Power Analysis Side Channel Attacks and Countermeasures for the Internet of Things

Christopher Liptak, Sanchita Mal-Sarkar and Sathish A.P. Kumar
{c.l.liptak78, s.malsarkar, s.kumar13}@csuohio.edu
Department of Electrical Engineering and Computer Science, Cleveland State University, Cleveland, OH, USA

2022 IEEE Physical Assurance and Inspection of Electronics (PAINE) | 979-8-3503-9909-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/PAINE56030.2022.10014854

***Abstract*—With the explosion in the number of internet of things (IoT) devices in recent years, the security of these devices has become an area of major concerns across the globe. Encryption is an essential means of protecting data in IoT devices, but cryptographic implementations are vulnerable to attacks as well. These vulnerabilities can allow attackers to completely bypass or significantly weaken the theoretical strength of the encryption algorithms. Resistance to cryptanalysis alone is not adequate for building a secure cryptosystem in practice. Power analysis side-channel attacks have emerged as a very effective method of discovering cryptographic keys by monitoring the power consumption of devices. Power consumption of the chips can reveal information about the cryptographic operations being performed or the data being processed. Numerous countermeasures against power analysis attacks have been discussed in the literature. However, all these countermeasures introduce various tradeoffs, such as increased power consumption, decreased performance, and increased space requirements for additional hardware. In addition to strong resistance to power analysis attacks and simplicity of implementation, developers need to evaluate the countermeasures by taking into consideration these tradeoffs while designing and implementing the IoT, which are often operated in heterogenous and highly resource constraint environments. This paper explores the state-of-the art power analysis attacks and their countermeasures and analyze the limitations of these countermeasures based on the constraints present in IoT devices.**

***Key Words* - Side channel attacks; Internet of Things; Cryptography and Encryption; Power Analysis.**

1. Introduction

The IoT is a network of physical objects or things, often embedded with software, sensors, actuators, and network connectivity, for connecting and exchanging information with other devices over the Internet. While IoT brings several benefits to manufacturing, automotive, transportation, retails, healthcare, and so on, it also opens several opportunities for adversaries to hack into systems and leak sensitive information by covert channels. These IoT devices include smart cars, smart thermostats, smart watches, health implants, sport wearables, smart toys, and so on. Connected smart cars are complex systems composed of several units that exchange large amount of data. Hackers can manipulate these systems and gain control of smart driverless cars. Adversaries can hack into the sensors that control the temperatures in a power plant. Hackers can access smart watches through Bluetooth or smart phone and track children's locations and their activities. With the increasing prevalence of IoT devices in our everyday life, it is becoming critical to secure those devices than ever before [33-38].

Establishing security at all layers will require a holistic and comprehensive approach. Only securing one layer is insufficient, as other compromised layers can lead to many compromised devices and serious real-world consequences. IoT devices are connected to the cloud-based services through the Internet directly or through smart phone app. IoT devices generally have limited storage space and low power consumption processors. If the firmware of the IoT devices are tempered, the data collected from the cloud will be unreliable. Even when a cryptographic algorithm is implemented on the devices and an encrypted link between IoT devices and the cloud is established, critical information can be leaked during its execution because of the optimization in the algorithm implementation. Information side channel analysis in the form of power consumption, electromagnetic radiation, timing analysis, or fault-injection can reveal enough information to allow for the recovery of the encryption keys. The analysis and exploitation of this information side-channels is referred as side-channel attacks.

These side channel attacks are possible despite the theoretical strength of the cryptographic solutions. Traditional cryptanalysis assumes that cryptographic algorithms are purely mathematical (black box) and internal components cannot be exposed or changed by the adversaries. However, side channel analysis attacks exploit the implementation of the cryptographic algorithms and secretly recover information about the operations being performed and data being processed and reveal the secret keys.

Since the side channel attacks can be non-invasive and passive, and the IoT generally operates in inexpensive and resource constrained environments, they pose a serious threat to the security of the embedded systems, such as smart cards, microcontrollers, and RFID, which are the major components of IoT applications. The proliferation of commercially available inexpensive tools used for side-channel attacks in recent years make IoT more vulnerable.

Thus, enabling the security of IoT devices in several levels is much more critical than ever.

