Multi-Screaming-Channel Attacks: Frequency Diversity for Enhanced Attacks

Jeremy Guillaume, Maxime Pelcat, Amor Nafkha, and Rubén Salvador, *Member, IEEE*

***Abstract*—Side-channel attacks consist of retrieving internal data from a victim system by analyzing its leakage, which usually requires proximity to the victim in the range of a few millimetres. Screaming channels are EM side channels transmitted at a distance of a few meters. They appear on mixed-signal devices integrating an RF module on the same silicon die as the digital part. Consequently, the side channels are modulated by legitimate RF signal carriers and appear at the harmonics of the digital clock frequency. While initial works have only considered collecting leakage at these harmonics, late work has demonstrated that the leakage is also present at frequencies other than these harmonics. This result significantly increases the number of available frequencies to perform a screaming-channel attack, which can be convenient in an environment where multiple harmonics are polluted. This work studies how this diversity of frequencies carrying leakage can be used to improve attack performance. We first study how to combine multiple frequencies. Second, we demonstrate that frequency combination can improve attack performance and evaluate this improvement according to the performance of the combined frequencies. Finally, we demonstrate the interest of frequency combination in attacks at 15 and, for the first time to the best of our knowledge, at 30 meters. One last important observation is that this frequency combination divides by 2 the number of traces needed to reach a given attack performance.**

***Index Terms*—side-channel attacks, EM side channels, screaming channel attacks, multi-channel attacks.**

I. Introduction

SIDE-CHANNEL ATTACKS [1], [2] allow attackers to retrieve confidential information from computing devices by exploiting the correlation of internal data with the leakage produced while computing over these data. The term *side channel* is therefore used to refer to physical leakage signals carrying confidential information. Side channels are general to CMOS computing devices and can take many forms, from runtime variations of system power consumption [3] to Electromagnetic (EM) emanation[4]. Screaming channels [5] are a specific form of EM side channel that occurs on mixed-signal devices, where a Radio Frequency (RF) module is co-located on the same die as digital modules. In this context, the leakage of the digital part reaches the RF module, which can transmit it over a distance of several meters. This phenomenon allows attackers to mount side-channel attacks at larger distances from the victim than regular side-channel attacks, which traditionally require very close access to targets. The seminal work from Camurati et al. [5] demonstrated how screaming-channel attacks can succeed at distances of up to 15 meters.

Leakage, generated by the switching activity of the transistors from the digital part of the victim system, operates at a clock frequency F_{clk}. When observed on a spectrum analyzer, the leakage power spectral density is shaped as peaks at the harmonics of F_{clk} (*i.e.* $n \times F_{clk}$ where $n \in \mathbb{Z}$). What makes screaming-channel attacks different from other Side-Channel Attacks (SCAs) is that the harmonics, after being modulated by the RF module, are visible around the carrier frequency F_{RF} of the legitimate RF signal. Contrary to previous works, Guillaume et al. demonstrated that exploitable leakage is also present at non-harmonic frequencies [6]. This increases the number of potential frequencies from which an attacker can select to build successful screaming-channel attacks.

Previous studies on side-channel attacks demonstrate that combining multiple channels, *i.e.*, multiple sources of leakage, can improve the attack performance [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17]. Indeed, the different channels can carry complementary information on the target data or be affected by statistically independent noise.

In this paper, **we study how combining the leakage carried by different frequencies can improve screaming-channels attacks performance**. We believe the different frequencies should carry common information on data computed internally by the victim device but should differ in the noise as they are spaced enough to have independent noises. We **propose considering these frequencies as different channels in a multi-channel attack** and answer the following questions.

- **How to combine multiple frequencies, *i.e.*, their carried leakages, for screaming-channel attacks?**
- **What impact does frequency combination have on the performance of screaming-channel attacks?**

This work brings state-of-the-art multi-channel attack methods that combine multiple side channels into the particular case of screaming channels, interpreting the different frequencies as individual leakage channels. Our contributions are:

- We propose **multi-screaming-channel attacks** by considering **different frequencies as separate side channels** in Section III, and we **compare the performance of the different combination methods** to find which performs best for screaming-channel attacks.
- We demonstrate the interest of **frequency combination to improve attack performance** in Section IV in two

Manuscript submitted on April 3, 2025.

Jeremy Guillaume is with Université Bretagne Sud, CNRS, Lab-STICC, Lorient, France (email: jeremy.guillaume@univ-ubs.fr).

Amor Nafkha is with CentraleSupelec Rennes, CNRS, IETR, France (email: amor.nafkha@centralesupelec.fr).

Maxime Pelcat is with INSA Rennes, CNRS, IETR, France (email: maxime.pelcat@insa-rennes.fr).

Ruben Salvador is with CentraleSupelec, Inria, CNRS, IRISA, Rennes, France (email: ruben.salvador@inria.fr).

experiments in laboratory conditions:

  - We show how to **mount successful attacks with fewer traces and analyze the impact of the frequency diversity order** in Section IV-A. This scenario considers attackers that can only collect a limited number of traces, with which all frequencies are too weak for a successful attack.
  - We **quantify the improvement in attack performance** according to the initial performance of the combined frequencies in Section IV-B.

- We prove the interest of **frequency combination in realistic scenarios**, reducing the number of needed traces and increasing the attack distance in Section V:
  - We mount a successful **attack at** 15 **meters with fewer traces** and discuss how the results from our approach compare to current art in Section V-A.
  - We mount the **first successful attack at the largest distance reported,** 30 **meters**, in Section V-B.

The rest of the paper is organized as follows: Section II reviews the background and presents the related work. Section III describes how to repurpose multi-channel attacks as *multi-screaming-channel attacks*, while Section IV proves their interest and quantifies the improvement brought over base attack performance using only one single frequency. Drawing from these observations, in Section V we report more realistic attacks, comparing our work with current art attacking at 15 meters, and reporting the first successful attack at 30 meters. Finally, we discuss our results in Section VI and conclude the paper in Section VII.

II. Related works on multi-channel attacks

In the context of side-channel attacks, a multi-channel attack combines multiple sources of leakage. These attacks are introduced by Agrawal et al. [7]. Authors propose combining traces from an EM and a power channel to attack Data Encryption Standard (DES). Combining multiple channels can increase the quantity of collected information on the targeted data or reduce noise by exploiting the noise independence between the combined channels. This is expected to increase the Signal-to-Noise Ratio (SNR) and, thus, the attack performance.

In this section, we discuss the existing methods to combine multiple channels. We use the classification proposed by Yang et al. [16] to categorize the combination methods in 3 groups: *data fusion*, *feature fusion*, and *decision fusion*. These three combination strategies are illustrated in Fig. 1. Initially, leakage traces are collected from N channels corresponding to any side channel vector like power, EM, timing, etc. In the attack chain, *trace reduction* is a step that can be applied before the analysis attack. Reducing the trace dimensions reduces the computational complexity during the analysis attack since it reduces the number of samples to analyze. This reduction can be done by identifying the Points of Interest (POIs), to keep only these trace samples, or by applying *feature extraction* methods like the well-known Principal Component Analysis (PCA) [18] on the traces. These methods aim to reduce trace dimensions while keeping the components of the traces that maximize the remaining information on the target data after trace reduction. Finally, the *analysis attack* analyzes the leakage traces and returns scores, *i.e.*, a probability for each key hypothesis to correspond to the correct key. In the following, we discuss how the three combination strategies are applied in previous works to combine these N attack chains and obtain a multi-channel attack.

