Revisiting a Realistic EM Side-Channel Attack on a Complex Modern SoC

Debao Wang Nanjing University of Science and Technology Nanjing, China wangdebao@njust.edu.cn

Yongbin Zhou Nanjing University of Science and Technology Nanjing, China zhouyongbin@njust.edu.cn

ABSTRACT

Side-channel analysis on complex SoC devices with high-frequency microprocessors and multitasking operating systems presents significant challenges in practice due to the high costs of trace acquisition and analysis, generally involving tens of thousands to millions of traces. This work uses a cryptographic execution process on a Broadcom 2837 SoC as a case study to explore ways to reduce costs in electromagnetic side-channel analysis. In the data acquisition phase, we propose an efficient electromagnetic probe positioning strategy that does not require additional tool assistance, significantly accelerating the collection of effective electromagnetic traces. In the side-channel analysis phase, we investigate the combined use of preprocessing techniques, where the optimal preprocessing approach successfully reduces the number of required electromagnetic traces by 12 times, significantly improving the success rate of attacks. Additionally, we implement profiling attacks on such devices for the first time, including traditional template attacks, MLP-based, and CNN-based side-channel analysis, demonstrating that even minimal modeling costs can yield excellent analysis performance. Our study confirms the feasibility of low-cost side-channel analysis on complex SoCs and indicates that the sensitive applications running on these devices still require protection.

KEYWORDS

Electromagnetic Analysis, System-on-Chip, Correlation Analysis, Profiling Attacks, Probing Techniques

1 INTRODUCTION

Compared with microcontrollers, complex SoC devices equipped with multi-core microprocessors and running modern operating systems demonstrate higher levels of integration and more powerful functionalities, making them the ideal choice for applications in industrial automation, high-end consumer electronics, smart Internet of Things, advanced medical equipment, etc.. However, it is not easy to ensure their security in real-world scenarios due to the threats of side-channel analysis (SCA), which is capable of revealing the sensitive information within devices from power consumption [13], electromagnetic (EM) emissions [1], and timing information during the execution of algorithms [5]. Therefore, it is essential to study side-channel analysis for complex SoCs. Yiwen Gao Nanjing University of Science and Technology Nanjing, China gaoywin@gmail.com

> Xian Huang Open Security Research Shenzhen, China 384811151@qq.com

Recently, the scope of side-channel analysis has expanded beyond extracting cryptographic keys to include tasks such as recovering hyperparameters in neural networks [24], memory data [2], and APP recognition [3]. This paper specifically focuses on EM side channels for cryptographic algorithms.

SoCs equipped with high-frequency processors (600MHz or higher) and running modern operating systems present new challenges for side-channel analysis, with relatively few publications focusing on such devices. In 2015, Longo et al. [15] conducted an EM side-channel analysis against the T-table based AES implementation on the AM335x SoC ARM core, demonstrating the feasibility of side-channel analysis on SoCs and noting that the complexity of operating systems increases the difficulty of analysis. In 2017, Frieslaar et al. [10] claimed to have recovered 12 AES subkeys using only 100 EM traces on a Raspberry Pi 2B v1.2. However, studies indicate that the number of measurements required for a successful EM side-channel analysis on SoCs far exceeds that needed on microcontrollers, typically ranging from tens of thousands to tens of millions. For instance, Haas and Aysu [12] recovered the ARM CE AES key on the Apple A10 Fusion SoC with 5 million to 30 million EM traces; Barenghi et al. [4] recovered the AES key by analyzing 100,000 EM traces on the Cortex-A7 processor, utilizing microarchitectural features of the superscalar CPU. Thus, reducing the substantial attack costs when evaluating the side-channel security of such devices is particularly crucial. Notably, the aforementioned studies all employed non-profiling attacks, such as Correlation Power Analysis (CPA) [6]; however, to the best of our knowledge, the effectiveness of profiling SCA on such devices has not been validated yet.

In this paper, we select the Raspberry Pi 2B v1.2 as the device under test, a widely recognized high-performance embedded device based on the Broadcom 2837 SoC, with a maximum clock frequency of 900MHz, running Raspberry Pi OS. This study comprehensively evaluates the device's resistance to side-channel analysis, covering the entire process from trace acquisition, preprocessing and analysis. We employ both non-profiling and profiling attacks through the EM side channel to recover the AES-128 key. To reduce the cost of attacks, we attempt a combination of various noise reduction and alignment techniques, significantly decreasing the number of traces required for key recovery. In addition, we validate the effectiveness of profiling attacks on such devices. We also find that effective EM trace acquisition is a critical and time-consuming step in the evaluation process, and the importance of this step is often underestimated.