Side-channel attacks have been successfully utilized to break the hardware or software implementation of all modern cryptographic primitives such as AES, RSA, ECC, and HMAC for extracting the keys from the devices. SCA can reveal the key from a device in minutes or days which would take decades by employing cryptanalysis alone [24]. It is important to develop countermeasures against side-channel analysis attacks which will provide similar level of security that modern cryptographic primitives offer against conventional cryptanalysis.

Power side-channel attack is one of the most important side-channel attacks. The power consumption of a device depends on the dynamic power and the leakage power. Dynamic power is caused by switching activities of the transistors from logic 1 to logic 0 or vice versa. Attackers prefer to capture the dynamic power signals since they reflect the functional behavior of the device over leakage power signals, which represent the leakage current generated during the off state. By measuring the power, the adversary can get information about the operations being performed or data being processed.

Compared to existing work [32], this paper contributes to the literature by (a) analyzing side channel attacks and evaluating different types of power analysis attacks; (b) developing taxonomy of side channel attacks in IoT (c) identifying current power analysis (PA) countermeasures and evaluate them based on their effect on the following criteria: performance, power consumption, space requirements for additional hardware, and their resistance to PA attacks; and (d) developing the classification of PSA countermeasures.

The remainder of this paper is organized as follows: Section II briefly discusses the taxonomy of side channel analysis attacks in IoT and explains power analysis side-channel attacks. Section III introduces various power analysis counter measures. Section IV evaluates the countermeasures with respect to the constraints present in IoT devices and their resistance to PA attacks. Finally, Section V presents the concluding remarks.

2. Side Channel Analysis Attacks

Side-channel attacks, first introduced by Kocher (1996), exploit the implementations of the cryptographic algorithms [23, 26]. Side channel analysis attacks (SCA) can be classified into invasive and non-invasive attacks, depending on the access level of the adversaries to the devices prior to the side channel attacks. Invasive attacks require direct access to the internal components of the device. An example of an invasive attack includes placing a wire on a data bus to see the data transfer. On the other hand, noninvasive attacks completely rely on externally available information, such as electromagnetic radiation, execution time and power consumption [26].

Side channel analysis attacks can also be classified into active and passive attacks, depending on the tampering with the proper functioning of the devices under attacks. Active SCA includes fault injection attacks in which the adversary can control how the devices operate and reveal the side-channel information to break the cryptographic module and extract the key [22]. In passive SCA, the attacker does not require control of the devices under attacks and the system works normally. The systems under passive attacks will leak sensitive information using the side channels while performing normal operations. Several passive SCA including power analysis attacks, timing analysis attacks, and electromagnetic (EM) analysis attacks were discussed in [2]. Both types of attacks are easy to mount on IoT and often they do not require high-end equipment.

The taxonomy of side-channel attacks in IoT is shown in fig 1. Depending on the source of information leakage, side-channel attacks can be classified as power analysis attacks, timing analysis attacks, electromagnetic analysis attacks, and fault injection attacks. They can be further classified depending on specific attack methods, signal generation methods, and analysis granularity [21]

Comparison of side-channels and an extensive literature review on electromagnetic side-channel attacks are provided in [28]. The use of faults to reveal the cryptographic key was first presented in 1996 by Boneh et al. [29][30] and has received significant attention. Our paper focuses on power analysis attacks, which is one of the most important attacks in IoT ecosystem.

