





**"IT'S A 1"**



110011010

010111010

111100100

000101101

100110010

**Collector**

110011010

010111010

111100100

000101101

100110010

**Spy**



MELTDOWN: THE ATTACK



3

9



1. **spy** will read the **secret**

2. Depending on the **value**, **Sp****y** will cache a grey block<sup>1</sup>

3. CPU detects **Spys** access validation and terminates **Spys**

4. **Collector** now reads all grey blocks and stops the time

1. Block "It's a 3" will be the block read the fastest















"IT'S A 2"

RAM

"IT'S A 3"



"IT'S A 1"

**SECRET ("3")**



Gamechangers



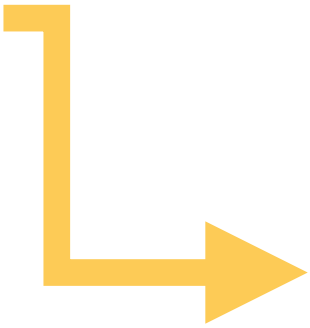


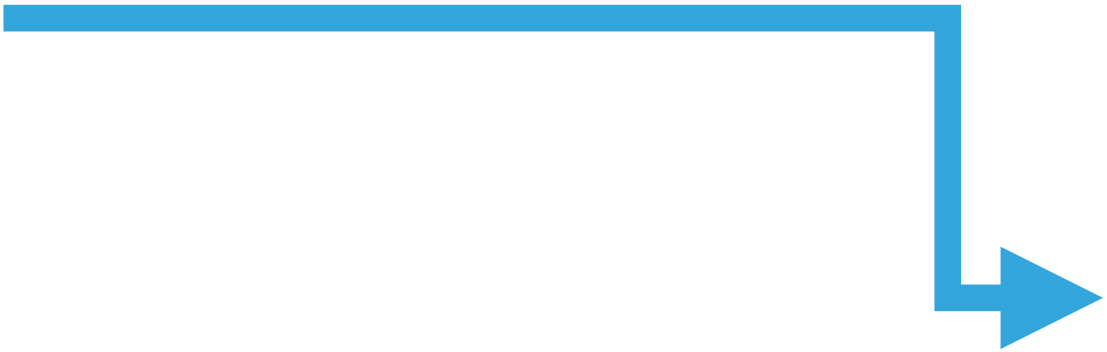
**SECRET ("3")**





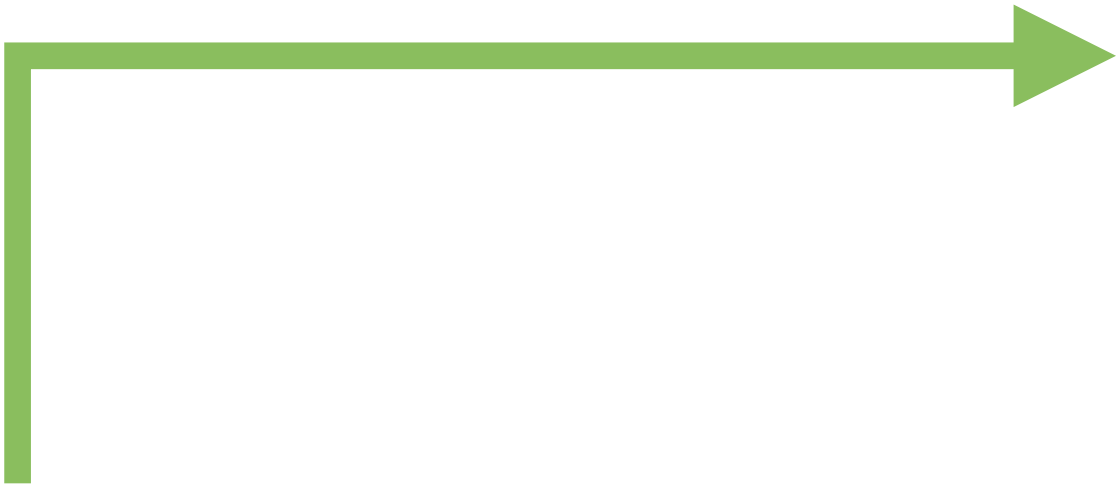


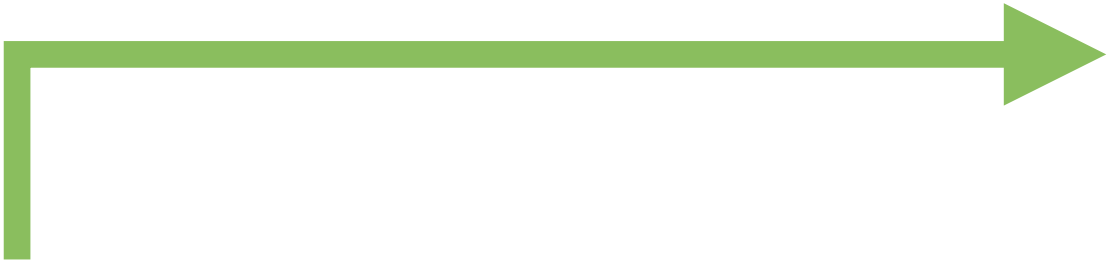




“IT’S A 3”







**"IT'S A 2"**





read: 103ns (uncached read)

read: 103ns (uncached read)

read: 3ns (cached)



2



1



<sup>1</sup>Actually Spyy will cache the address of block #3 and Collect will read the blocks addresses



# MELTDOWN: THE ATTACK

110011010  
010111010  
111100100  
000101101  
100110010

