

# Assignment 2: Side-Channel Analysis and Fault Injection

Amar Lakshya (AXX945), Calvin Menezes (CXM1010)

*School of Computer Science, University of Birmingham*

---

---

## 1. Task1

The average number of cycles per execution of the exponentiation for  $e = 17$  is:

## 2. Task2

### 2.1. Oscilloscope Capture

The image of the RSA operations can be seen in Fig 1

### 2.2. Manual Extraction

TODO

### 2.3. Acquired Trace

The graph of the acquired trace can be found in Fig 2 and **python** program file *load\_trace* contains the program.

### 2.4. SPA

TODO

## 3. Task3

### 3.1. Overlapping Traces

The plot of the first 10 traces can be seen in Fig 3.

The corresponding output can be seen below:

```
byte value for trace 0: D8
byte value for trace 1: E8
byte value for trace 2: 42
byte value for trace 3: 3D
byte value for trace 4: 96
byte value for trace 5: E5
byte value for trace 6: CB
byte value for trace 7: 0B
byte value for trace 8: 1F
byte value for trace 9: 33
```

### 3.2. AES XOR

The following function implements the required functionality (the complete program can be found in the aforementioned source code repository and submitted zip):

```
def AES_XOR(x, k_prime):
    return SBOX[x ^ k_prime]
```

### 3.3. Significant Bit

The method to extract the *MSB* involves two steps where we first convert an *integer* to a *bit-field* and then we get the first element of the *bit-field*. The following is an implementation of the function in **python**:

```
def int_to_bits(n):
    return [n >> i & 1 for i in range(BIT_SIZE-1, -1, -1)]

def MSB(n):
    return int_to_bits(n)[0]
```

### 3.4. DPA

The submitted program file *dpa.py* contains the program to produce the overlapping graphs of all difference-of-means curves as seen in Fig 4 and the zoomed-in part showing the candidate key  $K10 = 7$  as seen in Fig 5.

## 4. Task4

### 4.1. Fault Injection on RSA-CRT

From the given values  $n = 91, e = 5, S = 48, \bar{S} = 35, x = 55$  we calculate the valid signature  $y$  by:

$$y = x^e \pmod{n} \quad (1)$$

$$y = 55^5 \pmod{91} \quad (2)$$

$$y = 48 \quad (3)$$

#### 4.1.1. Using Bellcore Method

The bellcore method gives factor  $q$  by:

$$q = gcd(S - \bar{S}, n) \quad (4)$$

$$q = gcd(48 - 35, 91) \quad (5)$$

$$q = gcd(13, 91) \quad (6)$$

$$q = 13 \quad (7)$$

we know  $n = p * q$  therefore factor  $p$  is:

$$p = n/q \quad (8)$$

$$p = 91/13 \quad (9)$$

$$p = 7 \quad (10)$$

#### 4.1.2. Using Lenstra Method

The Lenstra method gives factor  $q$  by:

$$q = \gcd(\bar{Y}^e - x, n) \quad (11)$$

$$(12)$$

From eq 1, we know that  $y = 48$  and since  $\bar{y} = y, \bar{y} = 48$ , therefore

$$q = \gcd(48^5 - 55, 91) \quad (13)$$

$$q = \gcd(48 - 55, 91) \quad (14)$$

$$q = \gcd(7, 91) \quad (15)$$

$$q = 7 \quad (16)$$

we know  $n = p * q$  therefore factor  $p$  is:

$$p = n/q \quad (17)$$

$$p = 91/7 \quad (18)$$

$$p = 13 \quad (19)$$

The major advantage of Lenstra method is that only the faulty value  $\bar{S}$  needs to be known.