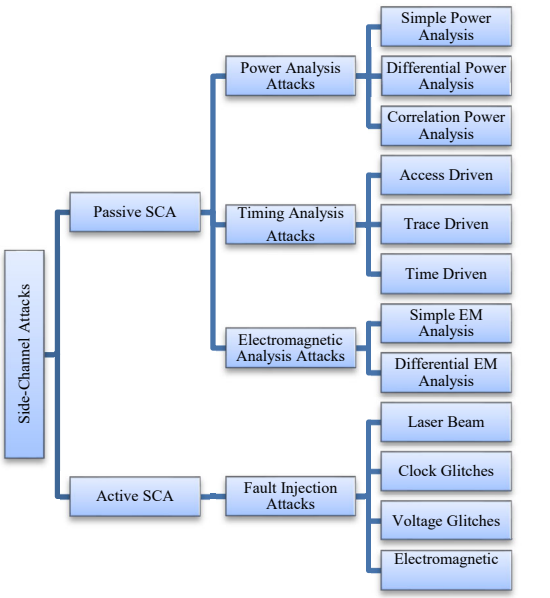

Fig 1: Taxonomy of side-channel attacks in IoT

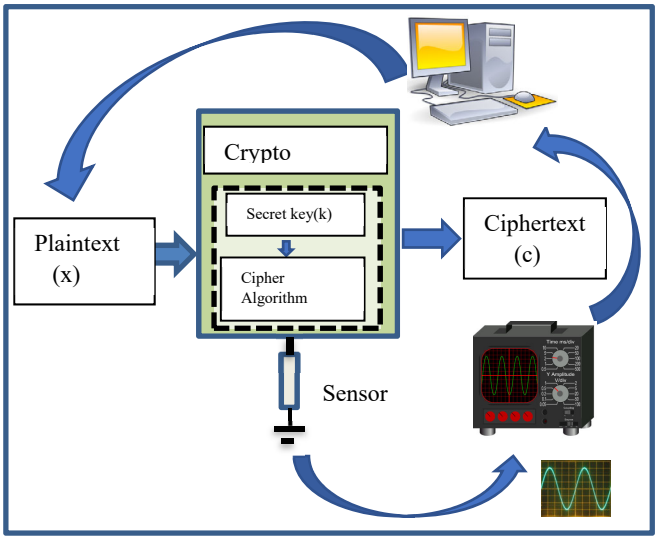

Fig 2: Power analysis attacks on crypto hardware (e.g. Smart card)

2.1 Power Analysis Attacks

PA attacks reflect the power consumption of a device due to the underlying cryptographic operations being performed and data being processed. For example, in asymmetric algorithms, such as RSA, the underlying cryptographic operation of interest is modular exponentiation [1]. Modular exponentiation is implemented in practice by performing square and multiply operations. For each bit of the exponent, a squaring operation is performed, while an additional multiplication operation is performed whenever the exponent bit is equal to 1. This allows for recovery of the bits of the exponent, since the squaring and multiplication operations have different power signatures and can be distinguished from one another in a power trace. Once the exponent has been recovered, the private key can then be derived from the exponent. Fig. 2 shows the typical power analysis attacks on a crypto hardware (e.g. smart card).

Power analysis attack is proven to be very effective in mounting attacks and recovering the secret keys from smart cards and other embedded systems. Important power side-channel analysis attacks include simple power analysis (SPA), differential power analysis (DPA), and correlation power analysis (CPA). SPA and DPA attacks were introduced in 1999 by Kocher at al. [2]. They mounted a power analysis attack on DES implementation in hardware to extract the secret key by analyzing the traces of power consumptions during the execution of the cryptographic algorithm. Extensive research has been published on DPA in [8]. The attacks have been mounted on several devices and platforms including FPGA, ASICs, and software. Coron [25] has mounted these attacks on elliptic curve cryptography (ECC) and proposed the SPA-resistant method for point multiplication and the DPA-resistant method for randomizing protective coordinates without compromising efficiency significantly.

SPA is a visual analysis of the power traces obtained while the device is in operation. This method is targeted to work with the devices with limited accessibility where a single or a few power traces are available. It does not require any advanced or statistical analysis to extract the key or sensitive information. In SPA, the goal is to guess which instruction is being executed at an instant of time and the input and output values by employing the power traces. Thus, the attackers need to have prior knowledge of the hardware implementation to mount the attacks. It exploits the relationship between instruction being executed and the side-channel leakage. This power analysis can identify the type of algorithm being executed by showing the sequence of patterns corresponding to squaring and multiplication operation, and the number of rounds of the block ciphers.

DPA has emerged as an extremely effective means of attack and has been successfully implemented against both symmetric and asymmetric cryptographic algorithms. DPA relies on the acquisition of numerous power traces, with each acquisition using a different known plain text input. The analysis then finds a correlation between the power consumption and specific cryptographic operations that are dependent on the bits of the cryptographic keys [2], [3].

