tullsen
*Prime+Abort: A Timer-Free High-Precision L3 Cache Attack using Intel TSX*.

Moritz Lipp, Michael Schwarz
*Meltdown & Spectre Side-channels considered hARMful*.

Jon Masters
*Exploiting modern microarchitectures: Meltdown, Spectre, and other attacks*.

# References VII



Moritz Lipp
*Cache attacks on ARM.*


Evtyushkin, D. Ponomarev, D. Abu-Ghazaleh, N.
*Jump over ASLR: attacking branch predictors to bypass ASLR.*