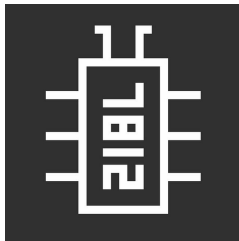


# Введение в атаки по сторонним каналам на микроархитектуру, основанные на исполнении кода

Abc Xyz  
@dura\_lex



DCG#7812  
2018

# План

Введение

Теория

Типы атак

Атаки, основанные на аппаратных дефектах

Meltdown & Spectre

Заключение

# План

## Введение

Атаки по сторонним каналам

Атаки на микроархитектуру

# План

## Введение

Атаки по сторонним каналам

## Атаки по сторонним каналам



Пример цели для атаки по сторонним каналам

# План

## Введение

Атаки на микроархитектуру

# Атаки на микроархитектуру

code1a:

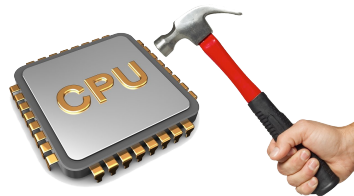
mov (X), %eax

mov (Y), %ebx

clflush (X)

clflush (Y)

jmp code1a



При эксплуатации аппаратных дефектов есть шанс нанести физические повреждения

# План

## Теория

Процессор

Кэш—память

DRAM



# План

## Теория

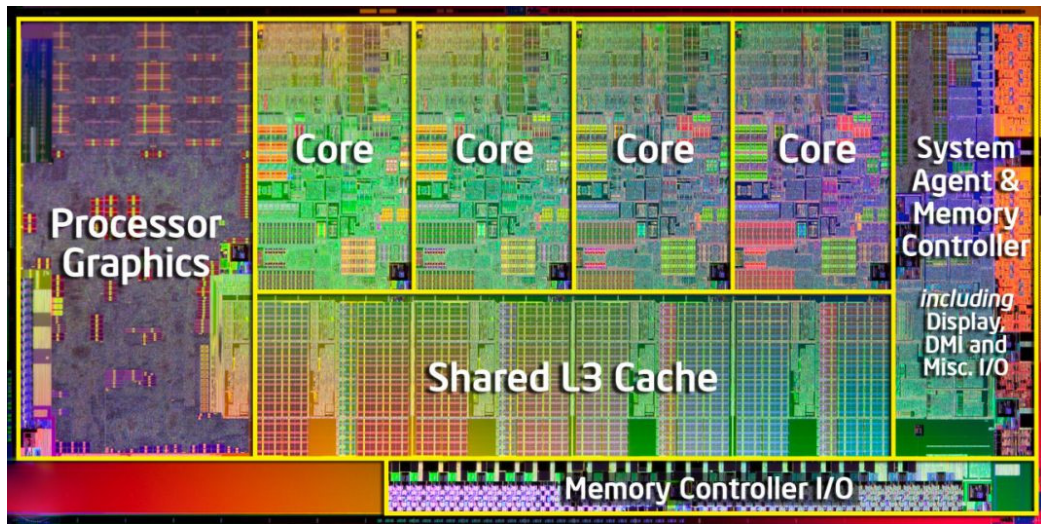
### Процессор

Конвейеризация

Оптимизатор потока инструкций

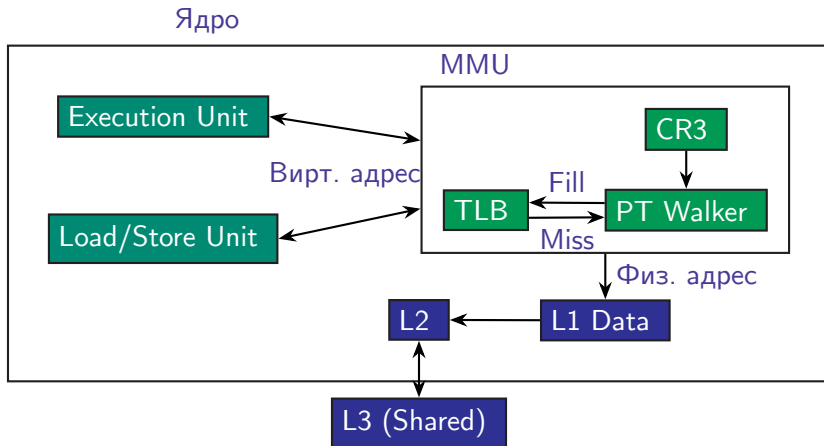
Многоядерность

# Процессор



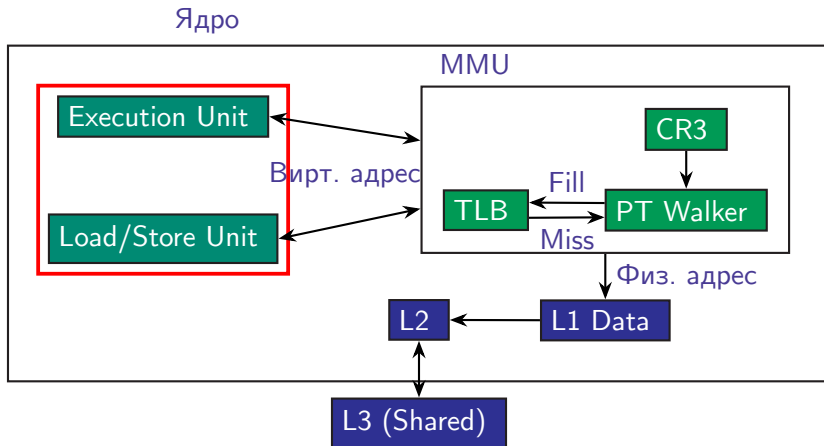
Архитектура многоядерного процессора

# Процессор



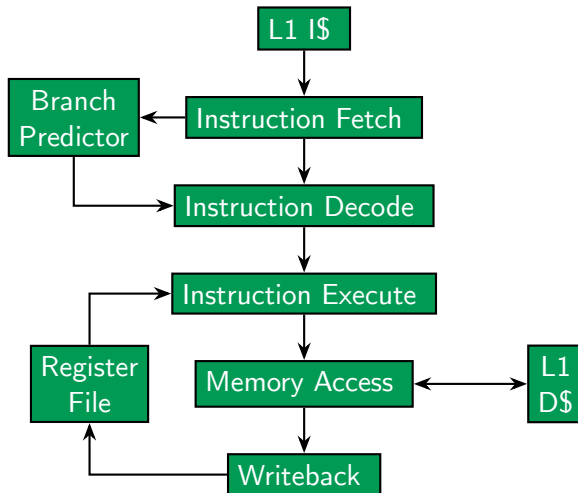
Абстрактная архитектура элементов ядра, работающих с данными

