Cryptographic Engineering

Assignment 6: Correlation Power Analysis

Sushmita Thakur
2752951/s.thakur@student.vu.nl

1 INTRODUCTION

Correlation Power Analysis (CPA) is a technique for recovering secret key bits of a cryptographic algorithm by analyzing the power traces at certain stages of execution. It is a type of side-channel attack that tries to recover the secret key of a cipher using a large number of repeated power measurements. In this report we analyze the power traces measured by the electromagnetic emission of the PRESENT cipher, using a Langer MiniProbe and a Lecroy oscilloscope.

The aim is to recover the secret key by attacking the intermediate value which depends upon the 4-bit portion of the 1st round key of the PRESENT algorithm and the input plain text.

That is done in the following steps-

- The provided dataset which contains 14900 4-bit inputs is used as plain text input (*in_arr*). After generating a set of all possible key values $k \in \{0, 1...15\}$, guessing all the values with 4 bits), a value-prediction (VP) matrix is constructed.
- The resulting bits of XOR operation between the input plain text and key are passed through PRESENT's 4-bit S-box and stored in the VP matrix. The value-prediction matrix is then converted into the power-prediction matrix by using the Hamming weight model (HW).
- In the next step, the traces dataset (*traces_arr*) is used, which contains 14900 aligned power traces, each one with 6990-time samples resulting in a 14900x6990 matrix. The column-wise correlation between the traces matrix and the power-prediction matrix (PP) is computed.
- Using the PP matrix, a maximum score is generated for each key. This basically checks for a column-wise pair of the highest power peak in the traces matrix and bit flips from the PP matrix for each key. The key candidates are then ranked from best to worst, based on the absolute value of the correlation function. The key candidate with the maximum score is declared as the *best_key_candidate*.

The mathematical representation of ciphering with PRESENT structure is as follows: All values(p,k,x,y)have the size of 4 bits, where p represents plain text, k is key and x and y represent intermediate values.

(1) $x = p \oplus k$
(2) $y = S(x)$

In order to make an attack the intermediate value y is used to reverse engineer the process and separate plain text and key.

(1) If v represents S-box output:
$v = y = S(p \oplus k)$
(2) For known x and unknown p, k can be computed as
$k = x \oplus p$
(3) So $VP = S(p_i \oplus key_j)$
(4) $PP_i, j = HW_i, j(VP_i, j) = HW(S(p_i \oplus key_j))$

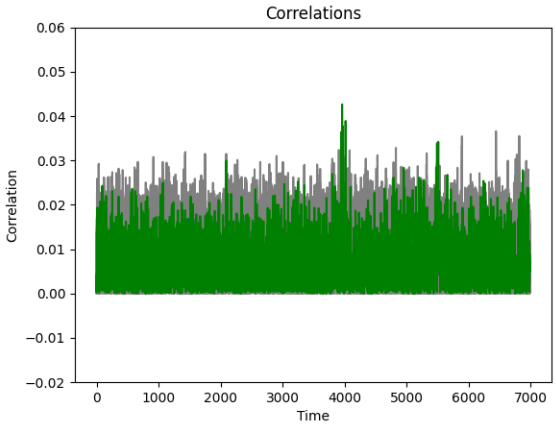

Figure 1: Correlation between PP and traces

(5) $key\_scores = correlation(PP_i, j, traces\_arr_i, j)$
(6) $best\_key\_candidate = max(PP_0...i, j) foreach j$

2 CORRELATION BETWEEN KEY AND POWER CONSUMPTION

Fig 1 shows the absolute correlation value for every key candidate, for every time sample. The correlation value of the power prediction matrix PP and traces dataset (traces_arr) is used to plot this relation. It can be noticed that for the total number of traces key = 6 shows the highest power surge of about 0.04, proving itself to be the best key candidate in doing so.

3 KEY RANKING WITH RESPECT TO TRACES

To analyze the effects of a varying number of traces on the resulting key candidates Fig. 2 is plotted, The attack is executed with 500, 1k, 2k 4k, 8k, and 12k power traces, and for every attack, the correct candidate is ranked based on the absolute correlation value. Fig. 2 clearly shows that for fewer power traces, the resulting key candidate is different from the correct key which has been discovered in the previous step. In fact, the correct key achieves close to the worst rank with the initial test cases (500, 1k, 2k). The correct key does not achieve the best rank until the traces rise up to 12k. Both 12k and 14k test cases result in the same key ($k = 6$) as the best-ranked key. This convergence can be used as a sign that the correct key has been recovered. The number of traces required to discover the right key value can also depend on the amount of noise available in the power values of the traces matrix.

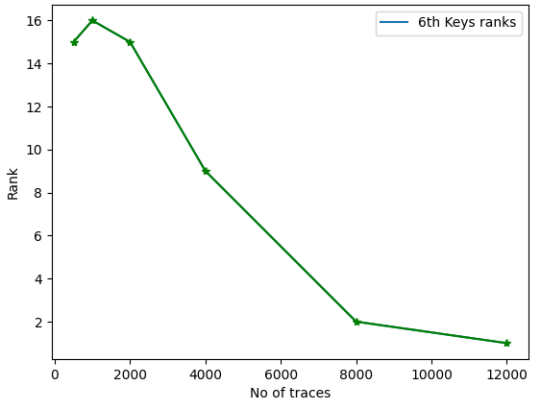

Figure 2: Ranking of Correct key with varying no of traces