Assignment Homework 3

Due date: March 2, 2025 (Track 1); Track2+3 aligned with hw4

Please complete this assignment (100 pts total – or 200 pts including bonus) and submit your report/program code on Canvas (all files compressed in one `.zip`). Please note, three different tracks are offered, as discussed in class:

Track 1 (default, homework-oriented, 100 pts + 20 bonus pts)

Please complete the following tasks.

1. For this task, we consider two types of devices. *i*) device of type A implements the AES with just a single S-Box in hardware; *ii*) device of type B implements the AES with 16 S-Boxes to do a full AES round within just one clock cycle. Both devices are now studied as part of a power analysis attack. Let us consider the behavior of the noise in more detail: (30 pts)

   a) In terms of a power analysis, which device is more likely to show higher (algorithmic) noise when assuming randomly distributed plaintext inputs? – Please justify your answer (10 pts)

   Device B exhibits higher algorithmic noise because it implements AES using 16 S-Boxes, allowing a full AES round to be completed in a single clock cycle. As a result, each byte in the state is processed by a different S-Box during AES operations, leading to greater noise in power consumption for each byte. In contrast, device A uses only a single S-Box, meaning all bytes in the state pass through the same S-Box in each round. This results in lower variability noise in power consumption.

   b) Assuming all other device characteristics are the same, which one will be more difficult to attack? Note that difficulty of attack is the number of traces required for a successful key extraction. – Please justify your answer (10 pts)

   Device B is more difficult to attack due to the higher noise introduced by multiple S-Boxes, making power consumption patterns more complex. This increased complexity makes it harder for an attacker to distinguish different byte values based on power consumption, requiring more traces to extract the key successfully. In contrast, device A processes all bytes using the same S-Box, leading to more consistent power consumption patterns. This consistency makes it easier for an attacker to differentiate byte values, reducing the number of traces needed for successful key extraction and thus making the attack easier.

   c) Assuming there are no algorithmic countermeasures, is it possible to adapt the attack on device of type B such that when attacking the first round of AES, the attack would behave similar to device of type A? (*hint*: the attacker only controls the plaintext input) (10 pts)

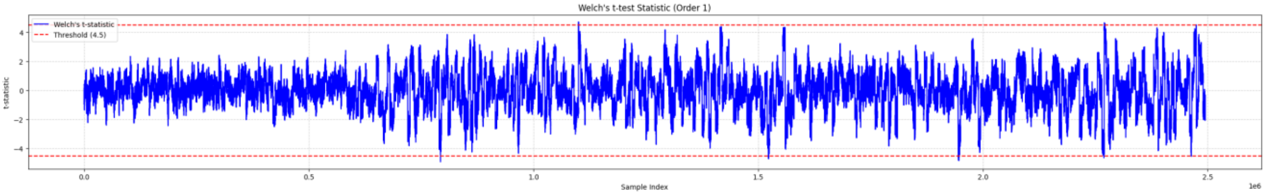
Yes, by carefully controlling the plaintext input, the attacker can adjust the attack on Device B to make its behavior in the first round of AES like that of an attack on Device A. The challenge with Device B is that all 16 S-Boxes execute in parallel, which leads to multiple encryption operations occurring simultaneously, making the power consumption patterns highly mixed and complex. This highly mixed power trace introduces significant algorithmic noise, increasing the difficulty of successfully extracting the key.

However, because the attacker can control the plaintext input, this provides a critical strategy to reduce the noise. Specifically, the attacker can choose to modify one byte of the plain text while keeping the other 15 bytes constant. This way, only one S-Box is involved in each trace, and the other S-Boxes continue to perform the same operations, thus maintaining consistent power consumption patterns. This method effectively isolates the power consumption contribution from a single S-Box, making the power trace for that S-Box easier to distinguish.

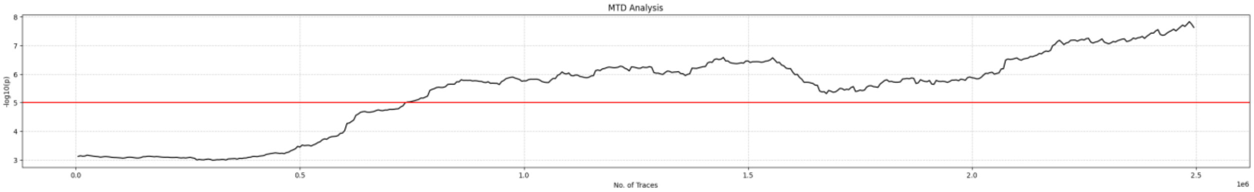
By systematically applying this technique to different S-Box positions, the attacker can analyze each S-Box individually, thus transforming the power consumption pattern to resemble that of Device A, where only one S-Box operates per clock cycle. This simplifies the analysis process. Not only does this reduce algorithmic noise, but it also significantly lowers the number of traces needed to successfully perform the attack, making key extraction from Device B more feasible. Although this method cannot eliminate all sources of noise, it reduces the complexity of the attack process to a level similar to Device A's behavior.