The initial step in a DPA attack is to acquire numerous power traces, while the target device performs cryptographic operations. The power traces are typically acquired using an oscilloscope. The oscilloscope records the voltage drop across a resistor that has been added to the power supply line leading to the device [2]. Similar attacks have also been developed, which rely instead on using the EM radiation emitted by the device, since the power consumption and EM emanations are directly related to one another. In contrast, these signals can be acquired non-invasively using a software-defined radio or an oscilloscope and EM probe placed in close proximity to the target device [4], [2].

DPA is one of the most powerful side-channel attacks since it does not require the adversary to have prior knowledge about the device hardware architecture. Another advantage of DPA is it can obtain high quality signal even in a noisy environment. Larger number of power traces make DPA more powerful to extract secret keys and sensitive information from the cryptographic systems by exploiting the data dependency of the power consumption. Power consumption at different instances of time for the same operation depends on the data being processed. As the correlation of the data and the side-channel leakage is usually very small, statistical methods are used to extract possible secret keys [2][21].

CPA attack was proposed by Briar et al in 2004 [27]. CPA is an advanced form of side-channel attacks that

exploits the correlation between the power consumption of the cryptographic devices and the Hamming distance or Hamming weight of the target function, such as the output of the SBOX operation. CPA is the most effective attack in white-box analysis where device leakage is known. It can also be used for black-box analysis if there exists some correlation between the actual leakage of the device and the leakage model being used for CPA [24]. Despite its efficiency and robustness, CPA can be more demanding and difficult to implement. [27]. Table 1 compares different PA attacks that can be mounted on IoT devices or systems.

Comparison Metric	SPA	DPA	CPA
Does it change the proper functionality of the device under attacks?	No (Passive attacks)	No (Passive attacks)	No (Passive attacks)
Does it require direct access to the device under attacks?	No Noninvasive attacks	No Noninvasive attacks	No Noninvasive attacks
Does it require advanced or statistical analysis?	No	Yes	Yes. Power analysis is based on the Hamming distance model.
Does it require prior knowledge of the device hardware architecture?	Yes	No It is a black-box attack.	Most effective in white-box analysis. However, can also be used for black-box analysis.
How many power traces are required?	Single or a few power traces	Larger number of power traces	Lower number of power traces than DPA
How effective to obtain high quality signal in noisy environment	Not very effective	Effective	Very effective
How efficient is this attack?	Less efficient and less powerful	More efficient and more powerful	Very efficient and powerful

Table 1: Comparison of different PA attacks: SPA, DPA, and CPA

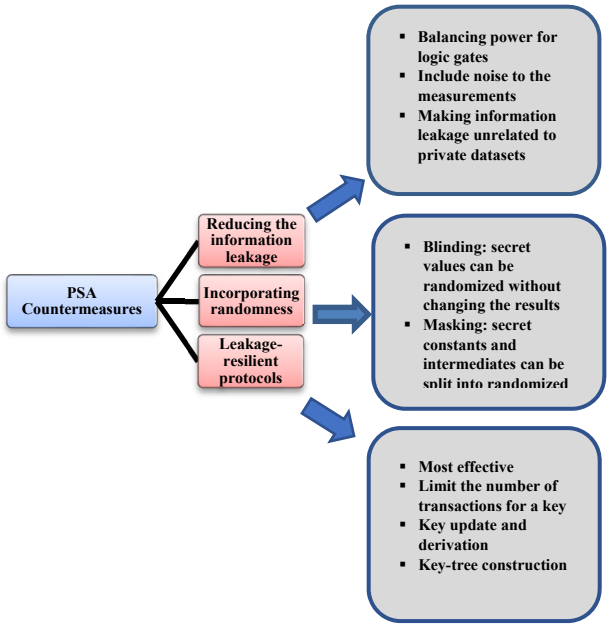

Fig 3: Classification of PSA countermeasures

3. Power Analysis Countermeasures

Power analysis countermeasures can involve modifications to multiple levels of a device, from modifications to the cryptographic algorithm, to modifications to the logic gates and hardware architecture. Typically, countermeasures can be classified into one of three categories, with most falling into a category commonly referred to as Hiding. Hiding countermeasures attempt to hide the underlying cryptographic operations by one of two means, either by randomizing the power consumption through the generation of noise or by equalizing the power consumption of the device, regardless of the computations being performed [5-7]. A second category of countermeasures is commonly referred to as Masking. For symmetric algorithms, masking countermeasures randomize the intermediate values produced during cryptographic operations. These intermediate values are dependent on the secret key and are the primary target of PA attacks [5-7]. A third category of countermeasures involve updating the cryptographic keys stored on a device [8]. Typically, cryptographic keys are stored within a register on a device and are never updated.