The main contributions of this paper include:

- We explore preprocessing schemes based on time and frequency domain, reducing the number of traces required for key recovery by 12 times, greatly improving the efficiency of side-channel analysis on such devices.
- Using the Broadcom BCM2837 SoC as a case study, we detail the process for side-channel evaluation, thereby providing the side-channel analysis community with a new practical case study. The methods presented are easily adaptable to other similar devices.

The remainder of this paper is organized as follows: Section 2 covers the background knowledge; Section 3 presents the details of EM trace acquisition from SoCs, the preprocessing, and the analysis of traces; Section 4 shows the experimental results and discussions; finally, Section 5 concludes the work.

2 BACKGROUND

2.1 An Overview of Broadcom 2837 SoC



Figure 1: A simplified block diagram of the components of the Broadcom 2837 SoC.

This subsection primarily introduces the components within SoCs that are directly relevant to side-channel analysis. As illustrated in Figure 1, the Broadcom 2837 SoC is equipped with an ARM Cortex-A53 multi-core microprocessor, based on the ARMv8-A 64bit instruction set, and supports operating system execution. This SoC utilizes a dual-issue superscalar architecture and an 8-stage pipeline design, capable of executing two independent instructions simultaneously, offering a higher degree of parallelism compared to microcontrollers, although this high parallelism tends to reduce the signal-to-noise ratio (SNR) [19] of traces. The ARM Cortex-A53 is equipped with 32KB of L1 data and instruction cache and 512KB of L2 cache, with cache hits or misses directly impacting the execution time of cryptographic algorithms and the synchronization of traces. Moreover, the ARM Cortex-A53 supports modern operating systems (e.g., Linux) and may therefore generate additional noise during trace acquisition, such as interruptions by high-priority processes, kernel switches, and the operation of other programs. Similar to microcontrollers, GPIO and UART modules are commonly used to implement the triggering and communication for EM trace acquisition.

In summary, complex SoCs result in the collected traces being out of synchronization and accompanied by more noise, thus requiring more traces to recover secret information.

2.2 Preprocessing

Prior to analysis, employing noise reduction and alignment strategies can reduce the number of traces needed. Specifically, these processes can be conducted from both the frequency domain and the time domain.

2.2.1 Frequency Domain. In the frequency domain, it is possible to implement noise reduction and alignment operations on traces. The objective of noise reduction is to filter out low-frequency or high-frequency noise that is unrelated to the cryptographic algorithm. Initially, the Fast Fourier Transform (FFT) is employed to convert traces from the time domain to the frequency domain, followed by the selection of an appropriate filter type (such as low-pass, band-pass, or high-pass filters) based on the requirements. The choice of filter parameters can be determined by plotting the frequency spectrum of multiple traces on the same plot and observing the signal amplitude. The parameters are then enumerated from high to low frequency. To address the issue of trace desynchronization, a frequency domain-based analysis method is utilized. This involves calculating the Power Spectral Density (PSD) of each trace [20].

2.2.2 Time Domain. For traces misaligned in the time domain, we can first perform denoising in the frequency domain, followed by using alignment strategies. We use elastic alignment [21] to align traces in the time domain, using the Dynamic Time Warping (DTW) algorithm, which measures the similarity between two sequences. In aligning a set of traces, a random trace is typically chosen as a reference (denoted as X), and other traces in the set (such as the trace Y) are aligned to it. The optimal matching path (warp path) determined by DTW, defines the best matching points between the reference trace X and target trace Y. To prevent an increase in the length of the trace after alignment, asymmetric projections (equations 14 and 15 in [21]) can be adopted, meaning the target trace Y is adjusted to match the reference trace X. In practice, the FastDTW algorithm is used instead of the DTW algorithm to reduce computational complexity and improve efficiency, while maintaining accuracy similar to that of DTW. Additionally, the setting of the radius parameter in the FastDTW, which controls the size of the local search range for the warp path at each scale level, must be considered. The radius defines the maximum distance at which a point in one sequence can be aligned with a point in another sequence during alignment. A smaller radius value leads to faster computation speeds but may decrease the accuracy of alignment; conversely, a larger radius value can enhance accuracy but will increase computational costs.

