Week 6 Report

Ben Chen

Dept of Computer Science and Engineering, SUSTech

October 23, 2024

TOC

Title	Conference	Institute	Authors	Idea
A Genetic Algorithm for a Spectre Attack Agnostic to Branch Predictors	CARRV '23	Telecom Paris	Dorian Bourgeoisat Laurent Sauvage	Branch predictor independent spectre attack.
BUSted!!! Microarchitectural Side-Channel Attacks on the MCU Bus Interconnect	Oakland '24	UMinho	Cristiano Rodrigues Daniel Oliveira Sandro Pinto	Contention on bus between MCU and DMA. Similar to interrupt

Genetic Algorithm for Spectre[1]

```
void gadget(int x) {
    if (x < array1_size)
        y = array2[array1[x] * CACHE_LINE_SIZE];
}
...
// attack
for (int i = 0; i < N_TRAIN; i++)
    gadget(0);
gadget(&secret - array1); // not working
```

Reason: BP changed from Gshare $\rightarrow$ TAGE-**L**.

Loop predictor kicks in, so no misprediction.

Genetic Algorithm for Spectre[1]

Different implementations of branch predictors in RISC-V: needs a generic attack.

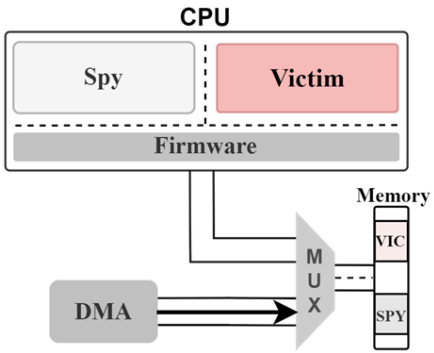
Intuition: **Evolutionary Algorithm** to find the training sequence.

Parameters	**x[0]**	**x[1]**	**x[2]**	**x[3]**	**x[4]**	**...**
Value	0	4	3	1	ATTACK	**...**

using the code

```
for (int i = 0; i < N_TRAIN; i++)
    gadget(x[i]);
```

Attack Overview – Toy Example



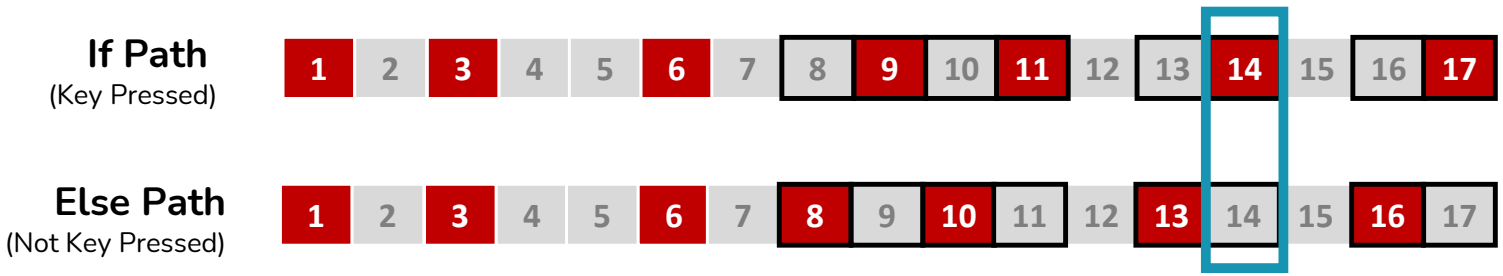

```
        cmp   r3, #0          1 clk
        beq.n ELSE            3 clk (else), 1 clk (if)
IF:     movs  r3, #1          1 clk
        str   r3, [r7, #0]    1 clk
        b.n   END             2 clk
ELSE:   movs  r3, #0          1 clk
        str   r3, [r7, #0]    1 clk
END:    nop                   1 clk
```

```
if(s==1)
  var=1;
else
  var=0;
```

Clock	t	t+1	t+2	t+3	t+4	t+5	t+6
If	cmp	beq	movs	str	b	b	nop
Trace	---	---	---	X	---	---	---
Else	cmp	beq	beq	beq	movs	str	nop
Trace	---	---	---	---	---	X	---

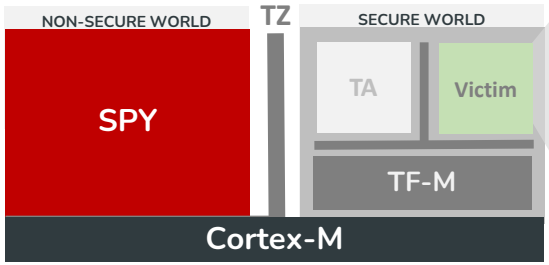
SECRET = 1

BUSted Profiling Phase



Let's Pick Contention Point 14

BUSted Attack




```
signed int read_keypad(void){
    int is_pressed, mask = 0x1;
    int new_key_state = get_keypad_state();                    read key
    for (int key = 0; key < KYPD_NB_KEYS; key++){
        is_pressed = (new_key_state & mask) & ~(key_state & mask);
        if (is_pressed)
          pin[pin_idx++] = key;                                if branch (leak)
        else
          dummy_pin[dummy_pin_idx++] = key;                    else branch
        dummy_pin_idx = 0;
        mask <<= 1;                                                          for loop
    }
    key_state = new_key_state;
    return (4 - pin_idx);
}
void read_pin(){
  signed int pin_len = PIN_LEN;
  while(pin_len>0)
    pin_len = read_keypad();                                   while loop
}
```

Code based in Sancus and Texas Reference Implementation of a Keypad [1,2]

[1] https://github.com/sancus-tee/vulcan/blob/master/demo/ecu-tcs/sm_tcs_kypd.c
[2] Implementing An Ultra-Low-Power Keypad Interface With MSP430™ MCUs

References I

[1] Dorian Bourgeoisat and Laurent Sauvage. "A Genetic Algorithm for a Spectre Attack Agnostic to Branch Predictors". In: June 2023. URL: https://telecom-paris.hal.science/hal-04210397.

[2] C. Rodrigues, D. Oliveira, and S. Pinto. "BUSted!!! Microarchitectural Side-Channel Attacks on the MCU Bus Interconnect". In: *2024 IEEE Symposium on Security and Privacy (SP)*. 2024.