# Процессор



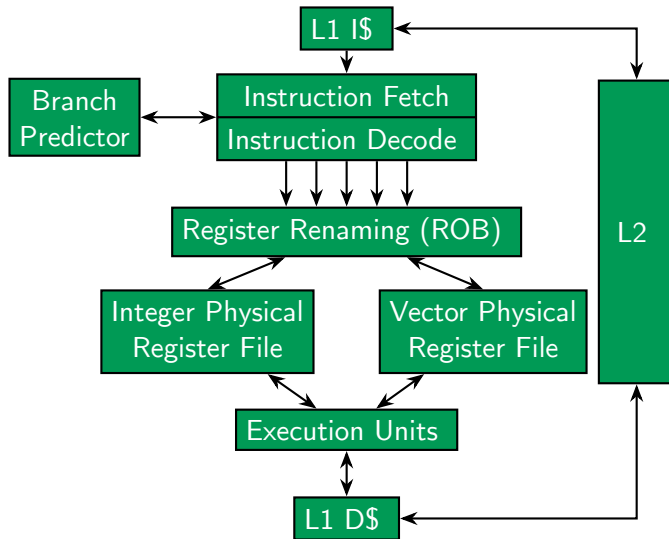
Абстрактная архитектура элементов ядра, работающих с данными

## Конвейеризация. По порядку



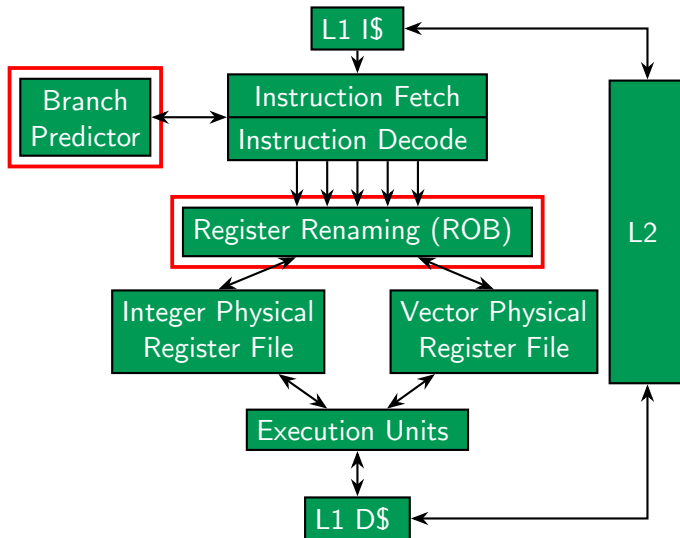
Элементы системы выполнения современного процессора (выполнение по порядку)

## Конвейеризация. Не по порядку



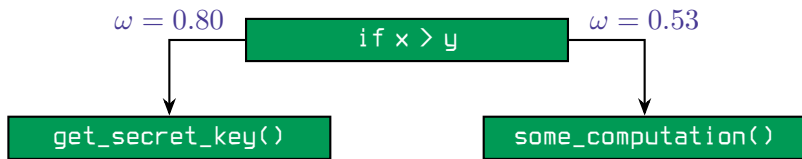
Элементы системы выполнения современного процессора (выполнение инструкций не по порядку)

## Конвейеризация. Не по порядку



Элементы системы выполнения современного процессора (выполнение инструкций не по порядку)

