Conference Paper Title

Zachary Driskill and Jacob Brown
Brigham Young University
450 Engineering Building
Provo, UT 84602
{zd227, jacobb00}@byu.edu

***Abstract*—We did a timing attack.**

***Index Terms*—timing attack, FPGA**

I. Introduction

Timing attacks are a type of side-channel attack.

To further learn about timing attacks, explored a timing attack that exploit a non-constant compare function to extract a secret key. Our source of inspiration is Joe Grand, who demonstrated this simple timing attack using a microcontroller, four buttons, and an oscilloscope [1]. The non-constant compare function is simply a compare function that exits as soon as a mismatch occurs (see the code snippet below).

```
def non_constant_compare(guess):
    for x, y in zip(key, guess):
        if x != y:
            return False
    return True
```

By measuring how long the compare function takes to execute, one can estimate how much of the input matched. For example, say we are guessing a 4 character string. We guess 26 strings, each simply a single letter repeated four times ("aaaa", "bbbb", "cccc" ... "zzzz"), and measure the execution time of the compare function. The guessed string with the lowest compare time has the correct first digit. This process is repeated for the other three digits, making sure to put the discovered digits in front of the guessed digits.

We implement two simple timing attacks on the HaHav3 board that exploit a non-constant compare function:

1) **MCU to FPGA** - the MCU is the victim and the FPGA is the attacker. The FPGA counts cycles to measure compare time.
2) **Human to FPGA** - the FPGA is the victim and a human is the attacker. The human uses an oscilloscope to measure compare time.

II. FPGA to MCU

TODO

III. Human to FPGA

For the human to FPGA timing attack, the non-constant compare function is implemented on the FPGA present on the HaHav3 board and a human carries out the attack. The FPGA contains a secret key that it compares to the input guess. A human uses the three buttons on the HaHav3 board to input a four pin guess and uses an oscilloscope to measure the time between the last button press and the compare function result. It is important to note that the FPGA waits until all four input values are received before beginning the compare function, so the final button press can act as a time reference for the start of the comparison function.

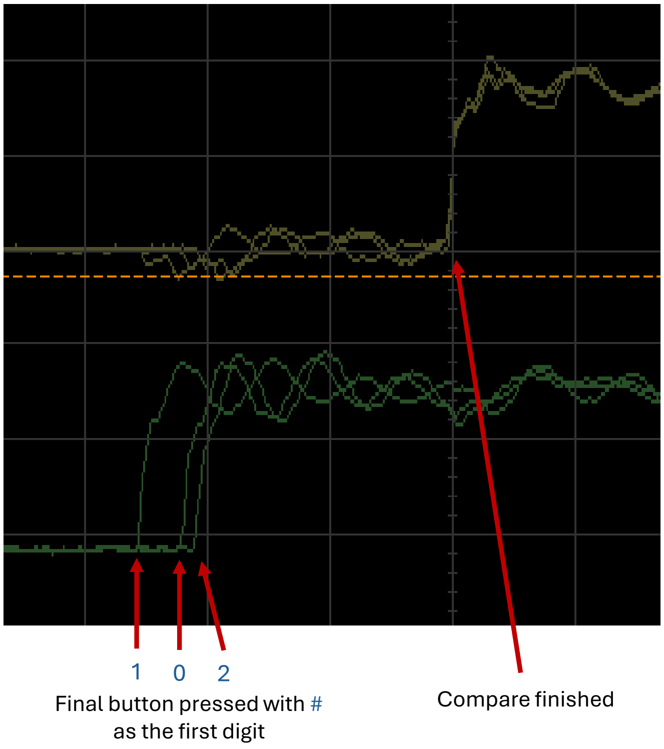

Fig. 1. Oscilloscope waveform capture showing when the final button was pressed for each sequence and when the comparison finished. The 'compare finished' point is centered at 0.

IV. Conclusion

Acknowledgment

Dr. Goeders taught us everything we know about hardware security. Joe Grand gave us the idea for the simple timing attack.

References

[1] Joe Grand. Side channel timing attack demonstration, 2017.

TABLE I
INPUT VALUES AND COMPARISON DELAYS FOR EACH TESTED DIGIT.

Digit	Input Value	Delay (ns)
0	0000	109
	1111	128
	2222	104
1	1000	160
	1111	131
	1222	127
2	1000	153
	1011	153
	1022	173
3	1020	165
	1021	174
	1022	172