SPA vulnerability can be prevented by eliminating large peaks, using constant execution paths, and avoiding conditional branches. In addition, primitives and the instructions should be chosen carefully so that they do not leak enough information to recover the secret key. However, countermeasures for DPA are more involved since DPA can exploit very small information leakage and obtain high quality signal in noisy environment by employing advanced statistical analysis. Most countermeasures derived for DPA attacks will work for CPA attacks as well [24, 27].

3.1 Masking

Masking is a counter measure that randomizes the secret key dependent intermediate values produced during cryptographic operations. For example, during AES encryption, numerous rounds of substitutions, bitwise XOR, row shift, and linear transformations, are performed on a plaintext input [9]. The resultant intermediate values between operations are the target of PA attacks, in particular the output of the Add Round Key operation of first round or the output of the SubBytes operation of second round [10]. Prior to performing these operations, a randomly generated mask can be applied to randomize the plain text input. As a result, the intermediate values become randomized. The cryptographic transformations of the mask are tracked along the way, so that the effect of the mask can be removed prior to outputting the ciphertext. Software- based approaches have been proposed, which use Boolean or arithmetic masks. Boolean masking performs bitwise XOR operations, while arithmetic masking uses mod32 addition and subtraction operations [9]. Hardware based approaches have also been proposed and implement masking at the logic gate level [11].

3.2 Operation Shuffling

A software-based countermeasure, referred to as operation shuffling, was developed for randomizing the power consumption of symmetric algorithms. The counter measure involves randomly shuffling the execution order of the substitution box (S-box) operations within the SubBytes stage for each byte of data [8]. The S-box operation refers to the transformation of the data using a substitution cipher.

3.3 Dummy Instructions

Another software-based counter measure consists of inserting a random number of dummy instructions sat various points of the instruction sequence. DPA attacks require obtaining and comparing a large number of power traces during an analysis. Inserting random dummy instructions has the effect of desynchronizing or breaking the temporal alignment between the power traces, making them more difficult to compare. The total number of instructions added to the instruction sequence is kept constant. This prevents attackers from gaining any information about how many instructions were added [8].

3.4 Secure Double Data Rate Registers

Bellizia et al proposed the use of secure double data rate registers (SDDR) in place of conventional registers when implementing AES-128 encryption in a cryptographic ASIC.AES-128 encryption consists of numerous rounds of SubBytes, AddRoundKey, ShiftRows, and MixColumns operations, which generate key dependent intermediate values [11]. In the proposed circuit, the intermediate value of each operation is temporarily stored in a SDDR, rather than a conventional register. A SDRR is essentially composed of a multiplexer and two cascaded registers. With each clock pulse, the multiplexer selects between inputting the intermediate value and a randomly generated value. Thus, at any point in time, one of the registers will be storing a random value, while the other register stores the actual value. As a result, the power consumption due to data storage becomes randomized.

3.5 Non-Deterministic Processor

Non-deterministic processors, though similar to general deterministic processors, include additional hardware for randomizing the execution of instructions. These processors take advantage of the use of Instruction Level Parallelism (ILP), which is commonly implemented in higher performance processors. ILP allows for the execution of non-dependent instructions simultaneously in a parallel processor pipeline. Instructions are considered non-dependent if they do not depend on the results of another instruction that has yet to be executed or finish executing. They are also considered non-dependent if they do not overwrite any data that is needed by other instructions is yet to be executed or finish executing. Grabher et al [12] proposed a design for a circuit for randomly selecting non-dependent instructions.

3.6 Clock Signal Randomization

One hardware-based countermeasure, referred to as Clock Signal Randomization, consists of continually varying the frequency of the system clock. In comparison to operating the clock at a constant frequency, this has the effect of compressing or stretching the power trace for each clock cycle. Similar to other countermeasures, this desynchronizes or breaks the temporal alignment between power traces. One approach is to vary the output frequency of the clock by randomly gating the clock causing it to skip clock pulses [13], while another approach used multiple phase shifted clocks and a multiplexer to randomly select between the clocks [14], [15].