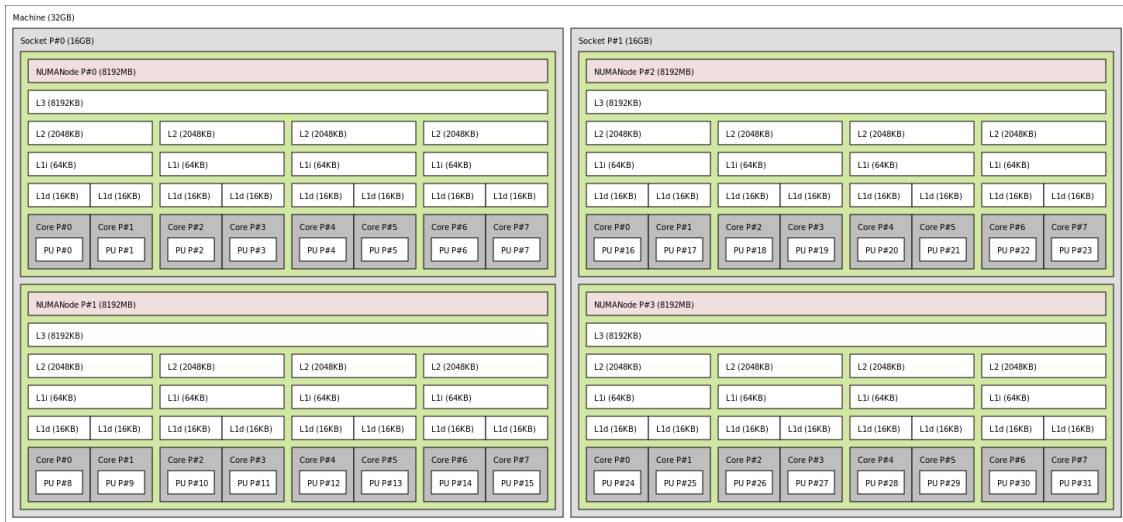
# Оптимизатор потока инструкций



`get_secret_key()` может выполняться спекулятивно



# Многоядерность



Архитектура многоядерного процессора AMD Bulldozer

# План

## Теория

### Кэш-память

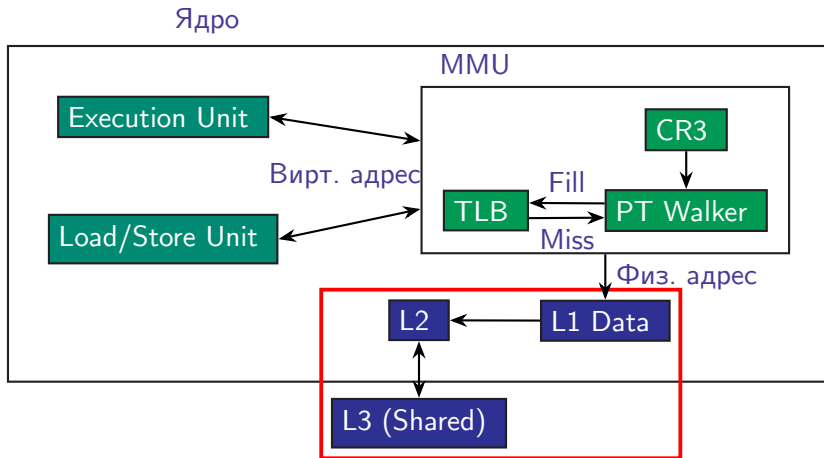
- Типы кэш-памяти

- Наборно-ассоциативный кэш

- Правила вымещения из кэша

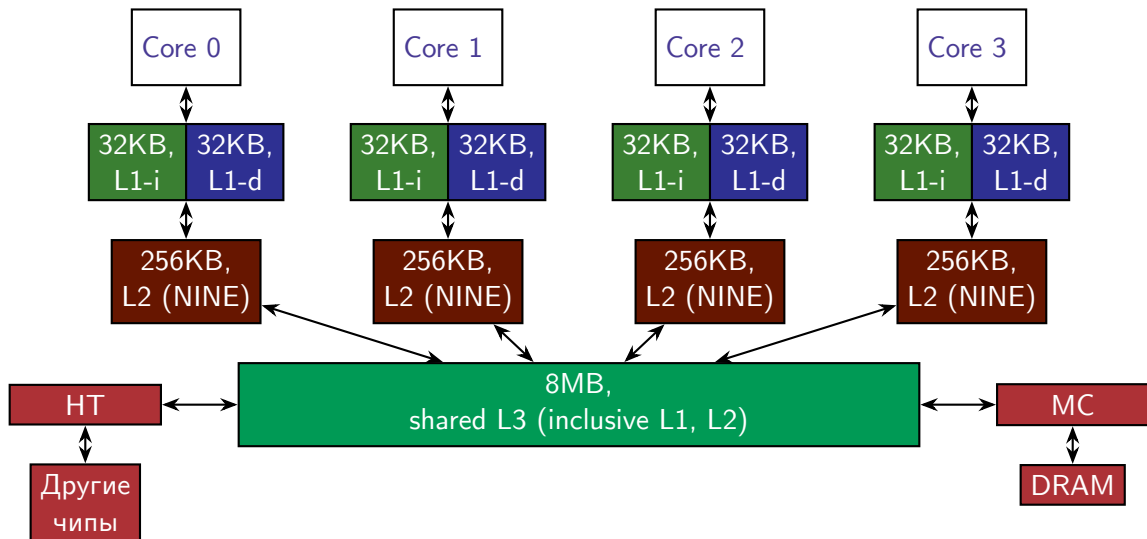
- Режимы адресации

# Кэш-память



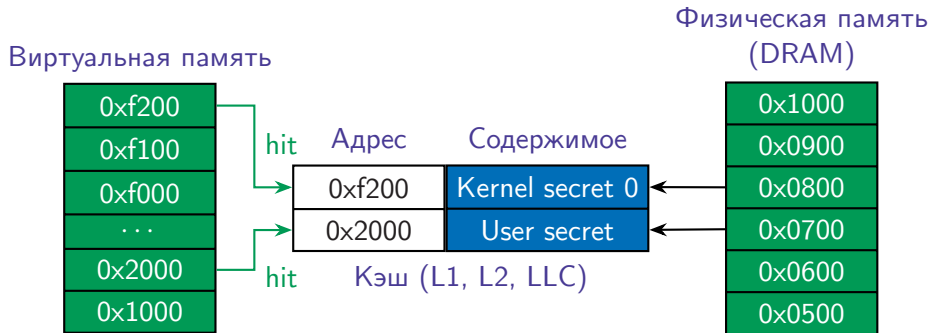
Абстрактная архитектура элементов ядра, работающих с данными

