[1]Amit Rathod

[2] Kajal S. Patel

Review of Side Channel Attacks and Comparison of Masking and Re-keying Countermeasures

Abstract: - Side Channel Analysis has been an open problem among the information security designers. As the industry is moving towards IoT based devices, security at hardware level has been of prime importance. Side Channel Analysis include various attacks like timing attack, power consumption related attack, Electro-magnetic leaks related attacks, memory footprint related attack, acoustic attack, attacks due to architectural vulnerabilities etc. These attacks bypass the mathematical security provided by the algorithms. Many of these attacks have been specific to cloud technology or server sides. However, power analysis attacks are such types of attacks that are common amongst all the processing devices like IoT, cloud server, smart cards etc. The Power Analysis attacks are further categorized to attacks like Simple Power Analysis (SPA), Differential Power Analysis (DPA), Profile attacks, etc. Successful Differential Power Analysis (DPA) attacks have happened in both research and real-world scenarios. Work has been done to make the attack powerful, efficient in the literature. Countermeasures such as Masking and Guarding have been implemented at various levels. Re-keying has also been studied by several researchers as a countermeasure for DPA and other power analysis attacks. Masking relies on making the statistical analysis quite difficult by adding layers. Whereas, re-keying focuses on making the discovery of key futile by leaving with less number of samples to make the statistical analysis infeasible. All these methods have been researched and discovered in isolation. This paper focuses on the holistic review of masking and re-keying to conclude the vital parameters and qualitative analysis to compare the two different categories of schemes and explores the possibility towards a hybrid approach that would develop an optimal solution to counter the DPA and other related attacks.

Keywords: Differential Power Analysis (DPA), Security, Side-Channel Analysis (SCA), Masking, Re-keying.

I. INTRODUCTION

The mathematical security of the encryption algorithms assume that the implementations of these algorithms are in an isolated, secure, and closed environment. However, the implementation instruments have their architectural and technology related vulnerabilities and they may unintentionally leak the information at the time of processing. This information can be in the form of a power signal spike, a sound, a light, time the algorithm takes to process the data, ability to manipulate the adjoining bit in the memory etc. As per [1], there are many types of attacks possible. The Timing attack, Power Analysis attack, Electromagnetic Analysis attack, Microprocessor architecture related vulnerability, Programming bug attacks, Acoustic attacks, and use of machine learning, deep learning and artificial intelligence to recover the full keys. In the paper [2], the history of different types of Side Channel attacks is discussed that gives a broader view on the open problems.

A. Timing Attacks

Cache-timing attack is a well-known timing attack that has been used by the attacker. The implementation vulnerability of using the square and multiply method to implement the RSA algorithm is the vulnerable technique due to the difference in the time taken to process the multiply and the square functions. Along with the power traces, the attacker can completely identify the full key of RSA by running the simultaneous program in the same cache where the processing of the RSA algorithm is going on. The timing attack can be countered by adding some random delays in the execution time. Moreover, the timing attack is difficult for the symmetric encryption. Additionally, the timing attack requires that the attacker gains the access of the system to run another program in parallel.

B. Power Analysis Attacks

In the paper [3], the domain of Power Side Channel Attacks has been explored since it was first identified. The Power Analysis attacks have been the most prominent and real-world powerful attacks since then. Power Analysis attacks include Simple Power Analysis (SPA), Differential Power Analysis (DPA), Template Attack (TA), Prototype Attacks, Correlation Power Analysis (CPA) etc. attacks. These attacks have been discovered by Paul

[1] Gujarat Technological University, Ahmedabad, India

[2] Computer Engineering Department Vishwakarma Government Engineering College,Chandkheda,Ahmedabad

*Corresponding author: rathod.amit.h@gmail.com

Kocher [4] and various techniques have been used by getting the power consumption traces and based on the amplitude, the processing time signals can be retrieved. Based on their further processing, the above classification is derived. An SPA attack simply looks at the amplitude of the power consumption trace and tries to deduce the operation being performed that further leads to get a part of the key. A DPA attack includes getting traces with further preprocessing and post processing techniques to try to deduce the operation and the part of the key. The DPA is the main focus of this paper and in the further discussion, we shall discuss these attacks in detail. The TA is an attack that can be performed on a similar IoT device after performing an attack on a few of the devices and finding the similar pattern for the respective device make and model. The CPA attack tried to correlate the power traces with the known plaintext attacks to be able to identify the underlying key patterns. The attack has been formalized and a kit for research and recreation of the attack has been developed called Chipwhisperer.