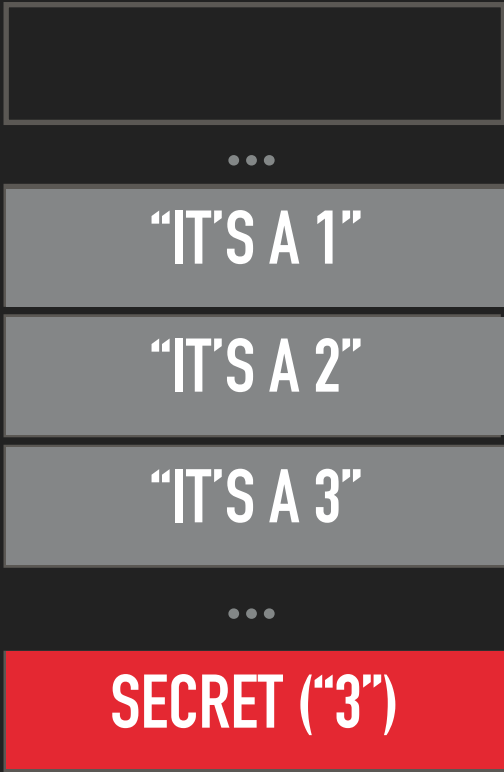
Spy

110011010  
010111010  
111100100  
000101101  
100110010

Collector



Cache



RAM

read: 103ns (uncached read)

read: 103ns (uncached read)

read: 3ns (cached)

- 2 1. **Spy** will read the **secret**
- 2 2. Depending on the **value**, **Spy** will cache a grey block<sup>1</sup>
- 1 3. CPU detects **Spys** access validation and terminates **Spy**
- 4. **Collector** now reads all grey blocks and stops the time
- 1 1. Block "It's a 3" will be the block read the fastest

<sup>1</sup> Actually Spy will cache the *address* of block #3 and Collector will read the blocks *addresses*



## MELTDOWN

Meltdown exploits two properties of modern CPUs

- ▶ *Out of order execution* of OPs and  $\mu$ OPs
- ▶ Timing side channels for the cache

This allows an attacker to

- ▶ Read all memory mapped<sup>1</sup> in a process
- ▶ This often includes all other processes memory
- ▶ This does NOT allow reading “outside of a VM<sup>2</sup>”

<sup>1</sup> [Virtual vs. physical memory](#) is a subject for another time   <sup>2</sup> For fully virtualised VMs