## Кэш-память

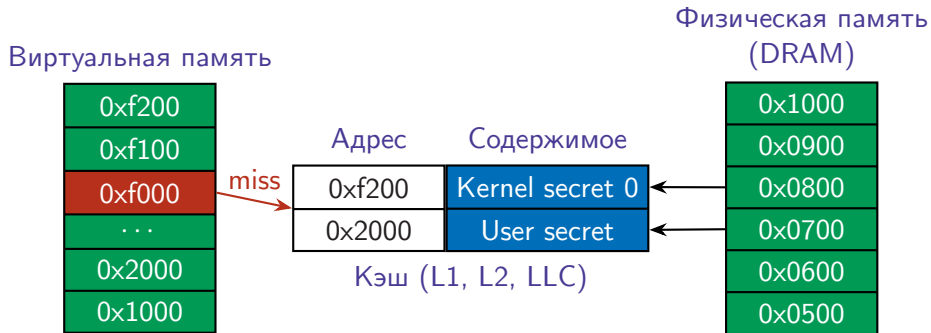


Архитектура процессора относительно кэшей

# Кэш-память

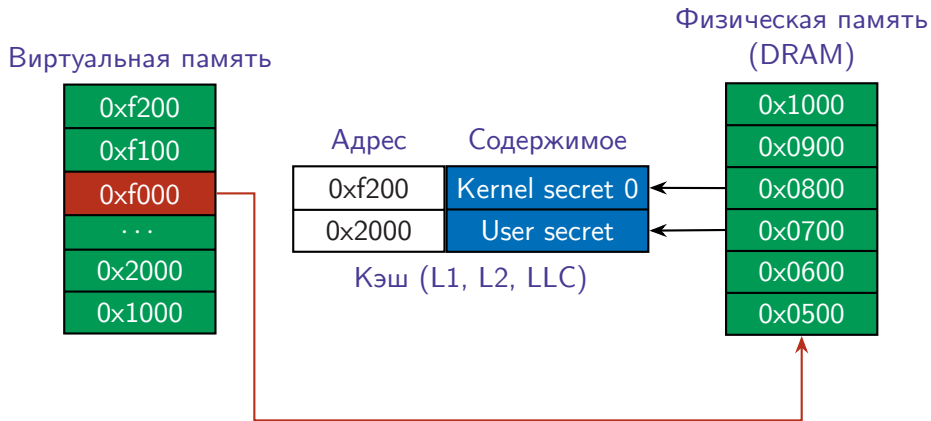


# Кэш-память



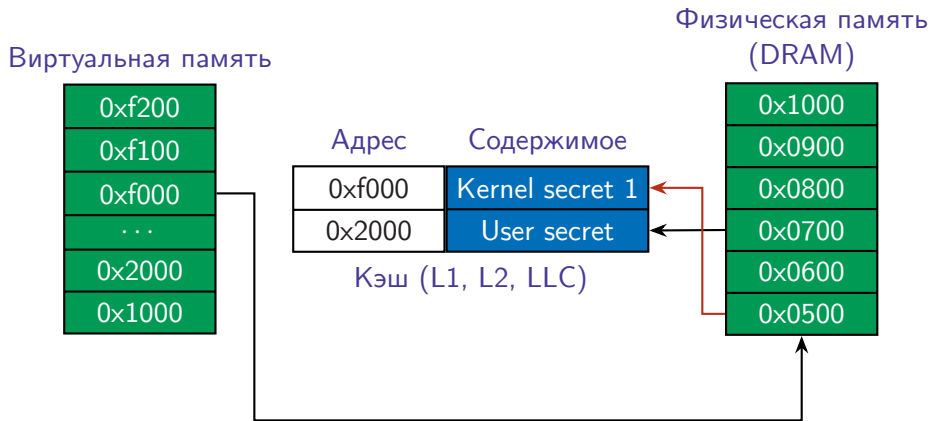
Пример взаимодействия с кэшем

# Кэш-память



Пример взаимодействия с кэшем

# Кэш-память



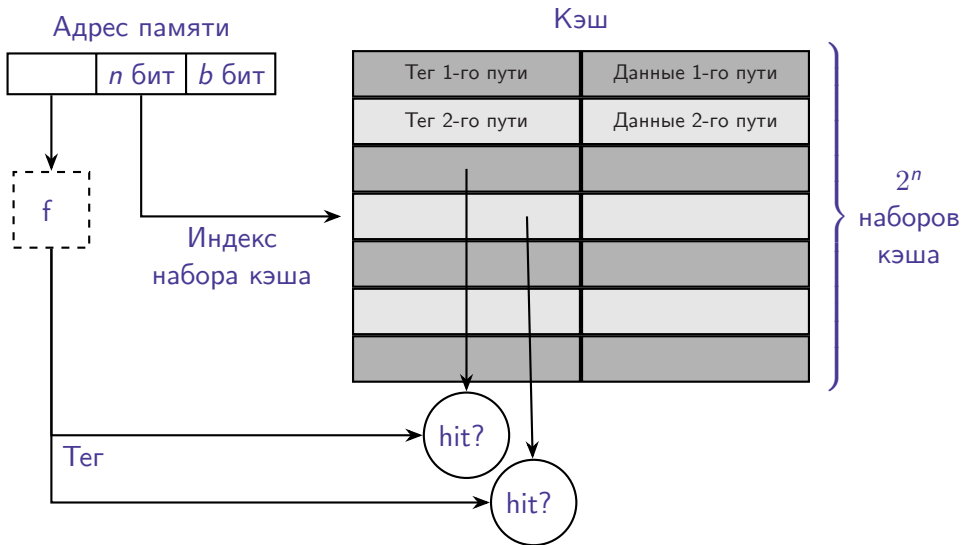
Пример взаимодействия с кэшем



## Типы кэш-памяти

- ▶ кэш с прямым отображением (direct mapped cache)
- ▶ полностью ассоциативный кэш (fully associative cache)
- ▶ наборно-ассоциативный кэш (2/4/8/12-way set associative cache)

# Наборно-ассоциативный кэш



## Правила вымещения из кэша

- ▶ 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
- ▶ и другие.