C. Electromagnetic Analysis Attack

In order to reduce the fabrication cost of wifi and communication modules, the processing module and Radio Frequency (RF) modules are mounted on a single chip. Due to this, the processing data is leaked on a specific frequency other than the frequency of the official RF communication assigned. Such an attack is called Electromagnetic (EM) attack. In the paper [5], the authors have used the EM attack to be able to retrieve the artifacts to be able to use in the digital forensics. Successful AES key recovery in a laboratory environment has been performed in the paper [6] and [7] and proven that the EM attacks from a distance are possible. The EM attack focuses on discovering the frequency of the side channel leakage and once that is recovered and pre-processing is performed, the attack becomes like the power analysis attack. As the scope of this paper is on power analysis countermeasures, further discussion will focus on Power Analysis attack.

D. Acoustic Attack

Acoustic attacks include the use of the sound of the keystrokes, the minute sound produced by the devices, and other forms of audibles. These auditory artifacts are collected for further analysis and try to recover the encryption keys. In the paper [1], discussion on acoustic attack is provided.

E. Attacks due to architectural vulnerability

For hardware implementations of the security modules, during the fabrication and manufacturing process, the vulnerabilities are left and later on their identified and exploited. Spectre, Rowhammer, and Meltdown are some of the examples mentioned in the paper [1] that have been identified and exploited by the adversaries. The countermeasure against them would only be the hardware changes in the subsequent fabrications. Driver patching also helps.

F. Attacks due to software bugs

Though the software bugs and vulnerability are related to cyber security, the attacker combines the information available through the side channel along with the vulnerability to be able to perform the attack. Heartbleed is one such vulnerability exploited in OpenSSL by the attackers.

As the Power Analysis Attack has been one of the most powerful attacks that has been explored for more than 25 years and examples of full key recovery has been demonstrated, and yet there is scope for improvement for both attack and countermeasure side, this paper focuses primarily on the Power Side Channel Analysis. Section II of this paper discusses the various types of Power Side Channel attacks. Section III discusses countermeasures primarily the masking and re-keying. Section IV discusses the advantages and limitations of the masking techniques. Section V discusses the re-keying strategies and the need for the combined strategy and the baseline parameters for comparing the two different strategies. The qualitative analysis has been discussed for the same. Section VI discusses the common grounds for evaluating the two different types of strategies, and section VII discusses the outcome of the discussion and conclusion of the review with future directions for developing the countermeasures for the Power Analysis attacks.

II. POWER SIDE CHANNEL ATTACKS

The Power side channel attacks can be classified as per the figure 1. Each of them is discussed in brief. The Differential Power Analysis (DPA) has attracted the maximum attention of the researcher and has been in practical usage and an open problem.

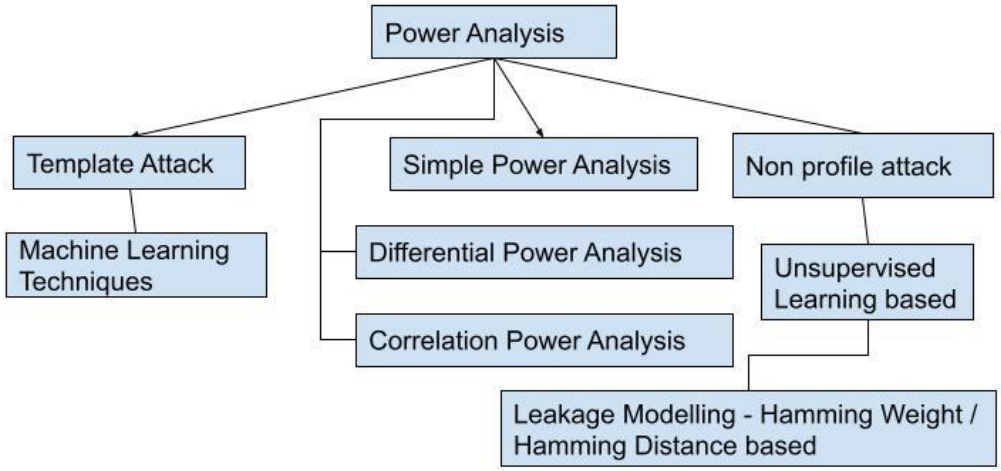

Figure 1: Classification of Power Side Channel Attacks

The simplified anatomy of the above classified attack has been given in the figure 2. Each of them has been discussed in detail in the subsequent sections.

