Assignment 2: Timing Attack on AES in Software

Due Date: 1. November 2023, 11:00

Important: Please submit your individual solution before the above-specified due date (and time) digitally in the Moodle course.

- You received the file "Timing_Noisy.csv" which contains 1 million timing samples of AES encryptions with random plaintexts (and a fixed key)
- The file format is .csv and consists of 1 million lines, each containing 17 comma separated values: pt_byte0 , pt_byte1 , . . . , pt_byte15 , time_value
- Each value is given by its decimal representation encoded in ASCII
- The task is to recover **all** key bytes as seen in the lecture by means of the **t-test**. Write the correct key in both hex and decimal representation into a file.
  (Deliverable: 1 file: key.txt)
- For each key byte create a separate plot, x-axis: all 256 key guesses, y-axis: t-value. Label your axes. Hint: The correct key byte should stand out by a large margin. (cf. Plot_example_A.png)
  (Deliverables: 16 files: plot_k_t_XX.png)
- Do you need all samples to perform the attack? For each key byte create a separate plot, x-axis: nr of traces used, y-axis: t-value (for all 256 key guesses). Label your axes. You may use increments of 10k traces on the x-axis, highlight the correct key guess. Summarize your findings on the number of necessary traces in two sentences. (cf. Plot_example_B.png)
  (Deliverables: 16 files: plot_nrtraces_t_XX.png, 1 file: nrtraces_explanation.txt)
- Please write your program in any environment that you like and include it in your submission. Please report the approximate run-time of your code to recover the key.
  (Deliverables: source code)
- Include all your files in an archive and name it properly: <lastname>_Assignment2.zip