a) Data-fusion: It consists of directly merging the leakage traces collected from the victim over the same operations. For example, Hutter et al. [10] collect power traces from two identical devices that compute the same operations. Authors average their traces, which also averages their noise, thus reducing it. Another way to perform data fusion is by placing traces collected from the different channels in a common trace dataset. For example, Heyszl et al. [12] collect EM leakage of an FPGA using 9 probes placed at different positions. They demonstrate that the attack using the traces from all positions performs better than the best of the individual attacks at each position. Genevey et al. [15] concatenate traces from the different channels and perform analysis attacks based on machine-learning approaches. They observe different improvements in attack performance depending on the contribution of the additional channel; in some cases, the channels can add more noise than additional information.

b) Feature-fusion: It can be considered as a particular case of data fusion. Compared to data fusion, feature fusion does not directly merge traces together. Instead, it jointly extracts features from traces of the multiple channels using feature extraction methods [8], [13], [19], [20]. Feature fusion is proposed by Standaert et al. [8] where they apply feature extraction methods on concatenated traces from two channels, an EM and a power channel. Authors compare Principal Component Analysis (PCA) [18] and Fisher's Linear Discriminant Analysis (LDA) [21] as feature extraction methods and observe that LDA extracts information better than PCA.

c) Decision-fusion: It combines results from mono-channel attacks performed separately for each channel. For example, an analysis attack against Advanced Encryption Standard (AES) returns 256 probabilities, *i.e.*, scores, for each AES sub-key. The highest probability is expected to correspond to the correct sub-key value. In multi-channel attacks at the decision level, the 256 probability values from the different attacks are merged using an aggregation function, *e.g.*, summing or averaging these probabilities. Souissi et al. [11] studied the use of these two aggregate functions, `sum()` and `max()`, and observed that `sum()` performs slightly better than `max()` function in their context.

In the context of screaming-channel attacks, Camurati et al. [22] investigate spatial diversity, *i.e.*, the combination of traces collected at the same frequency with two antennas spaced by 3 cm. The combination is performed with the data fusion strategy, by averaging the traces together. Before this, authors implement a method to equalize channels. To this end, they compare Equal Gain (EG) and Maximal Ratio Combination (MRC) methods and observe that MRC is better than EG. In an attack at 0.55 meter from the victim, the spatial diversity solves the difficulty when attacking in non-Line-of-Sight (nLoS) conditions, *i.e.*, in an environment with obstacles between the attacker and the victim.

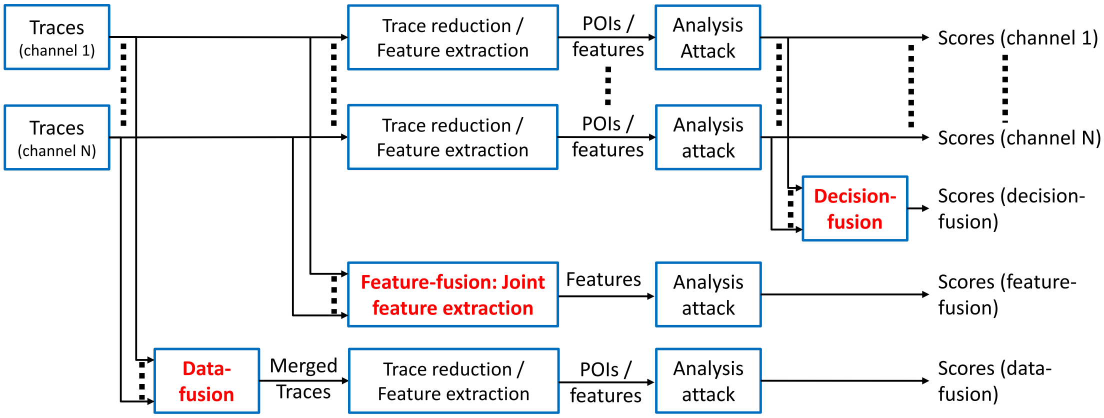


Fig. 1. **State-of-the-art on combination methods:** Leakage from N channels can be combined using 3 different strategies. 1/ Data fusion: merges the trace data from the different channels directly. 2/ Feature fusion: extracts features from data of all channels jointly. 3/ Decision fusion: merges decisions from attacks performed independently on each channel. POIs: Points of interest

III. Multi-Screaming Channel Attacks: Multi-Channel Attacks in the Context of Screaming Channels

Previous work [6] in screaming channels proved how compromising leakage signals could be found and exploited not only at the harmonics of the digital clock frequency of the victim but also widely spread throughout the EM spectrum at non-harmonics. As a result, the authors demonstrated how to mount successful screaming-channel attacks in cases where traditionally used frequencies were highly polluted due to other colliding signals like WiFi.

In this work, we *leverage this fact of leakage being present and exploitable in several locations of the EM spectrum* and propose **multi-screaming channel attacks**, which we formulate as screaming-channel attacks that leverage leakage captured at different frequencies by exploiting combination methods from multi-channel attacks.

In this Section, we evaluate these combination methods when the different channels are considered to be leakage signals captured at different frequencies. Our goal is to find the combination method that fits the best for multi-screaming channel attacks before formally assessing the impact of the combination on attack performance in Section IV.

A. Experimental setup and attack evaluation

To perform our experiments, we collect trace sets at 150 frequencies (from 2.450 GHz to 2.6 GHz, spaced by 1 MHz). The setup is the same as the one used by Guillaume et al. [6], with the victim being a PCA10040[1], that integrates a nRF52832[2], a System on Chip (SoC) from Nordic Semiconductor, and a USRP N210[3] as the radio used by the attacker to collect victim's leakage. To simplify the evaluation of the combination methods and the improvement in attack performance, we consider ideal laboratory conditions for our experiments in this and in Section IV. For this, we collect traces by cable, which avoids environmental pollution and increases the number of exploitable frequencies that can be combined. In section V, to study the impact and build attacks in more realistic conditions, traces are collected at a distance using a directive antenna with a gain of 26 dBi. Traces are collected with a time diversity order of 10, *i.e.*, multiple traces collected from the same leakage source and from Cryptographic Processes (CPs) that compute the same data, *i.e.*, the same plaintext and key, are averaged together. Therefore, under the assumption that the noise has a Gaussian distribution, averaging $N_{TimeDiv}$ segments of noise divides the noise energy by $\sqrt{N_{TimeDiv}}$.

The target victim process runs tinyAES AES-128 compiled without optimizations[4] and running on one instance of the nRF52832. To recover the secret key, the trace reduction phase involves selecting POIs containing information on Intermediate Values (IVs). Then, we perform the profiled correlation attack [23]. Most of the time, the attack does not directly recover the correct key, *i.e.*, the Key Rank (KR), which corresponds to the rank of the correct key among all the possibilities in decreasing order of their probability is higher than 0. In case this KR is low enough, a brute-force attack [24] can test the ranked possibilities and recover the correct key in a reasonable time.

KR is a convenient metric to evaluate attack performance in contexts where the attacks do not always recover the full key. As in previous works on screaming channels [22], we use Guessing Entropy (GE) [25] which is computed from the KR as expressed in Eq. (1) to return concise attack results. When the GE equals 32, our experimental computer (4-core Intel Xeon(R) CPU E3-1226 V3 @ 3.30 GHz and 8 GB RAM) takes about 5 minutes to brute-force the key. When it equals 35, the brute-force attack takes about 1 hour, and approximately 1 day when it equals 39. When performing N attacks in the same conditions to ensure statistically significant results, the attack score will be the average of the N GEs as expressed in Eq. (2).

$$GE = Log_2(\text{KR}). \tag{1}$$

[1] https://www.nordicsemi.com/Software-and-tools/Development-Kits/nRF52-DK.

[2] https://www.nordicsemi.com/Products/nRF52832

[3] https://www.ettus.com/all-products/un210-kit/

[4] tinyAES implementation included in Nordic Semiconductor SDK: https://www.nordicsemi.com/eng/Products/Bluetooth-low-energy/nRF5-SDK.