3.7 Dynamic Voltage and Frequency Scaling

Yang et al [16] proposed Dynamic Voltage and Frequency Scaling (DVFS), a method for randomizing both the clock frequency and voltage supplied to the cryptographic device. The proposed system consists of a scheduler (DVFSS), and a feedback loop (DVFSFL). During operations, the DVFSS unit randomly generates a data value representing the frequency and voltage and sends it to the DVFSFL. The DVFSFL then physically implements the desired clock frequency and outputs the supply voltage to the device. The generated values ranged from 1.8V at 450MHz to 0.9V at 250MHz.

3.8 Power Line Isolation

Another hardware-based countermeasure, referred to as power line isolation, involves decoupling the cryptographic device's power supply from the external power source [17], [18]. This was implemented by Tokunaga et al. by using a bank of capacitors and control switches. At any point in time, one capacitor supplies power to the cryptographic device, while the other capacitors are charged by the external power source. After the device executes a set number of clock cycles, the capacitor is then discharged to a predetermined voltage and the capacitors switch roles. Thus, when an attacker monitors the external power source, they only see the power consumed by a charging capacitor. This provides significantly less information to the attacker in comparison to monitoring the power consumed for each instruction or operation.

3.9 Constant Power Computing

Masle et al [19] proposed a design for a circuit that maintains a constant power consumption regardless of the computations being performed. The design for the circuit was based on a closed-loop control system and consists of three main components: a power monitor circuit, a power consumer circuit, and a controller. The power monitor measures the on-chip power consumption and sends the power measurements to the controller. The controller circuit consists of a proportional-integral-derivative (PID) controller, which uses the power measurements as feedback and makes continual adjustments to the power consumer circuit in order to maintain a constant power consumption.

3.10 Nonlinear Key Update

One countermeasure that has been proposed is to regularly update the secret key. This could consist of using a counter to track how many times the key has been used. After a predefined number of uses, a new key will be generated. This would prevent the attackers from gathering the large number of power traces required for performing a DPA. As a means of updating the key, one proposed approach uses the Secure Hash Algorithm (SHA) [8].

4. Analysis

Detailed reviews of PA attacks and countermeasures can be found in [5], [6], [7]. These reviews included numerous theoretical countermeasures that were not discussed here. Other countermeasures that have been tested in simulations, were also not discussed here. This paper only considered counter measures that were tested and verified to work in a hardware implementation. Table 2 summarizes the previously discussed countermeasures based on their effect on performance, power consumption, and space requirements for additional hardware. Also summarized in the table is a measure of the counter measures resistance to PA attacks. Typically, a cryptographic algorithm implemented in hardware is more resistant to PA attacks than when implemented in software.

Countermeasure	Performance	Power consumption	Area/Space	Resistance
Secure Double Data Rate Registers [11]	0%	+180%	+33%	Traces >100000
Non-Deterministic Processor [13]	0%	+180%	+33%	Traces >100000
Non-Deterministic Processor [13]	0%	Increase	+85%	Traces 20000
Clock Signal Randomiza-tion [14]	-5.33%	Increase	+70%	SNR -79%
Dynamic Voltage and Frequency Scaling [17]	-16%	-27%	Increase	PTE +7.5% TTE + ∞%
Power Line Isolation [19]	-50%	+33%	+7.2%	Traces > 10000000
Constant Power Computing [20]	0%	+28%	+26%	Autocorrelation 170%
Nonlinear Key Update [9]	-	Increase	-	High

Table 2: Countermeasures based on their effect on performance, power consumption, and space requirements

In hardware, cryptographic operations can be performed in parallel, causing the power consumption of these operations to overlap on a power trace. This is in contrast to software executing on a processor, where instructions travel through a processor pipeline and are executed in a sequential nature [8]. For example, in one study, algorithmic masking and shuffling were combined together requiring 50,000 power traces in order to recover the secret key [20]. These counter measures require no change to the underlying hardware. However, they typically introduce a significant performance overhead in comparison to hardware-based countermeasures. The secret key remains undisclosed after 10 million power traces [18, 10].