2.3 Non-profiling and Profiling Attacks

Side-channel analysis can be categorized into two main types: nonprofiling attacks and profiling attacks. Non-profiling attacks, such as Differential Power Analysis (DPA) [13], Correlation Power Analysis, and Mutual Information Analysis (MIA) [11], do not require a prior detailed understanding of the specific leakage and noise models of the target device when conducting the attack. In contrast, profiling attacks, such as template attacks (TA) [9], stochastic attacks [18], and machine-learning based attacks [14], require the attacker to obtain a replica of the target device in order to provide a detailed characterization of the device's leakage characteristics and noise environment. For instance, template attacks typically utilize multivariate Gaussian models to construct templates, which are then used to facilitate subsequent attacks.

Recently, Deep Learning-Based Side-Channel Analysis (DLSCA) [17] has attracted significant attention and is considered as one of the most powerful profiling attacks. DLSCA is capable of performing attacks directly on the original measurements, effectively addressing challenges associated with trace misalignment, such as random delays and clock jitter, while also enabling the evaluation of cryptographic devices under worst-case scenarios. Among the prevalent methods in DLSCA, models based on Multi-Layer Perceptrons (MLP) [22] and Convolutional Neural Networks (CNN) [8] are particularly common.In this work, we explore the effectiveness of these two models on complex SoCs.

3 OUR METHODS

This section presents the side-channel evaluation of the Raspberry Pi 2B. Initially, in Section 3.1, we discuss the experimental setup for trace acquisition and the techniques for efficient EM probe positioning. Subsequently, in Section 3.2, we apply noise reduction and alignment techniques to preprocess the traces. Finally, in Section 3.3, we employ three profiling attacks—TA, MLP, and CNN—to explore their effectiveness on complex SoC devices.

3.1 Acquisition

The Raspberry Pi 2B utilizes an advanced manufacturing process with very compact dimensions, which makes it very challenging to use jumpers to capture power consumption directly from the core power area of the chip. An intuitive alternative is to monitor the overall power consumption of the SoC, which is typically achieved by inserting a small resistor into the power line circuit and observing the voltage changes across it. However, the power consumption variations generated by a single AES encryption may be minor compared to the SoC's overall power consumption, resulting in a lower SNR, which makes detection and analysis difficult. Given this, we recommend the use of a magnetic field probe for localized and precise acquisition of traces during the execution of encryption algorithms, as this method proves to be more accurate and effective.

For the software implementation of cryptographic algorithms, it is generally recommended to set the oscilloscope's sampling rate to at least four times the target device's clock frequency to more accurately capture the actual leakage. However, the clock frequency of the Raspberry Pi 2B stably operates at 600MHz in most cases. We conduct the experiment using the maximum supported sampling rate of 1.25G/s with the PicoScope 6403E and observe that even when the sampling rate is reduced to 625M/s, it is still possible to successfully recover all AES subkeys, although this means that more traces need to be collected. We try all probes from Langer's RF1 set, RF2 set, and RF3 mini set. Based on the shape and amplitude of the collected traces, the *RF-U 2.5-2* near-field probe is proven to be the best choice, which is also consistent with experimental results. Compared to the circular probes used in the existing literature [4, 7, 10], we find that these circular probes are unable to capture effective leakage for the Raspberry Pi 2B.

The EM probe positioning is critical for trace acquisition, as different positions capture varying leakage information and affect the performance of the attack. For microcontrollers without operating systems and operating at lower frequencies, probes are typically placed in the core power supply area of the chip or on the surface of the chip packaging. Typically, a characteristic EM waveform generated by the 10-round operation of the AES-128 can be observed at multiple locations, often implying that leakage exists at those locations. However, identifying AES waveforms on a Raspberry Pi 2B, becomes more challenging. The resistors and capacitors around the chip packaging can generate significant electromagnetic interference, such as radiated and conducted emissions [16], obscuring effective leakage. A commonly used method is to use a three-axis platform to automatically scan the chip surface and locate the probe position based on the amplitude and waveform of the captured electromagnetic traces. This approach suits probes detecting vertical directions, which are simply positioned perpendicular to the chip packaging but is not applicable to all probe types. Our approach employs both high-power consumption programs



Figure 2: An EM trace showing one RSA-1024 algorithm and two usleep functions.