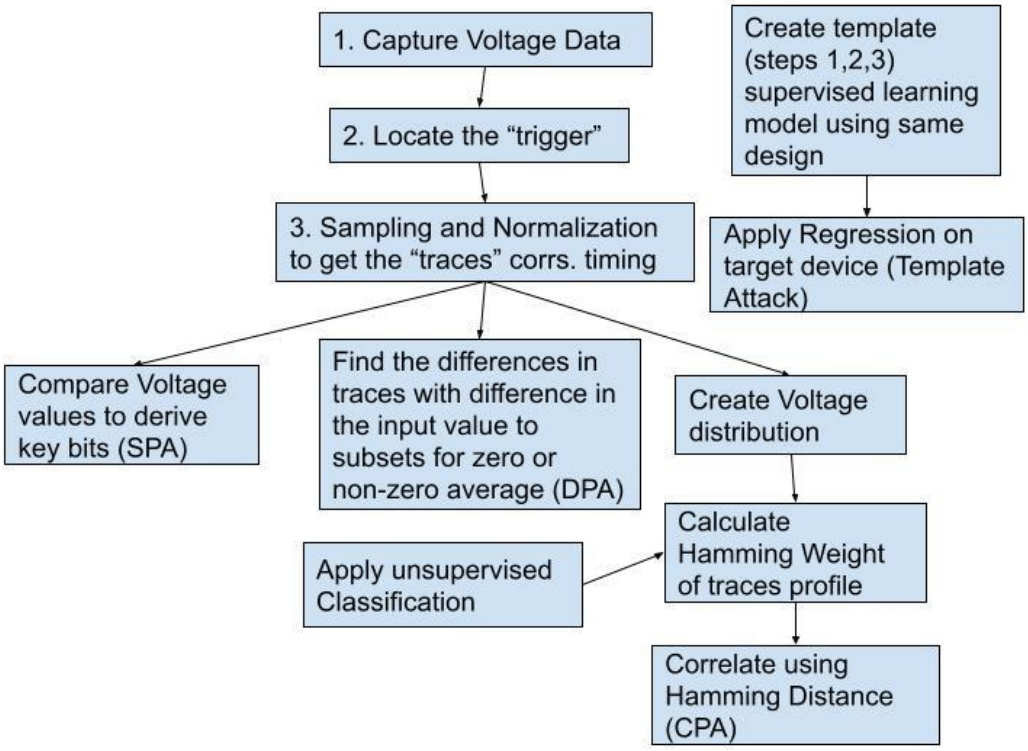

Figure 2: Anatomy of the Power Side Channel Attacks

A. Simple Power Analysis (SPA)

The SPA attack is the first power side channel attack that was discovered. It can be effectively implemented on public key cryptosystems like RSA. The Simple Power Analysis attack can be performed by using the oscilloscope and the power usage measurement. The RSA algorithm implementation using the Chinese remainder theorem, uses the square or multiply operation. Both the operations use the different power consumption values and hence, the power usage pattern we get can help us find the details in it. The paper [4] discussed the SPA attack on RSA and then they used the less attackable Elliptic Curve Cryptography. With respect to the symmetric key cryptosystem, the changes in the type of operation can be identified and using that, the start point of the encryption, the change of the operation can be identified. As discussed in the [9], the unsupervised approach has discussed the two stages of power side channel analysis in which the first one focuses on identification of the encryption operation using the SPA. There are some countermeasures available against the SPA attack like adding noise to the operations, using

dummy instructions and desynchronization. These countermeasures have been futile when further data analysis on Machine Learning is applied that discovers such patterns.

B. Differential Power Analysis (DPA)

The DPA attack is performed by extracting the underlying patterns from a large amount of power traces of the different plain and cipher text pairs, by using the statistical analysis methods. The basis for the DPA lies in the XOR operation of bit 0 with another 0 resulting in 1 and operation 0 with another 1 will result in 0. The DPA attack is more relevant with the symmetric key cryptosystems like AES. The AES extensively used the XOR operations and hence, if there is a control over the plain text, the algorithm is fed with the consecutive change of bits, and the power traces extracted are examined thereafter. Wherever, the key bit is 0, as per the above discussion, the power trace change is detected and hence, revealing the part of the round key for the AES algorithm. This process with sufficiently large amounts of pairs of plain text-cipher text and the power signal trace pairs can be examined using statistical analysis to determine the part of the round key. As per [17], these traces are small to recognize but the whole keyspace can be divided into various subsets of the such power profiles and the average and cumulative power profiles are classified. This classification can reduce the brute force key space to be able to determine and recover the full key. Machine Learning (ML) based techniques are used like PCA or LDA to find the discriminant and the model is prepared to reduce the key space to be able to recover the full key in case of the DPA attack. There have been countermeasures like guarding at architectural level, masking at software and hardware implementation level to further disrupt the statistical properties. With the advancements of the ML and Deep Learning (DL) techniques, the attacks have also been advanced. The paper [10] discusses the DL approach in detail and the dataset. The hardware that is architecturally guarded exhibits a different type of vulnerability called Correlation Power Analysis (CPA) that is discussed as follows.