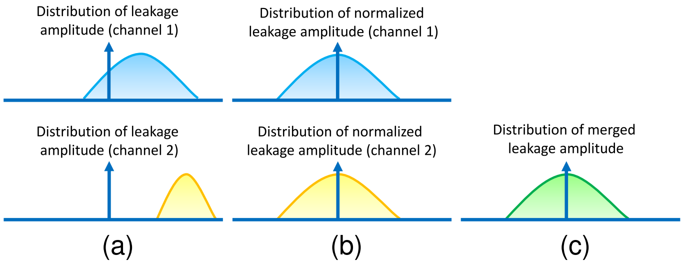

Distribution of leakage amplitude (channel 1)
Distribution of normalized leakage amplitude (channel 1)
Distribution of leakage amplitude (channel 2)
Distribution of normalized leakage amplitude (channel 2)
Distribution of merged leakage amplitude
(a)
(b)
(c)

Fig. 2. **Data-fusion steps:** (a) Leakage amplitude of the two initial channels at samples corresponding to POIs. (b) Normalized values with equivalent amplitudes between the two channels. (c) Merged values: averaging the normalized values.

$$GE = \frac{1}{N}\sum_{i=1}^{N} GE_i. \tag{2}$$

B. Comparing the combination strategies

Starting from the initial attack chain used in previous works, we propose to test and compare the data fusion and decision fusion methods and observe which method fits the best in the context of screaming-channel attacks. The study of feature fusion is kept for future work since we do not use a feature extraction method like PCA or LDA. To simplify the analysis, this first study considers a situation where only two frequencies, $f1$ and $f2$, are combined.

1) Data fusion: The hypothesis under a data fusion strategy is that leakage is expected to have similar amplitude over the two channels but with independent Gaussian distribution noise [10], [12]. Therefore, averaging leakage from these two channels also averages this noise, which, as with time diversity, improves the SNR. However, in the context of screaming channels, the two channels correspond to two frequencies, $f1$ and $f2$, in the radio frequency range where the digital leakage is transmitted. Leakage can undergo different attenuation and distortions through the path between transmitter and receiver from one frequency to the other. Therefore, the leakage can have different amplitudes in the two channels and must be equalized before being combined (Fig. 2a). *Pre-processing* steps are added to achieve this equalization. First, raw leakage amplitudes are normalized for every channel using z-score normalization (Fig. 2b). Thus, the leakage amplitudes from both channels will have the same amplitude range and can be merged. The traces are averaged together, returning a merged trace as if it came from a single channel (Fig. 2c). These merging steps are performed only after the POIs selection, *i.e.*, the trace reduction phase. This reduces the computation complexity since these merging steps are performed only on POIs and not on all the other trace samples not used for the analysis attack. Finally, the profiled correlation attack is computed using the combined values, *i.e.*, the merged traces, exactly as if it were a mono-channel attack.

To analyze the effectiveness of data fusion for frequency combination, we combine the first harmonic at 2.464 GHz with the 150 other collected frequencies. We show the results in Fig. 3. The black curve with circle markers corresponds to the attack performance at each of the initial 150 frequencies, and the orange horizontal line equals the GE of the attack at 2.464 GHz. The green curve with triangle markers corresponds to the combination results. Dashed lines indicate different KR thresholds. As a result of the combination, we expect the green curve to be below the black one, which means that combining individual frequencies with the original at 2.464 GHz improves performance. At each frequency, the combination is tested 20 times, and the results in the figure correspond to the average of the 20 GEs from these combined attacks. Each attack (initial and with combination) is made using 750×10 traces (traces×time diversity).

Fig. 3 shows performance improvements for many of the tested frequencies, however, we observe the combination is not efficient all along the spectrum. For example, the combination of 2.464 GHz is working well with 2.465 GHz and further frequencies like 2.596 GHz but not with 2.592 GHz. To try to explain this behavior, we show the *leakage profiles*, *i.e.*, the profiled leakage amplitude according to the IV, across the 256 possible IVs in Fig. 4. When looking at the leakage profile at 2.464 GHz, we see it is similar to the profile at 2.465 GHz, as shown in Fig. 4a, which is a frequency where the combination is effective. On the contrary, the profile is inverted with respect to the one at 2.592 GHz, as shown in Fig. 4b. This problem can be corrected by inverting the values from 2.592 GHz, which renders the combination effective. However, this adds the constraint of checking if two frequencies can be directly combined according to the similarity of their leakage profiles.

The *leakage distortion* observed and analyzed by Camurati et al. [22] could explain why the profile differs from one frequency to another. Indeed, this distortion, which is expected to happen through the path between leakage generation from the digital part to the RF module, can be different from one frequency to another. From our observations, this distortion over the spectrum is stable over time, *i.e.*, the same result is obtained when repeating the combination between two identical frequencies multiple times.

2) Decision fusion: With the decision fusion method, the combination consists of performing independent mono-channel attacks, one attack for each collected channel, and then combining the attack scores. The attack chain is exactly the same as in a mono-channel attack, with the addition of an aggregation function used to combine the decisions from the individual attacks.

In a first step, we propose to compare 3 different aggregation functions used in previous works to combine scores:

- `average(scores` $f1$`, scores` $f2$`)` [11]
- `maximum(scores` $f1$`, scores` $f2$`)` [11]
- `product(scores` $f1$`, scores` $f2$`)` [22]

To this end, a similar experiment as the one evaluating data fusion over the 150 frequencies is done with decision fusion. To distinguish which of the 3 aggregation functions is the most efficient, we go into a more challenging situation by selecting a weaker frequency than 2.464 GHz. When using 2.464 GHz, the combinations always return very low GEs, mainly because 2.464 GHz initially has a low GE, making the contribution from the combination more difficult to observe.

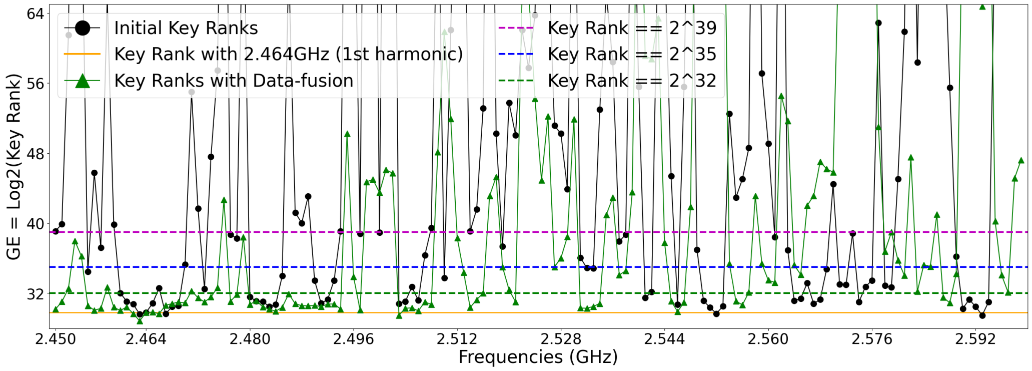
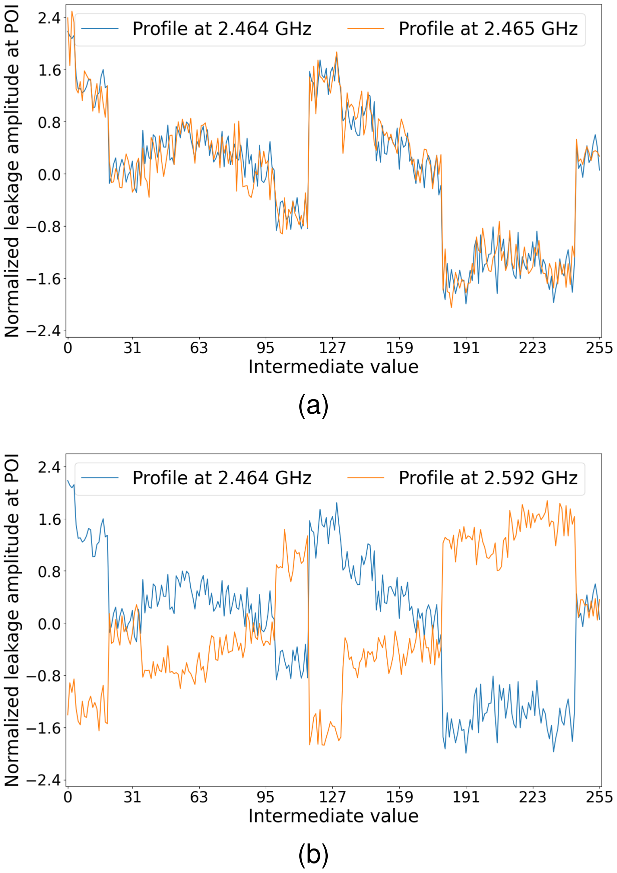