2. This task is about implementing a leakage detection test: Welch's t-test. Please use the following data set from Box.com: `cs5_group0_even_compressed.zarr` and `cs5_group1_even_compressed.zarr` (70 pts + 20 bonus pts)

   a) Implement Welch's t-test as explained in the referenced papers and create the corresponding plots. (40 pts)

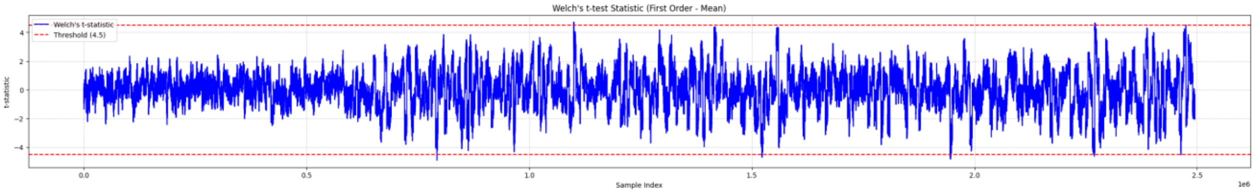


   b) Create a "measurements to disclosure" (MTD) plot for Welch's t-test (30 pts)

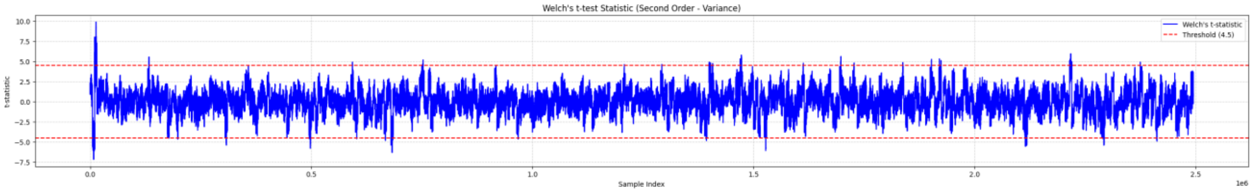


   c) *Bonus:* Compute Welch's t-test not just for distinguishing the first-order moment but add all equations needed to compute the Welch's t-test up to the 4th moment. (20 pts)

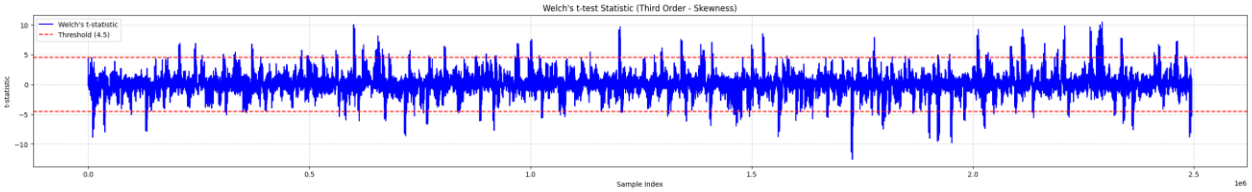
   First:



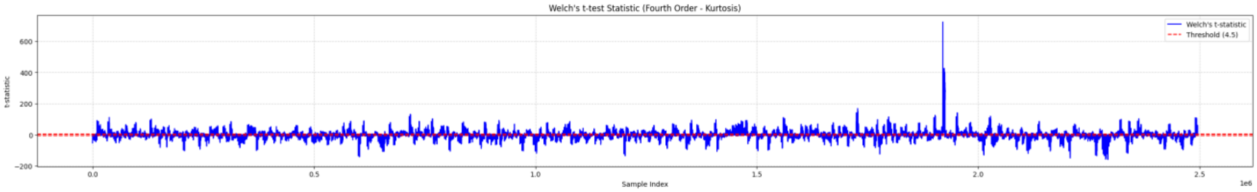
Second:



Third



Forth


Welch's t-test Statistic (Fourth Order - Kurtosis)
Welch's t-statistic
Threshold (4.5)
t-statistic
1e6
Sample Index