## Режимы адресации

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

# План

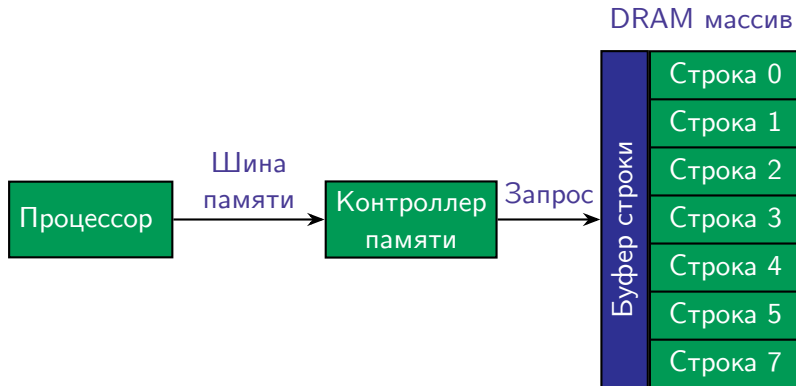
## Теория

### DRAM

Алгоритм работы

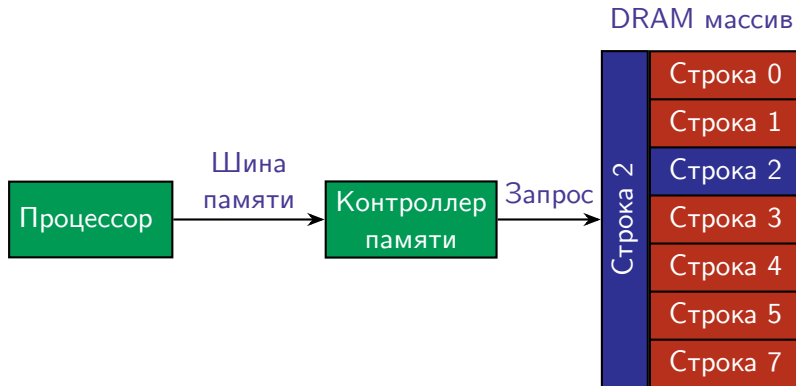
Физическое строение

# Алгоритм работы



Простая компьютерная система с единственным DRAM массивом

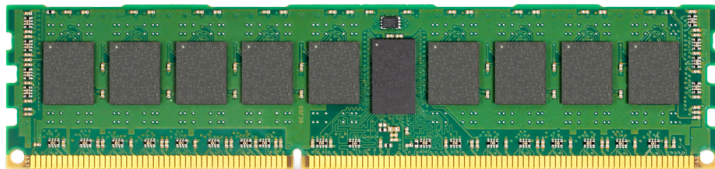
# Алгоритм работы



Простая компьютерная система с единственным DRAM массивом

# Физическое строение

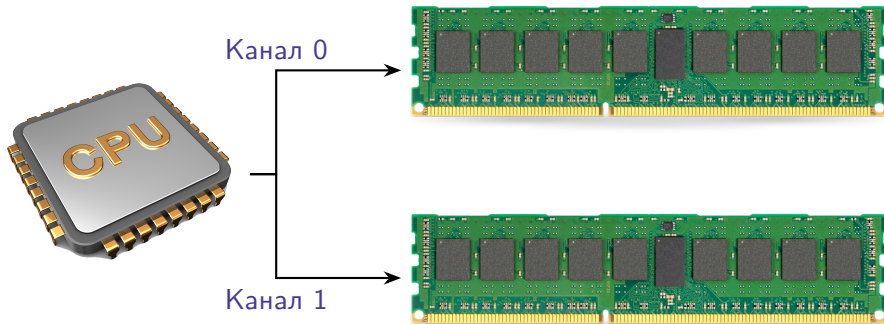
DIMM



Архитектура DRAM

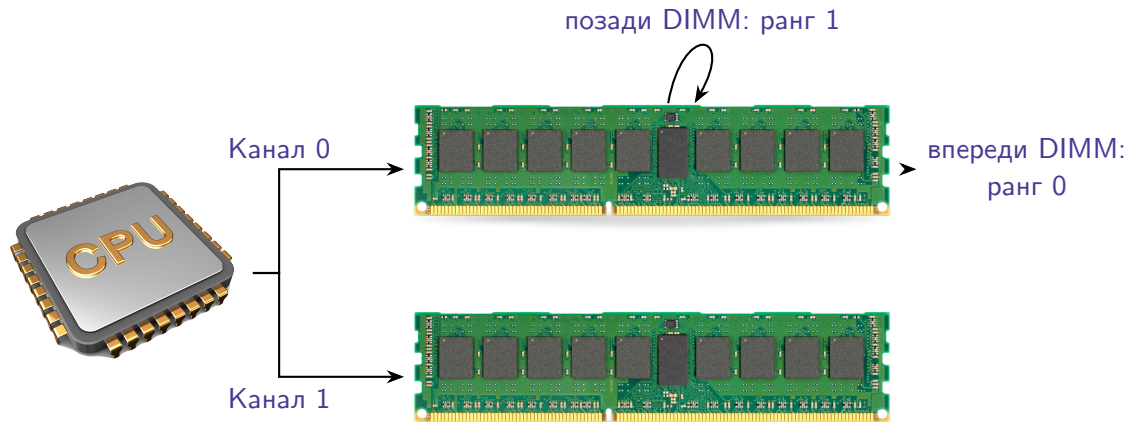


## Физическое строение



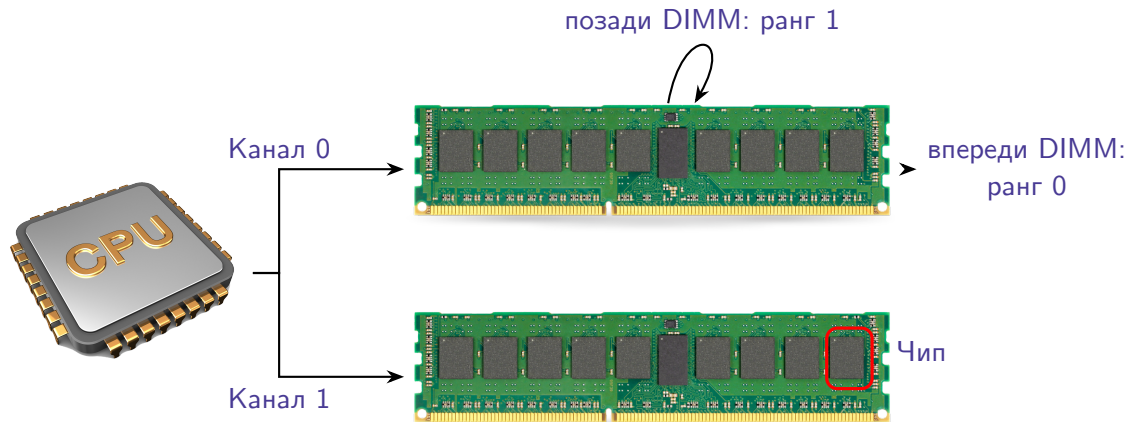
Архитектура DRAM

## Физическое строение



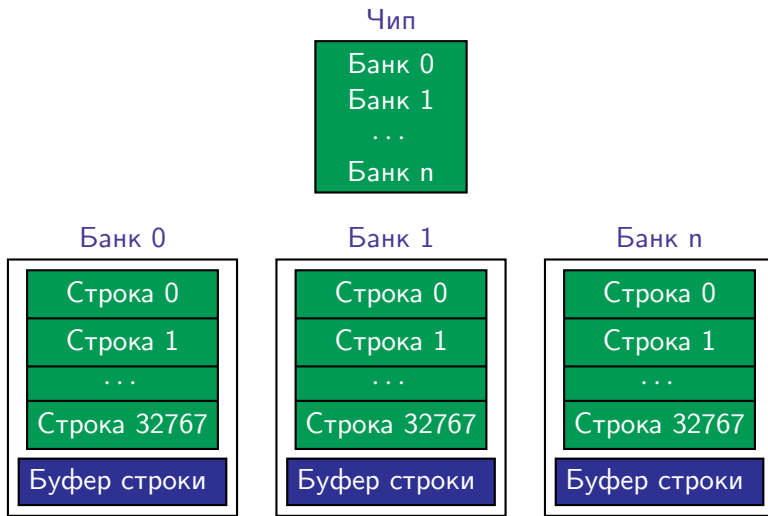
Архитектура DRAM

# Физическое строение



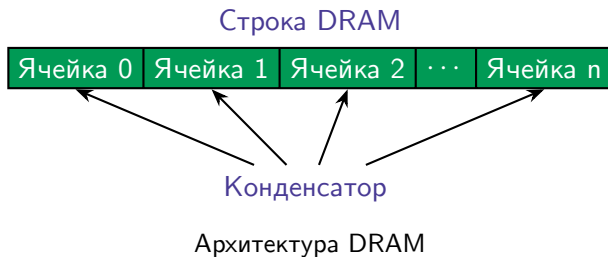
Архитектура DRAM

## Физическое строение



Архитектура DRAM

## Физическое строение



# План

## Типы атак

- Атаки на кэш

- Атаки на предсказатель переходов

- Атаки на буфер ассоциативной трансляции

- Атаки, основанные на срабатывании исключительных ситуаций

- Атаки на DRAM

- Скрытые каналы

# План

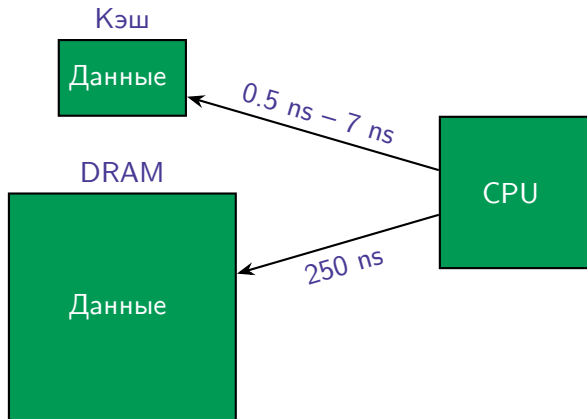
## Типы атак

### Атаки на кэш

Flush + Reload

Другие типы атак на кэш

## Атаки на кэш



Кэш — это не только полезно, но и опасно



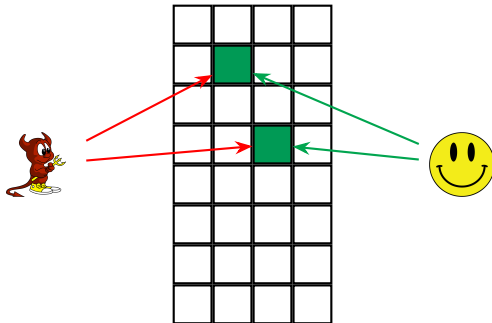
## Flush + Reload

1. Отобразить бинарный файл (например, разделяемый объект) в своё адресное пространство
2. Сбросить содержимое кэш-линии (код или данные)
3. Передать управление программе-жертве