Fig. 3. **Data fusion:** Combinations of 2.464 GHz with 150 frequencies using data fusion. The lower the GE, the better the result of the direct combination.



(a)

(b)

Fig. 4. **Profile similarity:** Normalized leakage profiles for byte 0 at (a) two frequencies with similar profiles: 2.464 GHz and 2.465 GHz; and (b) two frequencies with inverted leakage profiles: 2.464 GHz and 2.592 GHz.

TABLE I

COMPARISON OF AGGREGATION FUNCTIONS FOR DECISION FUSION.

Aggregation function	`AVG()`	`MAX()`	`PROD()`
Number of improvements[a]	84	83	31
Sum of GE reduction[b]	407	99	67

[a] Number of improvements: Corresponds to the number of frequencies where the combination improves performance.

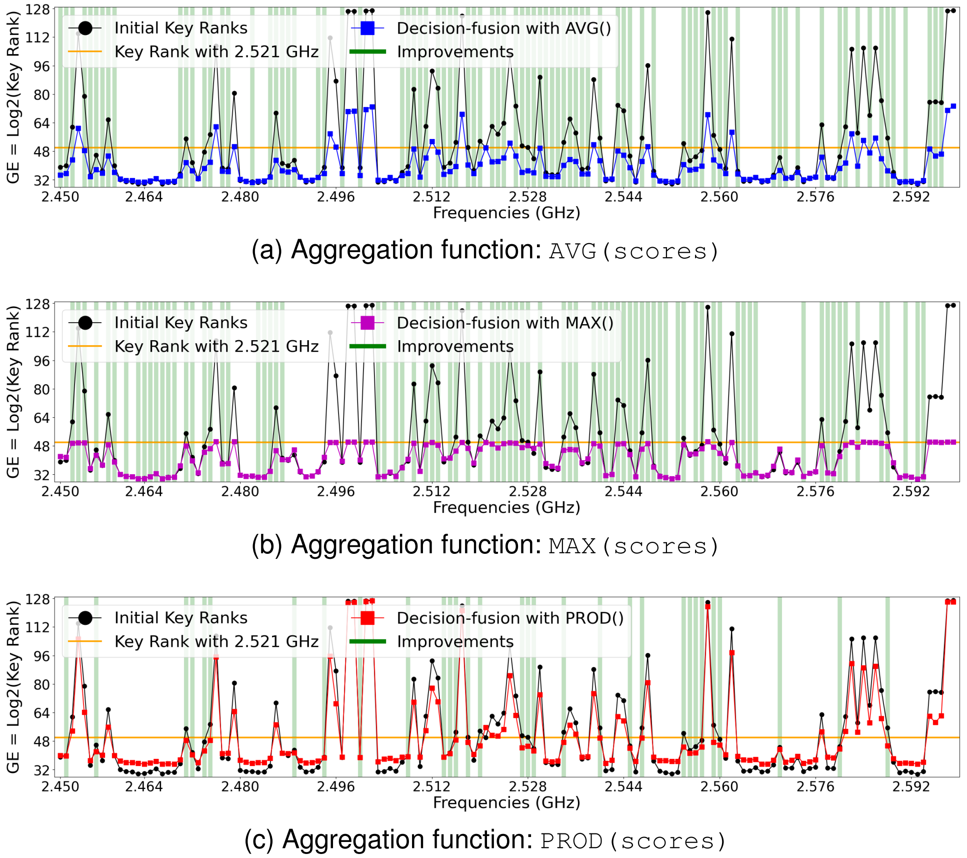
[b] Sum of GE reduction: The sum, over the 150 frequencies, of the difference between the lowest GE of combined frequencies and the GE of the combination.

For this experiment, we select the frequency 2.521 GHz because it has a GE of 50. A brute-force attack is considered feasible in a reasonable time only if the GE is less than or equal to 39. This GE threshold value is based on the capacity of our experimental computer to brute-force the key and could vary with the time of brute force considered reasonable by the attacker. With GE=39, our computer runs near to 24 hours to retrieve the key. This time can be decreased with a more powerful computer. Although a GE of 50 is close to a reasonable GE, it is too high for a successful attack since the experimental computer would need at least multiple weeks (even months) to brute-force the key. Therefore, it is an interesting case study to analyze which aggregation functions make the attack feasible by combining the frequency 2.521 GHz with other frequencies.

The frequency 2.521 GHz is combined with the remaining 150 frequencies using the 3 tested functions, and combination results are shown in Fig. 5. Every combination result corresponds to the average of 20 attacks using 750×10 traces each. Table I summarizes the improvement for the 3 functions. The improvement equals the lowest GE of the two combined frequencies minus the GE of the combination. The combination improves attack performance when this subtraction is positive, *i.e.*, when the combination reduces GE compared to the lowest GE of the initial attacks. To compare the 3 aggregation functions, the first row of Table I indicates the number of frequencies where the combination brings improvement. The second row corresponds to the sum of the improvement over all frequencies where it exists.

When looking at Fig. 5, one could intuitively argue that the `max()` function performs better as the GE never goes higher than 50. However, it mostly follows the results of the best frequency being combined and does not bring particular improvement. As highlighted by the second row of the table I, the `avg()` function is the one that returns the best improvement over the 150 combinations. In particular, it shows a better GE reduction in challenging cases where the two combined frequencies have high GE.

In the second step, we compare decision fusion with data fusion by analyzing the results shown in Fig. 6. This figure shows the result of a similar experiment as in III-B1: the combination between the first harmonic (2.464 GHz) with the 150 initial frequencies, but using both data fusion (green line with triangle markers) and decision fusion (with the `avg()`



(a) Aggregation function: `AVG(scores)`

(b) Aggregation function: `MAX(scores)`

(c) Aggregation function: `PROD(scores)`

Fig. 5. **Aggregation functions for decision fusion:** Combinations of 2.521 GHz (orange horizontal line) with the 150 frequencies (black circles). The frequencies where the combination improves attack performance are highlighted in green. The results for the 3 aggregation functions are shown in (5a) for the average, (5b) for the maximum and (5c) for the product of scores.

function) (blue line with square markers). It highlights the advantage of decision fusion over data fusion in combining multiple frequencies in the context of a screaming-channel attack: while data fusion is inefficient at some frequencies without any adaptation between frequencies, decision fusion fits all along the spectrum. Additionally, it can be observed that when data fusion is effective, it does not bring better improvement compared to decision fusion. Therefore, data fusion and decision fusion have similar performance when both are effective, with the advantage of decision fusion being independent of profile similarity between the combined frequencies, making it more interesting in practice.