C. Correlation Power Analysis (CPA)

The DPA needs the minor changes in the leakage profile to be able to develop the model for classification to perform the attack. On the other hand, if we are able to generate the power traces for all the keys for the particular chose plain text and we can extract the power traces and find the correlation with respect to the power values like Hamming Weight and Hamming Distance for the given key sets, we get the power usage Probability Density Function for the said device. We can then use the given plaintext and then find the correlation with respect to the pearson correlation coefficient [15] to be able to match them with the model to find the distance. The distance value reduces the number of possible permutations to be able to discover the key. The CPA attack has been widely researched, implemented and reconstructed using modules like Chipwishperer. The CPA concept has been used to develop the profiling and template attack. The device does have its own noise and leakage profile apart from the cryptographic information leakage and hence, this attack is useful only for the device it has created the model for. Moreover, the leakage profile also changes over the time and usage of the hardware chip.

D. Profiling Attack

For the profiling attack, the attacker uses the leakage profile of the device that can help with identifying the noise pattern in the device. The profile attack is a range of various attacks that include various parameters for building the device profiles [11]. The profile developed using a device can be the basis for the other similar make and model devices. The most common types of side channel profile attack are Template attacks.

E. Template Attack

The template attack is a specific type of profiling attack that uses the device specific leakage function. The template is built and then the power leakage trace is evaluated against the template [12]. Each device has its own noise profile; identifying the noise profile helps with the pre-processing to get the better power traces that can correlate with the data being processed. Based on this, the template attack executed as follows [13]:

Profiling phase: For each key k in key space of 8-bit {0,1,2,....,255}, 3072 traces were recording with sampling rate of 2500 samples per trace making total = 3072*255 traces and for each of them, 2500 samples of the voltage level. As the number of samples per trace is very high, dimensionality reduction techniques like 1 sample per clock(6-10 per instruction), 2 samples per clock, 3 samples per clock and all samples per clock are taken to check the effect. However, all of them still provide a large number of samples leading to singularity. To avoid this, the Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA) with 4 samples are chosen using the elbow method. The Sample mean and Sample covariance matrix are calculated with respect to the true mean

and true variance.The multivariate normal distribution model is applied on the data and Probability Density Function (PDF) is calculated with sample mean and sample covariance values.

In the attack phase, we use the same dimensionality reduction technique or compression method used in the profiling phase and "Discriminant score" is computed by using the PDF function of the profiling stage. The discriminant score helps in computing the likelihood computation based on a-priori probability and will find the reduced number of candidates for the key. Then applying the brute-force to recover the key. The various approaches of data compression or dimensionality reduction were compared using the Guessing Entropy. The lower the guessing entropy, the better the approach.

The above approach has proven that the PCA approach is better with the template. With this approach, the noise profile of the template device has been identified. In the later research [14], the template has been tested at different points of the life cycle of the hardware chip along with the other chips of the same fabrication. In [15], the authors have manipulated the eigenvalues of the DC content to overcome this issue and have proven to be working now on other devices with different DC content profiles. The question of portability to other devices were also examined and addressed in the paper [16] with a different approach of waveform realignment and acquisition normalization to apply the template on other devices. Recent development in the DL for the template attack has proven that with around 104, the CNN approach can recover keys with minimum efforts with the lowest guessing entropy even on the templates. The paper [17] has discussed in detail the various Machine Learning based approaches for the power side channel template attack for both the public key cryptosystems and the symmetric key algorithms.

If we look at the countermeasure strategies for the template attack, the recent attacks are using the ML and DL based approaches and to train the model they need at least a minimum number of traces with statistical mapping for the same key pair to be able to model the key. Masking has been one of the ways to hide the original implementation but it has to be reversed before it should be sent over and hence due to the reversible function, the masking features are also extracted by the algorithm. Masking makes the attack quite difficult but with some additional effort, that is removed using the ML and DL techniques. Another approach is re-keying where instead of the main key that is called master key, another key derived and shared with the receiver from the master key, is used so even if the part of key is discovered, it may not be able to find the master key. If the derived key is changed before enough traces are available for ML and DL to prepare a model, this strategy can prevent the attack. Re-keying comes with its own overhead of securely sharing the derived keys periodically. The next section describes these 2 countermeasures in detail.