and low-power consumption programs to determine effective probe positions. The underlying principle is that the EM traces generated by high-power consumption programs and those generated by lowpower consumption programs will have significant differences in waveform amplitude. Using this method, the chip surface can be manually scanned without the need for an expensive three-axis platform, quickly locating the areas with leakage. Upon identifying a potential probe position, correlation-based known-key analysis can be used to ascertain the presence of leakage. We employ the RSA algorithm (as a high-power consumption program) and the usleep function from the C standard library to locate effective probe positions. As illustrated in Figure 2, despite considerable noise, we still detect distinct waveform differences produced by the aforementioned programs at certain locations. Subsequently, we collect traces of the AES at these positions, where clear AES waveforms are observable (see Figure 4). In our experiments, the optimal position is shown in Figure 3.

In terms of communication , we adopt an asynchronous serial communication protocol (UART) to facilitate communication between the PC and Raspberry Pi 2B. Memory-mapped GPIO is employed, further reducing trigger delays and enhancing the quality of the traces.



Figure 3: The optimal probe position in our experiment.

3.2 Preprocessing and Analysis

3.2.1 Removing Outlier Traces. As shown in Figure 4, the collected traces contain apparent random noise waveforms that need to be eliminated to improve the quality of the dataset. The voltage range of the sampling points in the traces is set from -32,512 to +32,512, represented by a 16-bit integer. Since the voltage values of the noise waveforms exceed the sampling range, they are truncated to \pm 32,512. By counting the occurrences of voltage values at \pm 32,512 across multiple traces and setting a threshold, noise traces exceeding this threshold are filtered out, which results in a loss of about 5% of the traces in our experiments.



Figure 4: An EM trace of AES-128 with noise.

3.2.2 Correlation-based known-key analysis. Known-key analysis is employed to determine the presence of leakage in the collected traces and to identify the optimal intermediate value selections

within the AES algorithm. The effective intermediate values in side-channel analysis include: (1) the input of the first-round Sub-Bytes; (2) the *output* of the first-round SubBytes; (3) the *input* of the last-round SubBytes; (4) the input of the last-round AddRoundKey. By mapping the algorithm's intermediate values under the correct key to the Hamming-weight model or the value model, we obtain hypothetical leakages and compute the correlation coefficients between these and the actual leakages. As illustrated in Figure 5, the correlation coefficients at leakage points are significantly higher than those at non-leakage points, indicating that the optimal intermediate value position is the input of the last-round SubBytes, where a distinct peak in the correlation coefficients is observed. Although the input of the last-round AddRoundKey also exhibits a discernible peak, its leakage level is lower than that of the input of the last-round SubBytes, consistent with the theory that the nonlinearity of the S-box enhances the level of leakage. Additionally, we observe that the intermediate values from the first round of AES do not exhibit significant leakage, a finding corroborated by practical attacks. The Known-key analysis also facilitates dimensionality reduction in the traces, thereby enhancing analytical efficiency. For instance, we can opt to analyze only the sampling points around the peaks of correlation.



Figure 5: Results of correlation-based known-key analysis under four different intermediate values.

3.2.3 Noise Reduction and Alignment. First, we employ digital filtering techniques in the frequency domain, such as low-pass and band-pass filters, to denoise the traces. Subsequently, we utilize two mainstream alignment techniques: frequency domain-based alignment and elastic alignment. These methods demonstrate a higher degree of automation compared to traditional alignment techniques like peak alignment and pattern matching, and so on. We first denoise the traces, followed by alignment, given that alignment techniques are generally more effective in a low-noise environment. For instance, high-noise components in the time domain obscure actual leakage, thereby diminishing the efficacy of the alignment methods. In frequency domain-based alignment, we calculate the PSD of the traces using the periodogram method. This involves performing a discrete Fourier transform on the signal sequence, squaring its amplitude spectrum, and dividing by the sequence length N. For elastic alignment, we utilize the FastDTW algorithm, setting the radius parameter to 125 to balance performance and alignment accuracy.