In the remainder of this paper, to simplify the analysis, only the results from the most effective combination method are presented: the decision fusion with the `avg()` function. However, we notice that it is not essential for an attacker to know which combination method is the most effective in a given case. It is possible to compute multiple combination methods in parallel and wait until one GE from individual attacks or combination methods becomes low enough for a reasonable brute-force attack.

IV. Improving the Attack Performance with Frequency Combination

In this section, we highlight the interest in frequency diversity and quantify the improvement that frequency combination can bring in attack performance. We perform two studies:

1) Combinations of frequencies that are individually too weak for successful attacks. We analyze how increasing the frequency diversity order can make attacks feasible.
2) An analysis of the combined attack performance according to the initial performance of the combined frequencies. This analysis is done in two cases: combinations of frequencies having equivalent and non-equivalent performance.

A. Making the attack feasible with weak frequencies

In the first study, we consider a *scenario where the attacker can collect a limited number of traces*. This number of traces is too low to build a successful attack for the best frequency among the 150, *i.e.*, the GE from the best frequency is above 39. We investigate the improvement of the attack in these conditions according to the number of combined frequencies. More specifically, the question is: **can frequency diversity decrease the GE enough to make the attack feasible?**

The number of traces per attack is limited to 20×10 (traces×time diversity). Under these conditions, the best of the 150 frequencies has a GE close to 50. As explained in Section III-B2, with a GE of 50, one can consider the brute-force attack to be unfeasible in a reasonable time; at least multiple weeks would be necessary to retrieve the key. With only 20×10 traces per attack, there are enough traces in the

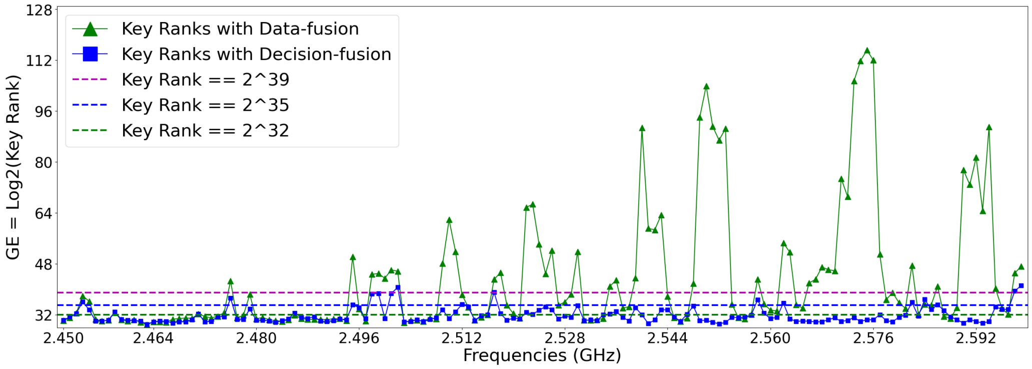


Fig. 6. **Data vs Decision fusion:** Combinations of 2.464 GHz with the 150 collected frequencies using both data fusion and decision fusion methods. The lower the GE, the better the result of the direct combination. While data fusion does not work with all frequencies, decision fusion returns a low GE (below 39) almost all along the spectrum.

trace sets to perform 100 attacks at each frequency. Therefore, each attack result in this experiment corresponds to the average of 100 GEs, guaranteeing strong statistical stability.

We perform the experiment presented in Algorithm 1. The 150 frequencies are sorted from best to worst, *i.e.*, from f_0 having the lowest GE to f_{149} having the highest GE. The GE of f_0 equals 49.9. The best frequency f_0 (included in $Freqs_{combined}$) is combined with the 150 initial frequencies (lines 5 – 8). The best of the 1^{st} order combinations get a GE equal to 41.8. This GE value is still very high as it would require at least one week to brute-force the key. Nevertheless, it is getting closer to a reasonable brute-force attack.

The best 1^{st} order combination corresponds to the one between f_0 and the second best frequency f_1. Therefore, f_1 is added to the list $Freqs_{combined}$ (line 9). To increase the frequency diversity order, another iteration is computed (lines 3 – 11) to combine the frequencies included in $Freqs_{combined}$: *e.g.*, f_0 and f_1 after the first iteration, with the 150 initial frequencies. Frequency diversity is increased up to the 4^{th} order, and the best GE, initially equal to 49.9, goes as low as 36.4, making a brute-force attack possible in a few hours.

These experimental results are shown in Fig. 7 summarized in Table II, showing how **frequency diversity can make attacks feasible when the number of CPs computed by the victim is too low for a successful mono-channel attack**. In essence, frequency diversity allows for increasing the number of traces by collecting traces at different frequencies. As we could expect, each time the diversity order is increased, the best combination is obtained with the next best frequency that is not included in $Freqs_{combined}$ yet. At the end of the 4^{th} iteration, $Freqs_{combined} = [f_0 : f_1 : f_2 : f_3 : f_4]$, with frequencies f_i by order of addition in $Freqs_{combined}$. One observation is that the more we increase the diversity order, the lower the new GE reduction is. A potential explanation is that most of the information carried by the new frequencies has already been brought by previous frequencies. We thus observe a law of diminishing return, while the increase in the setup complexity is linear. Indeed, collecting an additional frequency simultaneously increases the setup complexity since an additional Software Defined Radio (SDR) must be used.

Algorithm 1 Finding the best combination for each frequency diversity order

```
Require: Scores of the 150 frequencies: $f_i$ with $i \in [0; 149]$
    {frequencies sorted from best to worst}
Require: $Limit\_FreqDiv_{order}$ {Required frequency diversity order}
Ensure: $Lowest\_GE$
 1: $Freqs_{combined}$ = $[f_0]$
 2: $FreqDiv_{order} \leftarrow 1$
 3: repeat
 4:     $i \leftarrow 0$
 5:     repeat
 6:         $GEs\_Combination[\boldsymbol{i}] =$
            $Decision\_Fusion(Freqs_{combined}$ & $f_{\boldsymbol{i}})$
 7:         $i \leftarrow i + 1$
 8:     until $i >= 150$
 9:     $Freqs_{combined}.append(ArgMin(GEs\_Combination))$
        {the frequency returning the best combination is added
        to the list of frequencies to combine}
10:     $FreqDiv_{order} \leftarrow FreqDiv_{order} + 1$
11: until $FreqDiv_{order} > Limit\_FreqDiv_{order}$
12: $Lowest\_GE \leftarrow Min(GEs\_Combination)$
```

TABLE II
GE ACCORDING TO FREQUENCY DIVERSITY ORDER.

Diversity order	0	1^{st}	2^{nd}	3^{rd}	4^{th}
$Freqs_{combined}$	f_0	f_0:f_1	f_0:f_1:f_2	f_0:f_1:f_2:f_3	f_0:f_1:f_2:f_3:f_4
GE	49.9	41.8	39.2	37.5	36.4

f_i: i^{th} frequency among the 150 frequencies, sorted according to their individual performance.