4. Определить какие линии кэша были загружены программой-жертвой снова

## Flush + Reload

Отобразить бинарный файл (например, разделяемый объект) в своё адресное пространство

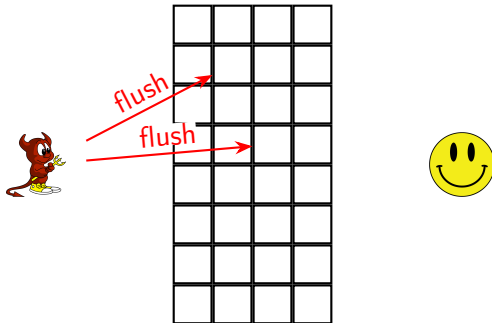
Кэш (8 наборов, 4 пути)



# Flush + Reload

Сбросить содержимое кэш-линии (код или данные)

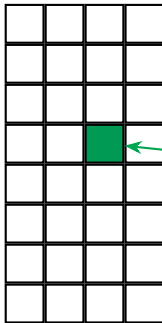
Кэш (8 наборов, 4 пути)



# Flush + Reload

Передать управление программе-жертве

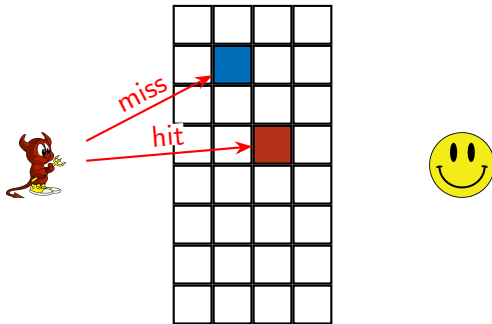
Кэш (8 наборов, 4 пути)



## Flush + Reload

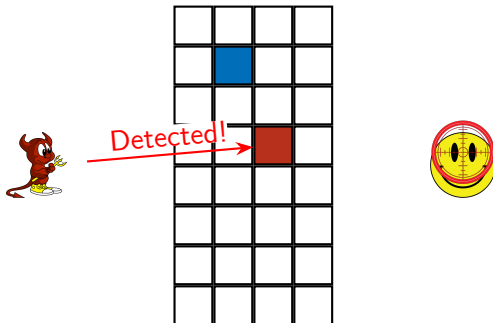
Определить какие линии кэша были загружены программой-жертвой снова

Кэш (8 наборов, 4 пути)



# Flush + Reload

Кэш (8 наборов, 4 пути)



## Другие типы атак на кэш

- ▶ Evict + Time
- ▶ Prime + Probe
- ▶ Prime + Abort
- ▶ Flush + Flush
- ▶ Evict + Reload
- ▶ AnC (ASLR  $\oplus$  Cache)
- ▶ и др.

# План

## Типы атак

Атаки на предсказатель переходов



# Атаки на предсказатель переходов

Виртуальный адрес (user space)

0x0000 EBE45A82

Функция  
индексации

$f(x)$

Branch Target Buffer

Тэг-адрес	Адрес перехода
0xeb45a82	???

0xFFFF EBE45A82

Виртуальный адрес (kernel space)

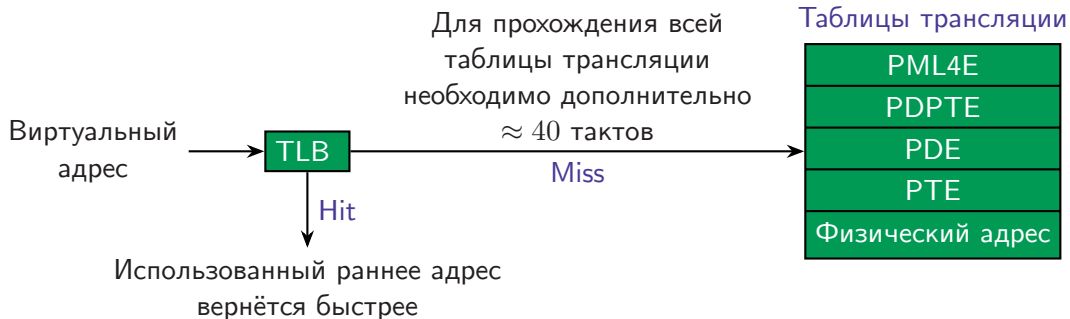
Тег вычисляется, основываясь на последних байтах виртуального адреса

# План

## Типы атак

Атаки на буфер ассоциативной трансляции

# Атаки на буфер ассоциативной трансляции



Translation lookaside buffer (TLB) используется как для ускорения трансляции виртуальных адресов ядерного пространства, так и пользовательского!