3.2.4 CPA Attack. CPA is one of the most powerful non-profiled side-channel analysis methods, which describes the linear relationship between hypothetical leakage and actual leakage by calculating the correlation coefficient. At the trace samples corresponding to the calculation of intermediate values, the hypothetical leakage corresponding to the correct key guess exhibits a higher correlation compared to that of the incorrect key guesses. Typically, by calculating the correlation coefficient between the hypothetical leakage and actual leakage and constructing a correlation matrix, the guessed key is determined, ultimately recovering all AES subkeys using a divide-and-conquer strategy. We select the guessed key corresponding to the maximum absolute value in the correlation matrix as the correct key. However, we prefer the Hamming-weight model, as it usually performs more accurately than the value model in software implementations of cryptographic algorithms. According to the known-key analysis, the input of the last-round SubBytes represents the optimal intermediate value for conducting attacks.

3.3 Profiling Attacks

3.3.1 Template Attack. TA is a robust profiled method that utilizes multivariate normal distributions to characterize actual leakages. Unlike non-profiled methods, template attacks are typically divided into two phases: initially, the construction of templates, which includes the computation of mean vectors and covariance matrices; subsequently, these templates are employed to facilitate the attack. Due to the quadratic growth of the covariance matrix dimensions with the number of samples in the trace, the selection of points of interest (POIs) is essential to reduce computational complexity [16]. Common techniques such as SNR, correlation, and Principal Component Analysis (PCA) are employed to identify POIs. In this study, we apply Analysis of Variance (ANOVA) [23] as the distinguisher because it can effectively handle different data partitions in a way that is highly consistent with the mechanism of template attacks. To balance attack performance and modeling costs, we chose to construct nine templates using the Hamming-weight model instead of 256 templates based on 8-bit intermediate values.

3.3.2 MLP and CNN Based Attacks. In DLSCA, key recovery is considered a classification problem. Hence, neural network models are employed to learn the relationship between the actual leakages and the hypothetical leakages, mapping the actual leakages to N probability categories associated with the intermediate values. Similar to TA, DLSCA consists of two phases: a training phase and an attack phase. During the training phase, we apply correlation-based feature selection to a dataset, using the extracted features as inputs to the model to enhance both training efficiency and attack performance. The model's labels are hypothetical leakages, with the labels encoded using *One-Hot Encoding*. The hypothetical leakage of AES that we use is defined by Equation 1.

$$F(C_i, K_i) = INV_SBOX(C_i + K_i)$$
(1)

Table 1: Datasets under different preprocessing methods.

Datasets	Preprocessing Methods	Number of Traces
RawSet	none	5,000,000
CleanSet	removing outlier traces	4,760,256
PSDSet	PSD only	4,760,256
AlignSet	elastic alignment only	4,760,256
FilterSet	band-pass filter only	4,760,256
FiltPSD	band-pass filter + PSD	4,760,256
FiltAlign	band-pass filter + elastic alignment	4,760,256

where K_i is the *i*-th byte of the last-round key, C_i is the *i*-th byte of the ciphertext, and *INV_SBOX* is the inverse S-BOX of AES.

In practical attacks, the key difference between MLP-based and CNN-based side-channel analysis lies in the neural network models used, although their predicted labels and model inputs are the same. We employ the same model architecture for all 16 bytes of the AES key. The structures of the MLP and CNN models are shown in Figures 6 and 7, respectively. Both models use the following hyperparameters.

- Optimizer: Adam with a learning rate of 0.001,
- Loss function: Cross-entropy loss,
- Labels encoded using One-Hot Encoding,
- Batch size: 512,
- Epochs: 100,
- Early stopping strategy with a patience of 30, only saving the best model,
- Activation function: *relu*,
- Weight Initialization: *He normal initializer* for MLP; *Glorot uniform initializer* for CNN.



Figure 6: MLP model architecture. The model input is the trace after feature selection. The output layer consists of 256 neurons, corresponding to the 256 possible values of the hypothetical leakage. A Dropout layer is added after each fully connected layer to improve the model's generalization capability.

4 EXPERIMENTAL RESULTS

In the following sections, we first use CPA to evaluate the effectiveness of various preprocessing schemes, then validate the feasibility of profiling attacks on the Raspberry Pi 2B, and finally compare the performance of the attacks between deep learning-based sidechannel analysis (e.g., MLP and CNN) and template attacks.



Figure 7: CNN model architecture.The Conv1D layer has 4 filters with a kernel size of 1, and the padding is set to "same". As for the AveragePooling1D layer, it has a pool size of 2 with a stride of 2.

4.1 Setups

To ensure the reproducibility of this paper, this section introduces the software and hardware platforms used in the experiments.

Target.