5. Conclusion and Future Work

The resistance to PA attacks was rated in number of power traces, time trace entropy (TTE), power trace entropy (PTE), signal-to-noise ratio (SNR), autocorrelation, and High or Low. An in-depth comparison of countermeasures is difficult due to the lack of a standard metric for PA resistance. Thus, the field of hardware security could benefit from the use of a standard metric, such as the number of power traces. For IoT developers, the selection of countermeasures would depend on the desired level of performance, energy efficiency, space constraints, level of protection, and costs. Future work is needed to perform a more in-depth survey of PA countermeasures.

REFERENCES

[1] M. Alam, H. A. Khan, M. Dey, N. Sinha, R. Callan, A. Zajic, and M. Prvulovic, "One&Done: A single-decryption EM-based attack on open ssl's constant-time blinded RSA," in *27th Security Symposium*, 2018, pp. 585–602.

[2] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Annual international cryptology conference*, 1999, pp. 388–397.

[3] T. S. Messerges, "Using second-order power analysis to attack DPA re- sistant software," in *International Workshop on Cryptographic Hardware and Embedded Systems*, 2000, pp. 238–251.

[4] J.-J. Quisquater and D. Samyde, "Electromagnetic analysis (EMA): Measures and counter-measures for smart cards," in *International Con- ference on Research in Smart Cards*, 2001, pp. 200–210.

[5] S. R. Shanmugham and S. Paramasivam, "Survey on power analysis attacks and its impact on intelligent sensor networks," *IET Wireless Sensor Systems*, vol. 8, no. 6, pp. 295–304, 2018.

[6] L. Zhang, L. Vega, and M. Taylor, "Power side channels in security ICs: Hardware countermeasures," *arXiv preprint arXiv:1605.00681*, 2016.

[7] A. Moradi and A. Poschmann, "Lightweight cryptography and DPA countermeasures: A survey," in *International Conference on Financial Cryptography and Data Security*. Springer, 2010, pp. 68–79.

[8] S. Mangard, E. Oswald, and T. Popp, Power analysis attacks: Revealing the secrets of smart cards, 2008, vol. 31.

[9] T. S. Messerges, "Securing the AES finalists against power analysis attacks," in *International Workshop on Fast Software Encryption*, 2000, pp. 150–164.

[10] D. Bellizia, S. Bongiovanni, P. Monsurro, G. Scotti, A. Trifiletti, and F. B. Trotta, "Secure double rate registers as an RTL countermeasure against power analysis attacks," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 26, no. 7, pp. 1368–1376, 2018.

[11] E. Trichina, "Combinational logic design for AES subbyte transforma- tion on masked data," *IACR Cryptol EPrint Arch.*, vol. 2003, p. 236, 2003.

[12] P. Grabher, J. Groschadl, and D. Page, "Non-deterministic processors: FPGA-based analysis of area, performance and security," in *Proceedings of the 4th Workshop on Embedded Systems Security*, 2009, pp. 1–10.

[13] K. H. Boey, Y. Lu, M. O'Neill, and R. Woods, "Random clock against differential power analysis," in *Asia Pacific Conference on Circuits and Systems*, 2010, pp. 756–759.

[14] A. G. Bayrak, N. Velickovic, F. Regazzoni, D. Novo, P. Brisk, and P. Ienne, "An EDA-friendly protection scheme against side-channel attacks," in *Design, Automation & Test in Europe Conference & Ex- hibition*, 2013, pp. 410–415.

[15] T. Guneysu and A. Moradi, "Generic side-channel countermeasures for reconfigurable devices," in *International Workshop on Cryptographic Hardware and Embedded Systems*, 2011, pp. 33–48.

[16] S. Yang, W. Wolf, N. Vijaykrishnan, D. N. Serpanos, and Y. Xie, "Power attack resistant cryptosystem design: A dynamic voltage and frequency switching approach," in *Design, Automation and Test in Europe*, 2005, pp. 64–69.

[17] A. Shamir, "Protecting smart cards from passive power analysis with detached power supplies," in *International Workshop on Cryptographic Hardware and Embedded Systems*, 2000, pp. 71–77.