# План

## Типы атак

Атаки, основанные на срабатывании исключительных ситуаций

## Атаки, основанные на срабатывании исключительных ситуаций

- ▶ прерывание планировщика
- ▶ инструкции прерывания
- ▶ ошибка отсутствия страницы в памяти
- ▶ поведенческие изменения (например, возврат кода ошибки)

# План

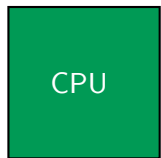
## Типы атак

### Атаки на DRAM

Алгоритм работы DRAM

Типы атак на DRAM

# Алгоритм работы DRAM



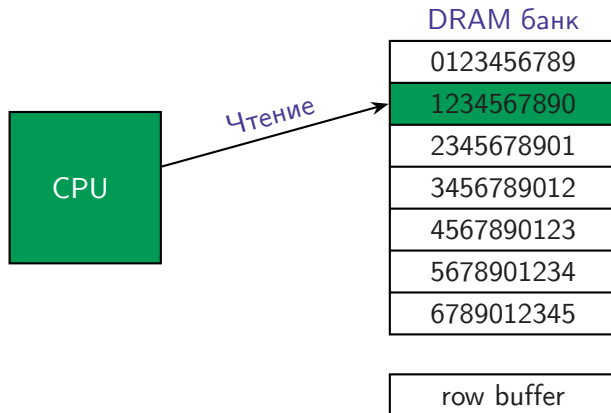
DRAM банк

0123456789
1234567890
2345678901
3456789012
4567890123
5678901234
6789012345

row buffer
------------

Работа DRAM (ещё раз)

# Алгоритм работы DRAM



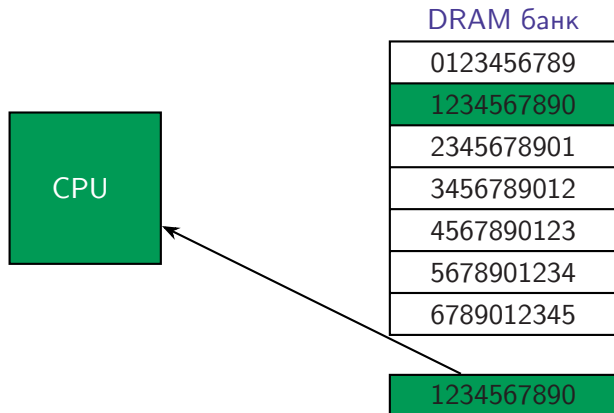
CPU запрашивает на чтение строку №1



# Алгоритм работы DRAM

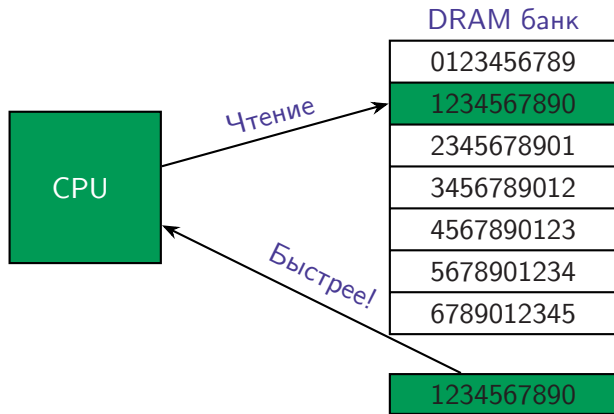


# Алгоритм работы DRAM



CPU читает строку №1 из буфера строки

## Алгоритм работы DRAM



CPU снова запрашивает на чтение строку №1, которая уже есть в буфере строки, чтение происходит быстрее

# Типы атак на DRAM

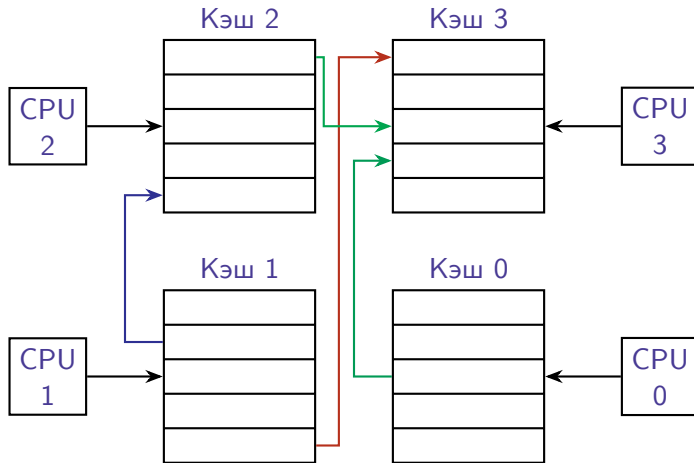
- ▶ DRAMA
- ▶ Row hit (Flush + Reload)
- ▶ Row miss (Prime + Probe)
- ▶ и др.

# План

Типы атак

Скрытые каналы

# Скрытые каналы



Скрытые каналы между процессорами

# Скрытые каналы

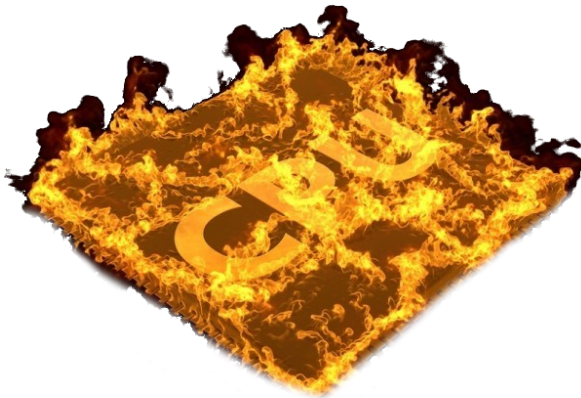
- ▶ Атаки на кэш (использование разделяемой библиотеки)
- ▶ Row miss атака (DRAM)
- ▶ Тепловой канал
- ▶ Радио канал (без специализированного аппаратного обеспечения)

# План

Атаки, основанные на аппаратных дефектах  
Rowhammer



## Атаки, основанные на аппаратных дефектах



Аппаратные дефекты можно эксплуатировать с помощью исполнения кода

# План

## Атаки, основанные на аппаратных дефектах

### Rowhammer

- Необходимые примитивы Rowhammer

- Разновидности Rowhammer

## Необходимые примитивы Rowhammer

- ▶ быстрый некэшируемый доступ к памяти
- ▶ определение местонахождения уязвимых строк DRAM
- ▶ знание функций адресации физической памяти

## Разновидности Rowhammer

- ▶ Flip Feng Shui — целенаправленный Rowhammer
- ▶ Throwhammer — удалённая атака
- ▶ Nethammer — улучшенная удалённая атака
- ▶ Drammer, RAMpage — атака на ARM
- ▶ Glitch — улучшенная атака на ARM

# План

## Meltdown & Spectre

Производные и не только

Абстрактный пример эксплуатации

# План

## Meltdown & Spectre

Производные и не только

# Производные и не только

## Spectre-NG

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

# Производные и не только

- ▶ Spectre 1.1, 1.2 (Speculative Buffer Overflows)
- ▶ SpectreRSB
- ▶ NetSpectre
- ▶ ...