- Target Board: Raspberry Pi 2 Model B v1.2 (Broadcom 2837 SoC),
- CPU: a quad-core ARM Cortex A53 (ARMv8),
- Clock Frequency: fixed at 600 MHz,
- Memory: 1GB SDRAM operates at 450MHz,
- OS: Raspberry Pi OS Lite.

Acquisition and measurement equipments.

- PicoScope 6403E Oscilloscope, 1.25GHz sampling rate,
- A Langer RF-U 2,5-2 near-field probe,
- A Langer PA303 pre-amplifier,
- Trigger: Memory-mapped GPIO,
- Communication: UART.

4.2 Comparison of the preprocessing methods

By combining different denoising and alignment methods, we create seven datasets (see Table 1). Notably, preprocessing for the third through seventh datasets occurs after the removal of outliers. Based on the results of known-key analysis, we insert a trigger signal between the AddRoundKey operations of the penultimate and final round of AES to facilitate trace acquisition. In total, we collect 5 million traces, retaining 4,760,256 effective traces after removing outliers. We evaluate the effectiveness of various preprocessing schemes using CPA. As illustrated in Figure 8, the preprocessing scheme combining band-pass filtering with elastic alignment (FiltAlign) achieves a 100% success rate using 80,000 traces for attacks, reducing the number of required traces by approximately 12 times compared to the unprocessed dataset (RawSet). Digital filtering reduces noise, and when combined with elastic alignment, significantly improves the success rate of attacks. However, direct alignment of traces without prior filtering decreases trace quality, because the elastic alignment depends on matching the voltage

distances between traces, and high noise environments exacerbate voltage fluctuations, reducing the SNR and disrupting the accuracy of leakage point matching. We also find that frequency-domain based CPA performs poorly with or without filtering, suggesting that the method is extremely sensitive to noise.



Figure 8: Success rate of AES with different preprocessing methods.

4.3 Performance of the Template Attack

In this section, we evaluate the efficacy of TA under FiltAlign dataset. Using the ANOVA distinguisher, we select the top 30 POIs based on scores, which sufficiently encompass the primary leakage information in the dataset. To evaluate the impact of using various numbers of traces used for modeling (ranging from 150,000 to 2,000,000), we employ guessing entropy as the metric and conduct 50 experimental trials on a random subset of the attack set, consisting of 1,000,000 traces. As illustrated in Figure 9, modeling with 150,000 traces and executing the attack with 10,000 traces yields a success rate of 100%. Compared to the CPA, TA achieves high attack performance using fewer traces to model. The number of attack traces used for successful key recovery is one-eighth that of CPA, although there are some differences in the difficulty of attacking different bytes (e.g., bytes 3 and 14 are more difficult). In addition, we note that the attack performance does not improve significantly when the number of traces used for modeling is increased from 500,000 to 2,000,000, indicating that the attack performance saturates as the number of traces used for modeling increases.

4.4 TA vs DLSCA

To compare the performance between TA and DLSCA, we select the *FiltAlign* dataset, with 500,000 traces used for training and 10,000 traces for attacking. For TA, templates are created using both the Hamming-weight model (TA-HW) and the intermediate values (TA-ID), and 30 POIs are selected using the ANOVA distinguisher. For DLSCA, we apply correlation-based feature selection, extracting 30 POIs as the model input. We train the same model architecture for the 16 last-round subkeys of AES (see Section 3.3.2), resulting in 16 MLP and CNN models. We continue to use guessing entropy as the metric to measure the performance of the three attack methods. As shown in Figure 10, the performance difference between TA-ID,



Figure 9: Guess entropy of the last-round key of AES under six different numbers of EM traces for modeling.

MLP, and CNN is minor. When using approximately 1,500 traces for attacking, the guessing entropy of the 16 last-round subkeys of AES reaches 0, which is better than TA-HW (which requires around 5,600 traces).



Figure 10: Comparison of guessing entropy between TA, MLP, and CNN on the FiltAlign dataset.

4.5 Discussion

Sampling Rate Setting. Our experiments show that using a sampling rate lower than Nyquist frequency can still be successful when performing side channel analysis on complex SoCs. This may be due to the fact that in the software implementation of AES, the execution of a program statement requires multiple processor clock cycles to complete. Therefore, even if the sampling rate is below the Nyquist frequency, it is sufficient to capture the electromagnetic leakage of the operation. Since the operation continues over multiple clock cycles, the leakage does not change rapidly over multiple clock cycles. Theoretically, the higher the sampling rate used, the better the analysis. We therefore recommend that the oscilloscope sample rate be set to more than four times the target clock frequency when performing side channel analysis on such devices.