B. Combination efficiency according to the performance of the combined frequencies

In this second study, we evaluate the improvement provided by frequency diversity according to the performance of the initial frequencies. We analyze under which conditions it is interesting to collect and combine frequencies together. To simplify the analysis, we focus on the combinations of only two frequencies. Nevertheless, our results can theoretically be generalized to any number of combined frequencies. Two cases

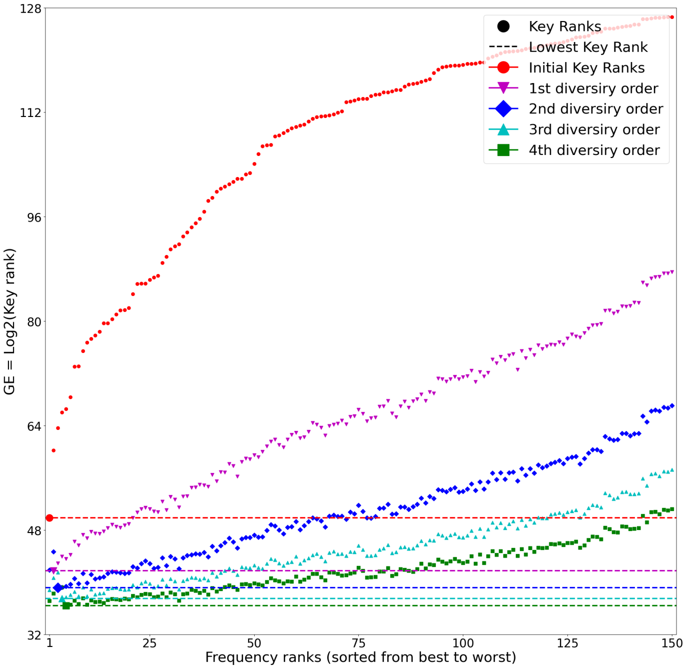


Fig. 7. **Guessing Entropy (GE) according to frequency diversity order:** (Red) The 150 frequencies are sorted according to their GE (red points) from the best to the worst. The number of traces is low enough for the best frequency to have a GE close to 50. With 20 × 10 traces, the best GE equals 49.9 (red horizontal dashed line). (Purple) First-order diversity: the combination of the best frequency with all others (purple points). The new lower GE equals 41.8 (purple horizontal dashed line). (Blue) Second-order diversity: best combination from first-order combined with all initial frequencies, best GE = 39.2. (Light blue) Third-order diversity, best GE = 37.5. (Green) Fourth-order diversity, best GE = 36.4.

are identified:

1) Combination of two equivalent frequencies, *i.e.*, frequencies with equivalent GEs.
2) Combination of two non-equivalent frequencies, *i.e.*, frequencies with different GEs. We try to answer the question: **How does the weakest contribute to increasing or decreasing the performance of the strongest?**

Table III gives results for the first case, and Table IV for the second. Each result of initial and combined attacks in these tables corresponds to an average of 20 attacks using 750 × 10 traces each. To evaluate the combination improvement, both tables give, for a fixed number of traces (750 × 10), the GE of each combined frequency and the GE of the combination.

The first observation is that the higher the GEs of combined frequencies, the more improvement the combination brings. Starting from the first row, in Table III, GEs from initial frequencies increase by 5 between most rows, except for the last two rows, where it increases by 10. Regarding the GE of the combination, it increases at a slower pace: between 2 and 3, and it increases by 8 between the last two rows. Therefore, when the GEs from combined frequencies increase, the difference between their GE and the GE of the combination increases, too.

A second observation when looking at both Table III and IV is that the combination of equivalent frequencies, having each a GE $= ge$, returns a similar result as the combination of non-equivalent frequencies with $Avg(GEs) = ge$. For example, combining two non-equivalent frequencies with GEs of 40 and 50 is similar to combining two equivalent frequencies having GEs of 45.

Additionally, the tables indicate the number of traces to reach given GEs of 39 and 35. We observe that the combination is also efficient when the initial frequencies are good enough to reach these GEs with less than 750×10 traces (rows 1&2 in Table III). Frequency diversity reduces the number of traces needed to obtain these GEs.

The key takeaways of these observations are: (1) **how frequencies otherwise considered useless for carrying too weak of a leakage component can be combined together to build a successful attack**; and (2) **how the combination is still efficient with strong frequencies, improving the required number of traces to reach a given GE.**

V. Attacking in More Realistic Conditions

The limitation of our studies in the previous section is that traces from the different frequencies are collected by cable and at different times. In this Section **we demonstrate the benefits of frequency diversity for multi-screaming channel attacks and verify our observations in more realistic conditions**. We simultaneously collect leakage at two different frequencies using 2 SDRs and **build successful attacks at challenging distances of** 15 **and, for the first time reported in the literature,** 30 **meters**. Profiles are built during a profiling phase on one instance of the nRF52832, collecting leakage by cable. During the attack phase, a directional antenna collects leakage at a distance of multiple meters from the victim, which is another instance of the same device.

A. Attack results at 15 meters

To compare our results with previous works on screaming-channel attacks at 15 meters [22], [26], [27], we reproduce the attack under similar conditions as Camurati et al. [22]: 10000×500 profiling traces and 5000×500 attacking traces are collected at each selected frequency. However, a proper comparison of attack performance is difficult as the conditions differ: environment, RF pollution, etc. As shown in Fig 8a, when using the second harmonic (2.528 GHz), the results differ: The GE from Camurati et al. decreases slowly but continuously. On the contrary, our GE decreases fast at first but then stabilizes and does not strictly decrease, as we can see in Fig. 8a between traces 1000 and 2000 and also between traces 3000 and 4000. This is probably because this frequency was partially polluted in our environment. Thus, some traces decrease the attack performance, likely because they mostly collect noise. It is also difficult to make a proper comparison with works from Wang et al. [26], [27] since they evaluate their attack on the recovery of only one key byte. Therefore, we evaluate the combination by comparing its performance with the individual performance of the combined frequencies.

To consider a realistic use case, and since the second harmonic leakage is very strong when not polluted, we consider a situation where this frequency would be too polluted so the attacker would have to use other non-harmonic frequencies.

TABLE III
Combinations of two equivalent frequencies: Combination of 2 frequencies having a similar performance, *i.e.*, a similar GE when using the same number of traces (750×10 traces here).

Frequency 1		Frequency 2		Combination			Best frequency	
Freq (GHz)	GE	Freq (GHz)	GE	GE	min_{traces} for GE		min_{traces} for GE	
					< 39	< 35	< 39	< 35
2.552	29.7	2.593	29.5	29.3	36	52	50	76
2.470	35.3	2.532	34.9	32.2	185	240	405	719
2.488	40.0	2.508	39.5	34.5	400	645	>750	>750
2.456	45.8	2.545	45.4	37.5	645	>750	>750	>750
2.521	50.1	2.560	49.1	39.7	>750	>750	>750	>750
2.452	61.5	2.511	62.1	47.4	>750	>750	>750	>750

GE: Guessing entropy.
$min_{traces} < 39$ & 35: minimum number of traces required for the GE to be lower than 39 & 35 respectively.

TABLE IV
Combinations of two non-equivalent frequencies: Combination of 2 frequencies not having a similar performance, *i.e.*, different GEs when using the same number of traces (750×10 traces here).

Frequency 1		Frequency 2		Combination			Best frequency	
Freq (GHz)	GE	Freq (GHz)	GE	GE	min_{traces} for GE		min_{traces} for GE	
					< 39	< 35	< 39	< 35
2.470	35.3	2.488	40.0	33.6	252	500	431	>750
2.470	35.3	2.456	45.8	33.3	283	531	431	>750
2.470	35.3	2.521	50.1	34.2	310	639	431	>750
2.488	40.0	2.456	45.8	35.2	425	>750	>750	>750
2.488	40.0	2.521	50.1	36.5	499	>750	>750	>750
2.488	40.0	2.452	61.5	39.6	>750	>750	>750	>750

GE: Guessing entropy.
$min_{traces} < 39$ & 35: minimum number of traces required for the GE to be lower than 39 & 35 respectively.