#### 4.2. Counter Measures

**TODO**

## 5. Appendices



Figure 1: The picture of oscilloscope showing the first 6 operations of RSA

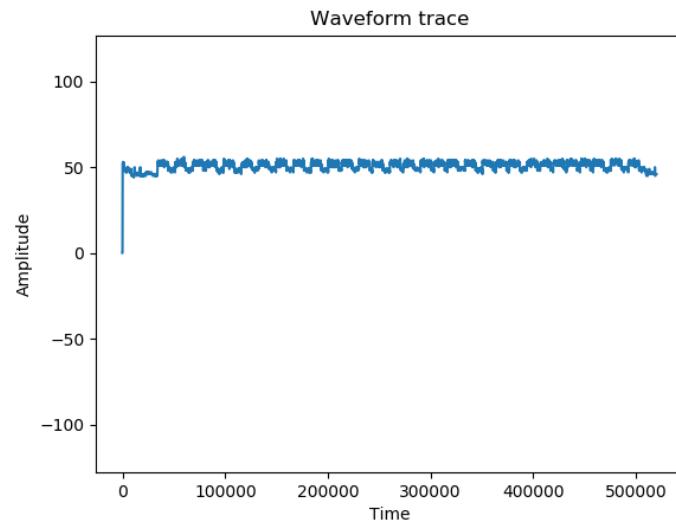


Figure 2: Graph of the acquire trace in full scale

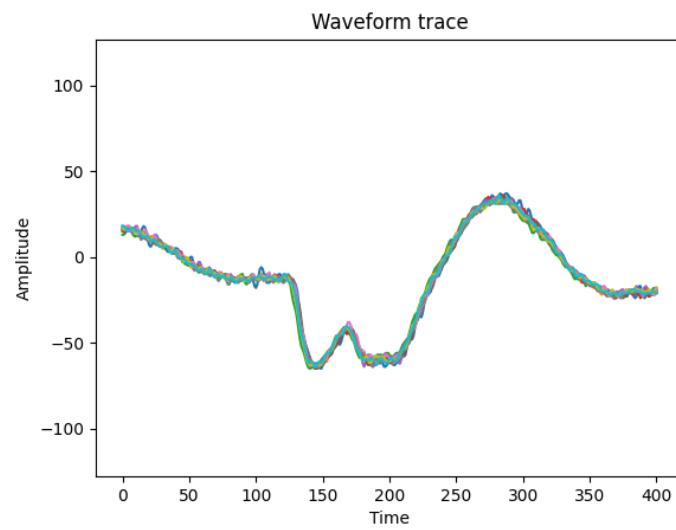


Figure 3: Graph of superimposed 10 traces

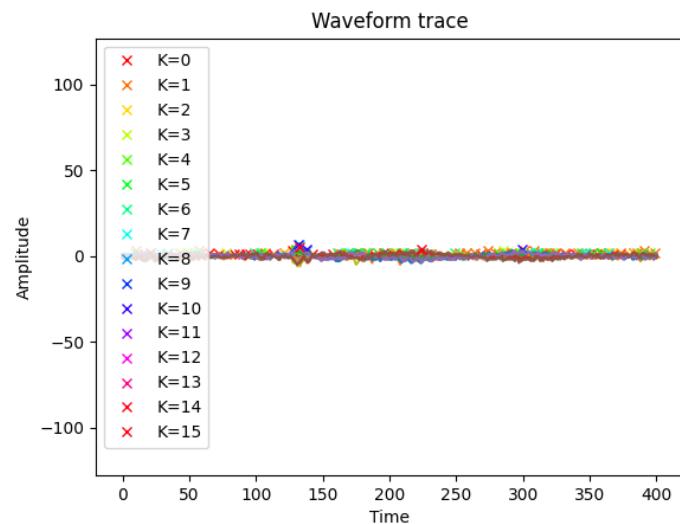


Figure 4: Graph of superimposed difference-of-means

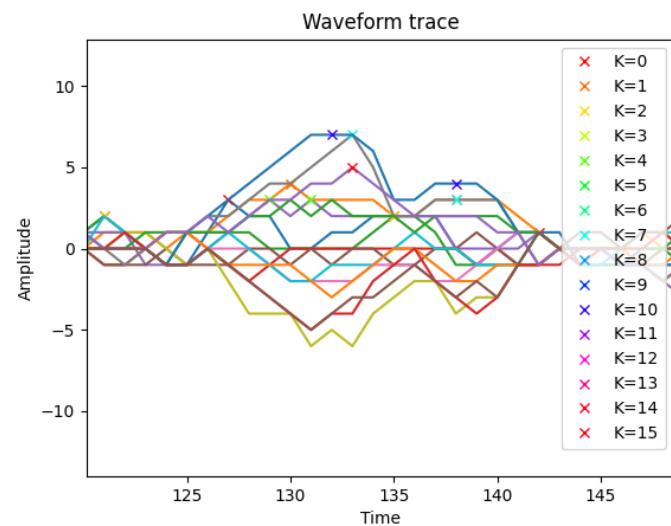


Figure 5: Graph of zoomed-in part of difference-of-means