[18] C. Tokunaga and D. Blaauw, "Secure AES engine with a local switched-capacitor current equalizer," in *International Solid-State Circuits Conference-Digest of Technical Papers*, 2009, pp. 64–65.

[19] A. Le Masle, G. C. Chow, and W. Luk, "Constant power reconfigurable computing," in International Conference on Field-Programmable Tech- nology, 2011, pp. 1–8.

[20] S. Tillich and C. Herbst, "Attacking state-of-the-art software counter- measuresa case study for AES," in International Workshop on Crypto- graphic Hardware and Embedded Systems, 2008, pp. 228–243.

[21] S. Bhunia and M. Tehranipoor, "Hardware security: A Hands-on learning approach", Morgan Kaufmann publishers, 2019, pp 1-502.

[22] M. Tunstall, D. Mukhopadhyay, S. Ali (2011), "Differential fault analysis of the advanced encryption standard using a single fault." In: Information security theory and practice. Security and privacy of mobile devices in wireless communication. Springer, pp 224–233.

[23] P. Kocher, "Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems". In: Koblitz, N. (ed.) CRYPTO, Lecture Notes in Computer Science, vol. 1109, pp. 104–113. Springer, Berlin (1996).

[24] P. Kocher, J. Jaffe, B. Jun, and P. Rohatgi, "Introduction to Differential power analysis," J Cryptgr Eng, 2011, Vol 1, pp. 5–27.

[25] J.S. Coron. "Resistance against Differential Power Analysis for Elliptic Curve Cryptosystems", CHES'99, LNCS 1717, pp.292-302,1999.

[26] Y. Zhou and D. Feng, "Side-Channel Attacks : Ten years after its publication and the impacts on cryptographic module security testing", IACR Cryptology, 2005.

[27] E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model", in: International Workshop on Cryptographic Hardware and Embedded System (CHES2004), Lecture Notes in Computer Science, Vol 3156, Springer, Berlin, pp. 16–29, 2004.

[28] C. Lavaud, R. Gerzaquet, M. Gautier, O. Berder, E. Nogues, and S. Molton, Whispering Devices: "A survey on how side-channels lead to compromised information", Journal of Hardware and Systems Security, vol 5, pp 143-168.

[29] D. Boneh, R. DeMillo, and R. Lipton. On the importance of checking cryptographic protocols for faults. In W. Fumy, editor, Advances in Cryptology — EUROCRYPT '97, volume 1233 of Lecture Notes in Computer Science, pages 37–51. Springer-Verlag, 1997.

[30] D. Boneh, R. DeMillo, and R. Lipton. On the importance of checking cryptographic protocols for faults. Journal of Cryptology, 14(2):101–119, 2001.

[31] P. Kocher. Timing attacks on implementations of Diffie-Hellmann, RSA, DSS,and other systems.CRYPTO'96, LNCS 1109,pp.104-113,1996

[32] M. Devi, and A. Majumder (2021). "Side-Channel Attack in Internet of Things: A Survey". In: Mandal, J., Mukhopadhyay, S., Roy, A. (eds) Applications of Internet of Things. Lecture Notes in Networks and Systems, vol 137. Springer, Singapore.

[33] S. Sullivan, A. Brighente, S. Kumar, and M. Conti. "5G security challenges and solutions: a review by OSI layers." IEEE Access (2021).

[34] M. Gohil, and S. Kumar. "Evaluation of classification algorithms for distributed denial of service attack detection." In 2020 IEEE Third International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), pp. 138-141. IEEE, 2020.

[35] S. Srinivasan and S. P. Alampalayam. "Intrusion detection algorithm for MANET." International Journal of Information Security and Privacy 5, no. 3 (2011): 36-49.

[36] S.P. Alampalayam, and A. Kumar. "Security model for routing attacks in mobile ad hoc networks." In 2003 IEEE 58th Vehicular Technology Conference. VTC 2003-Fall, vol. 3, pp. 2122-2126. IEEE, 2003.

[37] S. Kumar, A. Kumar, and S. Srinivasan. "Statistical based intrusion detection framework using six sigma technique." IJCSNS 7, no. 10 (2007): 333.

[38] D. Eastman, and S. Kumar. "A simulation study to detect attacks on internet of things." In 2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, pp. 645-650. IEEE, 2017.