We select the two non-harmonic frequencies that performed the best at 7 meters in the works exploring leakage at non-harmonics [6]: 2.484 GHz and 2.593 GHz, and compare the result of the combination with the attack performance at each individual frequency. Fig. 8b shows the results of both individual frequencies and the result of their combination according to the number of attack traces. Each attack result is the average of 5 attacks using 1000×500 traces. Table in Fig. 8 summarizes the performance of each frequency and their combination. To get a GE of 32, the two frequencies need 410×500 and 686×500 traces. When combined, 136×500 traces are enough to get the same result. We can see that their combination improves the attack by decreasing the number of necessary traces to reach the lowest GE. However, it does not decrease the reachable GE. When both frequencies reach a limit, their combination does not bring any improvement but stabilizes around 32. We hypothesize that these non-harmonic frequencies transmit information on fewer bits than the second harmonic. Then, even decreasing the noise level by combining the two frequencies cannot allow the recovery of information not carried at these frequencies.

B. Attack results at 30 meters

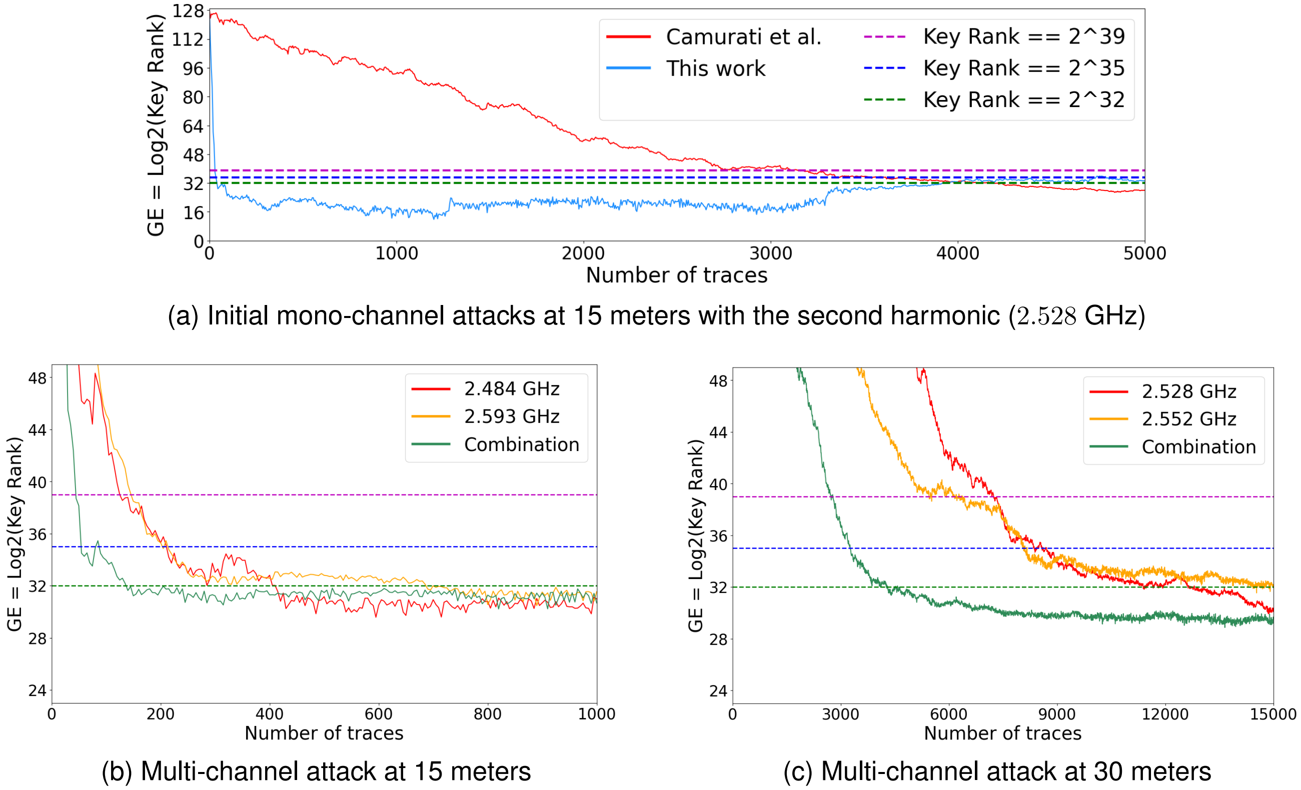
Going beyond previous works, we now attack at a larger distance of 30 meters, where it is more difficult to find frequencies with observable leakage patterns. In the environment where this experiment is performed, the second harmonic is not polluted. We select 2.528 GHz and 2.552 GHz by empirically testing frequencies where the traces correspond to the leakage pattern. Our observation highlights a potential interest in harmonics compared to non-harmonics, which is that leakage seems to be visible at further distances at these frequencies. To have a better resolution on the number of traces needed to get a given GE, we reduce the time diversity compared to the attack at 15 meters and go from 500 to 50.

Fig. 8c shows the results vs. the number of traces. Here again, each result corresponds to the average of 5 attacks in the same conditions. We can clearly see how the combination divides by 2 the number of traces needed to get a GE lower than 39 compared to the best of the initial frequencies.

VI. Discussion

In previous studies on screaming-channel attacks, increasing the number of collected traces is the only solution to reduce the GE until it becomes low enough for a brute-force attack to be possible in a reasonable time. In our experiments for **multi-screaming channel attacks**, by increasing the frequency diversity order, we demonstrated a successful attack in a situation where the number of CPs executed by the victim is not enough for any mono-channel attack to succeed. Therefore, **increasing the frequency diversity order for screaming-channel attacks by collecting traces at different frequencies can effectively increase the number of collected traces without having the victim execute more CPs**. We have shown how even if these traces contain leakage from the same CP executions, their combination increases attack performance. Our results contribute to a **more realistic threat model** where a lower number of CP executions is needed for a successful attack.

We confirm, at a distance of 15 meters, the results from Guillaume et al [6] that attacking at non-harmonics can perform similarly to attacking at harmonics. The results also



(a) Initial mono-channel attacks at 15 meters with the second harmonic (2.528 GHz)

(b) Multi-channel attack at 15 meters

(c) Multi-channel attack at 30 meters

Distance	Frequency	min_{traces} for GE		
		<39	<35	<32
15 meters	2.484 GHz	125 × 500	210 × 500	410 × 500
	2.593 GHz	146 × 500	203 × 500	686 × 500
	Combination	45 × 500	54 × 500	136 × 500
30 meters	2.528 GHz	7 242 × 50	8 370 × 50	11 390 × 50
	2.552 GHz	5 452 × 50	8 070 × 50	14 439 × 50
	Combination	2 727 × 50	3 264 × 50	4 105 × 50

Fig. 8. **Experimental results:** (a) Comparison of attacks at 15 meters using the second harmonic (2.528 GHz) in (red) Camurati et al. conditions [22] and (blue) this work conditions. Initial frequencies vs their combination at (b) 15 meters and (c) 30 meters.

highlight how the combination of non-harmonics, in a scenario where harmonics would be polluted, makes it possible to get closer to the performance obtained with harmonics.

Our study considers a GE to be reasonable for a brute-force attack when equal to or less than 39, making a brute-force attack possible in less than 24 hours. In our experiments, combining two frequencies with GEs close to 50 returns a GE very close to 39. Then, an attack with frequency diversity performs similarly to a mono-channel attack whose reasonable GE would be 50 instead of 39. With a more powerful computer capable of brute-forcing a key with higher GE, *e.g.*, 48, according to our results, it would be possible to succeed in an attack with two frequencies having individual GE of 60. The results of this paper increase the number of exploitable frequencies. Since weaker frequencies with higher GEs can make the attack possible when combined together, the surface of potential frequencies to use is increased again.