Electromagnetic Probe Selection. We use the Langer RF series near-field probes and find that probes of different shapes exhibit varying attack performance. In highly integrated SoCs with dense circuit components, smaller EM probes provide higher spatial resolution and lower sensitivity, allowing us to more accurately localize specific circuit components, thereby reducing noise from nearby components.

DLSCA. In our DLSCA experiments, we find that without feature selection and directly analyzing the raw traces, only partial subkeys from the last round could be recovered. However, using the same dataset, TA with feature selection can successfully achieve key recovery. Therefore, we believe that feature selection is a crucial step in DLSCA on complex SoCs, as it can significantly improve attack performance. While it is feasible to directly recover keys from raw traces in deep learning-based side-channel analysis on microcontrollers, this effectiveness is hard to replicate on complex SoCs. In most cases, even if the model converges, the accuracy remains a random prediction probability value. Therefore, this situation implies that it is more difficult to perform side channel analysis on complex SoCs.

5 CONCLUSION

In this paper, we detail the entire process of side-channel analysis on complex SoC devices. To address the time-consuming issue of probe positioning during EM trace acquisition for such devices, we propose using high and low power consumption programs to identify leakage points and determine the presence of leakage based on trace characteristics. Given the high noise and alignment issues of such devices, we employ various preprocessing techniques, with the most effective method reducing the required number of attack traces by approximately 12 times. This indicates that trace preprocessing is necessary when conducting side-channel analysis on these devices. Furthermore, we implement profiling attacks on these devices. The experimental results show that on the FiltAlign dataset, template attacks based on intermediate value modeling, MLP-based, and CNN-based side-channel analysis have comparable attack performance. Future research directions include exploring cross-device performance of deep learning-based side-channel analysis on such devices.

REFERENCES

- Dakshi Agrawal, Bruce Archambeault, Josyula R Rao, and Pankaj Rohatgi. 2003. The EM side—channel (s). In Cryptographic Hardware and Embedded Systems-CHES 2002: 4th International Workshop Redwood Shores, CA, USA, August 13–15, 2002 Revised Papers 4. Springer, 29–45.
- [2] Can Aknesil and Elena Dubrova. 2022. Towards Generic Power/EM Side-Channel Attacks: Memory Leakage on General-Purpose Computers. In 2022 IFIP/IEEE 30th International Conference on Very Large Scale Integration (VLSI-SoC). IEEE, 1–6.
- [3] A Amodei, D Capriglione, L Ferrigno, G Miele, L Tari, G Tomasso, and G Cerro. 2023. Experimental Analysis of Side-Channel Emissions for IoT Devices Activities'

Profiling. In 2023 IEEE International Workshop on Metrology for Industry 4.0 & IoT (MetroInd4. 0&IoT). IEEE, 42–47.