TotalMeltdown?



# План

## Meltdown & Spectre

Абстрактный пример эксплуатации

# Абстрактный пример эксплуатации

Требуется для атаки с помощью техники спекулятивного выполнения:

1. Примитив спекулятивного выполнения

# Абстрактный пример эксплуатации

Требуется для атаки с помощью техники спекулятивного выполнения:

## 1. Примитив спекулятивного выполнения

- ▶ Обход проверки границ
- ▶ Тренировка предсказателя переходов
- ▶ Чтение памяти после сохранения её в регистр
- ▶ Отложенная исключительная ситуация
- ▶ Засорение таблиц с историей шаблонов переходов
- ▶ Засорение Return Stack Buffer
- ▶ Спекулятивная запись (buffer overflow)

Микроархитектура — ?

# Абстрактный пример эксплуатации



Фундамент башни  
*атаки на основе спекулятивного выполнения*

## Абстрактный пример эксплуатации

Требуется для атаки с помощью техники спекулятивного выполнения:

1. Примитив спекулятивного выполнения
2. Гаджеты для создания «окна» спекулятивного выполнения

# Абстрактный пример эксплуатации

Требуется для атаки с помощью техники спекулятивного выполнения:

1. Примитив спекулятивного выполнения
2. Гаджеты для создания «окна» спекулятивного выполнения

- ▶ Загрузка некешированных данных
- ▶ Цепочка из зависимых загрузок данных
- ▶ Цепочка из зависимых целочисленных операций в АЛУ

## Абстрактный пример эксплуатации



Башня

*атаки на основе спекулятивного выполнения*

# Абстрактный пример эксплуатации

Требуется для атаки с помощью техники спекулятивного выполнения:

1. Примитив спекулятивного выполнения
2. Гаджеты для создания «окна» спекулятивного выполнения
3. Гаджеты обнародования информации



# Абстрактный пример эксплуатации

Требуется для атаки с помощью техники спекулятивного выполнения:

1. Примитив спекулятивного выполнения
2. Гаджеты для создания «окна» спекулятивного выполнения
3. Гаджеты обнародования информации

- ▶ ASLR
- ▶ CFI
- ▶ SMAP
- ▶ DEP/NX
- ▶ retpoline
- ▶ И т. д.

## Абстрактный пример эксплуатации



**Вавилонская башня**

*атаки на основе спекулятивного выполнения*

## Абстрактный пример эксплуатации

Требуется для атаки с помощью техники спекулятивного выполнения:

1. Примитив спекулятивного выполнения
2. Гаджеты для создания «окна» спекулятивного выполнения
3. Гаджеты обнародования информации
4. Примитив обнародования информации

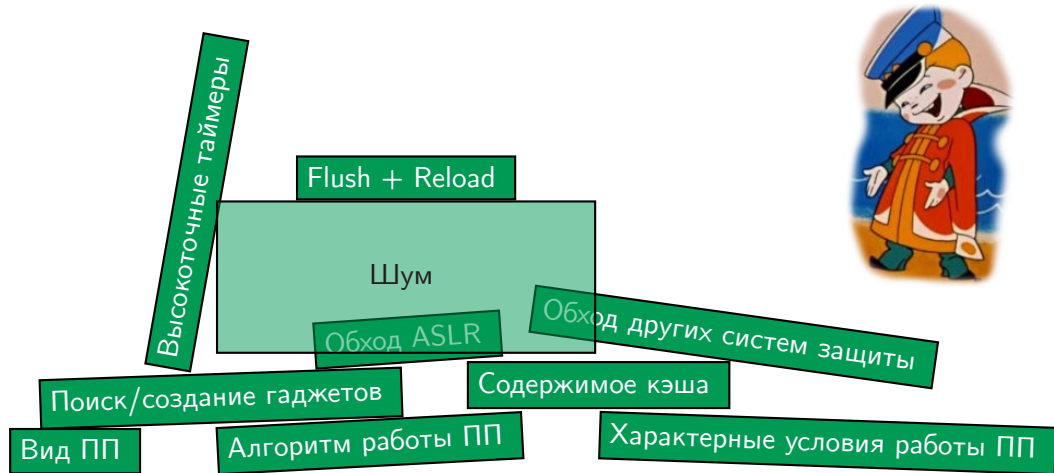
## Абстрактный пример эксплуатации

Требуется для атаки с помощью техники спекулятивного выполнения:

1. Примитив спекулятивного выполнения
2. Гаджеты для создания «окна» спекулятивного выполнения
3. Гаджеты обнародования информации
4. **Примитив обнародования информации**

- ▶ Устройство кэша
- ▶ Правила вымещения из кэша
- ▶ Эксклюзивность и инклюзивность
- ▶ Тип атаки
- ▶ Зашумлённость
- ▶ Счётчики
- ▶ И т. д.

# Абстрактный пример эксплуатации



**Вавилонская башня**

*атаки на основе спекулятивного выполнения*

План

Заключение

## Заключение

- ▶ атаки на микроархитектуру становятся **популярными**

## Заключение

- ▶ атаки на микроархитектуру становятся **популярными**
- ▶ требуется **много ресурсов** для разработки эксплоита



## Заключение

- ▶ атаки на микроархитектуру становятся **популярными**
- ▶ требуется **много ресурсов** для разработки эксплоита
- ▶ атаки на микроархитектуру могут быть **автоматизированы**

## Заключение



- ▶ атаки на микроархитектуру становятся **популярными**
- ▶ требуется **много ресурсов** для разработки эксплоита
- ▶ атаки на микроархитектуру могут быть **автоматизированы**
- ▶ множество атак ещё **не опубликовано/найдено**

## Заключение

- ▶ атаки на микроархитектуру становятся **популярными**
- ▶ требуется **много ресурсов** для разработки эксплоита
- ▶ атаки на микроархитектуру могут быть **автоматизированы**
- ▶ множество атак ещё **не опубликовано/найдено**
- ▶ создание контрмер — **не тривиальный процесс**

Вопросы?

# Источники 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.*

## Источники 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.*

## Источники 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.*

## Источники 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.*



## Источники 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.*

## Источники 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.*