However, we did not demonstrate that frequency diversity can lower the reachable GE, *i.e.*, the limit to which the GE stabilizes to when enough traces are collected. Our hypothesis, whose study is kept for future work, is that the combined frequencies do not carry information on all key bits. Frequency diversity increases the SNR by reducing the noise, but it cannot solve this lack of information.

A limitation of the first studies in this paper is that they are done with traces collected at different times. However, we assume the results obtained in these conditions are similar if frequencies were collected simultaneously since the frequencies are spaced enough to be in different coherence zones and thus expected to carry independent noise. We verified this hypothesis in our last experiment with leakage from common CPs collected simultaneously at two frequencies and whose combination improves attack performance.

VII. Conclusion

In this paper, we have studied the interest of frequency diversity in the context of screaming-channels attacks and proposed multi-screaming channel attacks. We have demonstrated the effectiveness of the possible methods to combine frequencies and analyzed the conditions under which this diversity is the most interesting.

Two combination methods are studied: data fusion and decision fusion. To compare them, we combined the second harmonic with 150 frequencies along the considered range of the spectrum and showed that decision fusion is superior to data fusion as the combination works throughout the spectrum. On the contrary, data fusion depends on profile similarity between combined frequencies.

To study the improvement of multi-screaming channel attacks, we proposed two case studies. First, we demonstrated the improvement in attack performance in a situation where the number of traces is too low to mount a successful mono-channel attack. In these conditions, frequency diversity keeps the attack feasible. Second, we quantify the improvement according to the performance of the combined frequencies by looking into the reduction of both the GE and the number of traces needed to get exploitable GEs. We demonstrate that combinations of both equivalent and non-equivalent frequencies are of interest. Finally, we show the interest of frequency diversity in realistic cases with attacks performed at 15 and, for the first time reported, at 30 meters, collecting the leakage simultaneously at two frequencies.

In this work, when multiple frequencies are combined, they all have the same weight in the final decision, even when combining two non-equivalent frequencies. One potential future work is to investigate the addition of weights for each frequency according to their respective performance. Also, we only focus on the right side of the spectrum with respect to the legitimate signal at 2.4 GHz. Another possible line of work would be to study the combination of symmetric frequencies: *i.e.*, 2.4 GHz + f and 2.4 GHz - f.

References

[1] J. Choi, H.-Y. Yang, and D.-H. Cho, "Tempest comeback: A realistic audio eavesdropping threat on mixed-signal socs," in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, 2020.

[2] F.-X. Standaert, "Introduction to side-channel attacks," *Secure integrated circuits and systems*, 2010.

[3] S. Mangard, E. Oswald, and T. Popp, *Power analysis attacks: Revealing the secrets of smart cards*. Springer Science & Business Media, 2008, vol. 31.

[4] K. Gandolfi, C. Mourtel, and F. Olivier, "Electromagnetic analysis: Concrete results," in *Cryptographic Hardware and Embedded Systems–CHES*. Springer, 2001.

[5] G. Camurati, S. Poeplau, M. Muench, T. Hayes, and A. Francillon, "Screaming channels: When electromagnetic side channels meet radio transceivers," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018.

[6] J. Guillaume, M. Pelcat, A. Nafkha, and R. Salvador, "Attacking at non-harmonic frequencies in screaming-channel attacks," in *International Conference on Smart Card Research and Advanced Applications*. Springer, 2023.

[7] D. Agrawal, J. R. Rao, and P. Rohatgi, "Multi-channel attacks," in *Cryptographic Hardware and Embedded Systems–CHES 2003*. Springer, 2003.

[8] F.-X. Standaert and C. Archambeau, "Using subspace-based template attacks to compare and combine power and electromagnetic information leakages," in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2008.

[9] M. A. Elaabid, O. Meynard, S. Guilley, and J.-L. Danger, "Combined side-channel attacks," in *Information Security Applications: 11th International Workshop, WISA 2010*. Springer, 2011.

[10] M. Hutter, M. Kirschbaum, T. Plos, J.-M. Schmidt, and S. Mangard, "Exploiting the difference of side-channel leakages," in *Constructive Side-Channel Analysis and Secure Design: Third International Workshop, COSADE*. Springer, 2012.

[11] Y. Souissi, S. Bhasin, S. Guilley, M. Nassar, and J.-L. Danger, "Towards different flavors of combined side channel attacks," in *Topics in Cryptology–CT-RSA 2012: The Cryptographers' Track at the RSA Conference*. Springer, 2012, pp. 245–259.

[12] J. Heyszl, A. Ibing, S. Mangard, F. De Santis, and G. Sigl, "Clustering algorithms for non-profiled single-execution attacks on exponentiations," in *Smart Card Research and Advanced Applications: 12th International Conference, CARDIS*. Springer, 2014.

[13] R. Specht, J. Heyszl, M. Kleinsteuber, and G. Sigl, "Improving non-profiled attacks on exponentiations based on clustering and extracting leakage from multi-channel high-resolution em measurements," in *International Workshop on Constructive Side-Channel Analysis and Secure Design*. Springer, 2015.

[14] W. Yang, Y. Zhou, Y. Cao, H. Zhang, Q. Zhang, and H. Wang, "Multi-channel fusion attacks," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 8, 2017.

[15] C. Genevey-Metat, B. Gérard, and A. Heuser, "Combining sources of side-channel information," in *C&ESAR 2019*, 2019.

[16] W. Yang, X. Xiang, C. Huang, A. Fu, and Y. Yang, "Mca-based multi-channel fusion attacks against cryptographic implementations," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 2023.

[17] P. Ayoub, A. Hernandez, R. Cayre, A. Francillon, and C. Maurice, "Phasesca: Exploiting phase-modulated emanations in side channels," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2025, no. 1, 2025.

[18] T. Kurita, "Principal component analysis (pca)," *Computer Vision: A Reference Guide*, 2019.

[19] R. Specht, V. Immler, F. Unterstein, J. Heyszl, and G. Sig, "Dividing the threshold: Multi-probe localized em analysis on threshold implementations," in *IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. IEEE, 2018.

[20] B. Hettwer, D. Fennes, S. Leger, J. Richter-Brockmann, S. Gehrer, and T. Güneysu, "Deep learning multi-channel fusion attack against side-channel protected hardware," in *57th ACM/IEEE Design Automation Conference (DAC)*. IEEE, 2020.

[21] P. Xanthopoulos, P. M. Pardalos, T. B. Trafalis, P. Xanthopoulos, P. M. Pardalos, and T. B. Trafalis, "Linear discriminant analysis," *Robust data mining*, 2013.

[22] G. Camurati, A. Francillon, and F.-X. Standaert, "Understanding screaming channels: From a detailed analysis to improved attacks," *IACR transactions on cryptographic hardware and embedded systems*, no. 3, 2020.

[23] F. Durvaux and F.-X. Standaert, "From improved leakage detection to the detection of points of interests in leakage traces," in *Advances in Cryptology–EUROCRYPT 2016*. Springer, 2016.

[24] R. Poussier, F.-X. Standaert, and V. Grosso, "Simple key enumeration (and rank estimation) using histograms: An integrated approach," in *Cryptographic Hardware and Embedded Systems–CHES*. Springer, 2016.

[25] B. Köpf and D. Basin, "An information-theoretic model for adaptive side-channel attacks," in *Proceedings of the 14th ACM conference on Computer and communications security*, 2007.

[26] R. Wang, H. Wang, and E. Dubrova, "Far field em side-channel attack on aes using deep learning," in *Proceedings of the 4th ACM Workshop on Attacks and Solutions in Hardware Security*, 2020.

[27] R. Wang, H. Wang, E. Dubrova, and M. Brisfors, "Advanced far field em side-channel attack on aes," in *Proceedings of the 7th ACM on Cyber-Physical System Security Workshop*, 2021.