- [4] Alessandro Barenghi and Gerardo Pelosi. 2018. Side-channel security of superscalar CPUs: Evaluating the impact of micro-architectural features. In Proceedings of the 55th Annual Design Automation Conference. 1–6.
- [5] Daniel J Bernstein. 2005. Cache-timing attacks on AES. (2005).
- [6] Eric Brier, Christophe Clavier, and Francis Olivier. 2004. Correlation power analysis with a leakage model. In Cryptographic Hardware and Embedded Systems-CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings 6. Springer, 16–29.
- [7] Sebanjila Kevin Bukasa, Ronan Lashermes, Hélène Le Bouder, Jean-Louis Lanet, and Axel Legay. 2018. How TrustZone could be bypassed: Side-channel attacks on a modern system-on-chip. In Information Security Theory and Practice: 11th IFIP WG 11.2 International Conference, WISTP 2017, Heraklion, Crete, Greece, September 28–29, 2017, Proceedings 11. Springer, 93–109.
- [8] Eleonora Cagli, Cécile Dumas, and Emmanuel Prouff. 2017. Convolutional neural networks with data augmentation against jitter-based countermeasures: Profiling attacks without pre-processing. In Cryptographic Hardware and Embedded Systems-CHES 2017: 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings. Springer, 45–68.
- [9] Suresh Chari, Josyula R Rao, and Pankaj Rohatgi. 2003. Template attacks. In Cryptographic hardware and embedded systems-CHES 2002: 4th International Workshop Redwood Shores, CA, USA, August 13–15, 2002 Revised Papers 4. Springer, 13–28.
- [10] Ibraheem Frieslaar and Barry Irwin. 2017. Recovering AES-128 encryption keys from a Raspberry Pi. In Southern Africa Telecommunication Networks and Applications Conference (SATNAC). 228–235.
- [11] Benedikt Gierlichs, Lejla Batina, Pim Tuyls, and Bart Preneel. 2008. Mutual information analysis: A generic side-channel distinguisher. In International Workshop on Cryptographic Hardware and Embedded Systems. Springer, 426–442.
- [12] Gregor Haas and Aydin Aysu. 2022. Apple vs. EMA: electromagnetic side channel attacks on apple CoreCrypto. In Proceedings of the 59th ACM/IEEE Design Automation Conference. 247–252.
- [13] Paul Kocher, Joshua Jaffe, and Benjamin Jun. 1999. Differential power analysis. In Advances in Cryptology—CRYPTO'99: 19th Annual International Cryptology Conference Santa Barbara, California, USA, August 15–19, 1999 Proceedings 19. Springer, 388–397.
- [14] Liran Lerman, Gianluca Bontempi, and Olivier Markowitch. 2015. A machine learning approach against a masked AES: Reaching the limit of side-channel

attacks with a learning model. *Journal of Cryptographic Engineering* 5 (2015), 123–139.

- [15] Jake Longo, Elke De Mulder, Dan Page, and Michael Tunstall. 2015. SoC it to EM: electromagnetic side-channel attacks on a complex system-on-chip. In Cryptographic Hardware and Embedded Systems-CHES 2015: 17th International Workshop, Saint-Malo, France, September 13-16, 2015, Proceedings 17. Springer, 620–640.
- [16] Stefan Mangard, Elisabeth Oswald, and Thomas Popp. 2008. Power analysis attacks: Revealing the secrets of smart cards. Vol. 31. Springer Science & Business Media.
- [17] Loïc Masure, Cécile Dumas, and Emmanuel Prouff. 2020. A comprehensive study of deep learning for side-channel analysis. *IACR Transactions on Cryptographic Hardware and Embedded Systems* (2020), 348–375.
- [18] Werner Schindler, Kerstin Lemke, and Christof Paar. 2005. A stochastic model for differential side channel cryptanalysis. In Cryptographic Hardware and Embedded Systems-CHES 2005: 7th International Workshop, Edinburgh, UK, August 29–September 1, 2005. Proceedings 7. Springer, 30–46.
- [19] François-Xavier Standaert, Tal G Malkin, and Moti Yung. 2009. A unified framework for the analysis of side-channel key recovery attacks. In Advances in Cryptology-EUROCRYPT 2009: 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, April 26-30, 2009. Proceedings 28. Springer, 443–461.
- [20] Chin Chi Tiu. 2005. A new frequency-based side channel attack for embedded systems. Ph. D. Dissertation. Citeseer.
- [21] Jasper GJ Van Woudenberg, Marc F Witteman, and Bram Bakker. 2011. Improving differential power analysis by elastic alignment. In *Topics in Cryptology–CT-RSA* 2011: The Cryptographers' Track at the RSA Conference 2011, San Francisco, CA, USA, February 14-18, 2011. Proceedings. Springer, 104–119.
- [22] Leo Weissbart. 2020. Performance analysis of multilayer perceptron in profiling side-channel analysis. In Applied Cryptography and Network Security Workshops: ACNS 2020 Satellite Workshops, AIBlock, AIHWS, AIOTS, Cloud S&P, SCI, SecMT, and SiMLA, Rome, Italy, October 19–22, 2020, Proceedings 18. Springer, 198–216.
 [23] Wei Yang and Anni Jia. 2021. Side-channel leakage detection with one-way
- [23] Wei Yang and Anni Jia. 2021. Side-channel leakage detection with one-way analysis of variance. Security and Communication Networks 2021 (2021), 1–13.
- [24] Xiang Zhang, Aidong Adam Ding, and Yunsi Fei. 2023. Deep-Learning Model Extraction Through Software-Based Power Side-Channel. In 2023 IEEE/ACM International Conference on Computer Aided Design (ICCAD). IEEE, 1–9.