III. COUNTERMEASURE STRATEGIES

The SCA countermeasures are primarily the guarding, masking and re-keying. Guarding is to prevent the leaks from the chips and involves changes in the architectural level. That is beyond the scope of this paper. The other two scheme are as follows:

A. Masking

In the Power or EM SCA, the adversary is listening to the power leakage profile to deduce the information being processed to ascertain the key. The Masking operation directs towards diverting the adversary towards a part of value only by splitting the original value being processed into multiple shares of values so that the leaked power profile does not render any value for the analysis [18]. The Masking has to be complimented by removing the mask before it is sent over the network. Hence, the unmasking process is the remaining part of the share that is applied again to gain the original data. This would distort the statistical property of the processing from one point over the number of points depending on the number of shares.

The basis for the need of masking was discussed in the paper [24]. The authors discussed that the amount of leakage increases with the multipoint probing and that lowers the entropy. A mechanism to increase the entropy is needed. A generic two share masking scheme for AES S-box would have the following formula [19] where min and mout are the masking and unmasking shares on the S-box.

S'[x]:=S[xm in]m out

B. Re-keying

Another way to protect the key from the SCA is to use the derived key using the main master key and shared randomness and then change the derived key after a certain period to make the power analysis futile with insufficient samples to develop the model for the attack. As the derived key is changed after a certain period of time, the number

of power traces are insufficient. Theoretically, the scheme is secure against the power analysis attack. However, it uses the re-keying algorithm which needs to be guarded against and the sharing of the random number needs to be over the securely transmitted to protect the algorithm. Additionally, to derive the session key, there is an overhead of periodically sharing the randomness. Several practical re-keying schemes have been discussed in [30]. Detailed analysis on re-keying schemes is discussed in the subsequent sections.

IV. MASKING

In the Power SCA attack, the adversary tries to read the operation that is being performed and wishes to deduce the data being produced to ultimately discover the key. The Masking technique tries to make this process difficult by manipulating the data and the key such that even if the adversary is able to deduce the data being processed, the data is masked with some random number which will be inverted in the subsequent operations. For this reason, randomness variable r function is made in such a way that the value of r is such that r = r'x r'' where x is an operation and when the adversary tries to read the data, either the data with some operation on r' or the data with r'' operation is read but not both by the adversary making the analysis difficult. Masking can be as easy as XOR operation and can be complicated based on the type of masking. Any symmetric encryption like AES has two types of operations namely linear operations and non-linear S-box types of operations[3]. The figure 3 provides the general idea of masking with respect to unmasked operation. The left side is an unmasked operation and the right side is a masked operation. Meaning that the data a is a function of a0 and a1 and, b is a function of b0 and b1 and they're the shares of a and b respectively. In the masked operation, the first operation signifies the masking and the second operation is unmasking, rendering the original expected value.

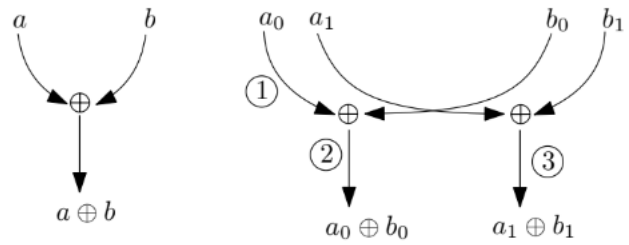

Figure 3: Masked and Unmasked Operations[19]

The masking uses both types of linear and non-linear operations to hide the original data from the adversary. Masking applied on the data needs to be removed also so the functions need to be reversible. Researchers have proposed many masking methods so far. Their easy representation and understanding of the schemes are discussed in the following subsections.

A. Boolean Masking

The most basic type of masking is boolean masking. It is applicable generally on the linear functions as it is used to mask the linear operations through the simple XOR operation [19]. Boolean masking with 2 shares is useful for first order masking. If the mask is divided into more than 2 shares, it is called the second and higher order masking. It is also called additive masking.

B. Multiplicative Masking

The multiplicative masking is applied over the Galois Field G.F.(256) and is generally applied on the non-linear components of an algorithm like S-box in the AES[19]. The first scheme for multiplicative masking was proposed in [20]. The authors have used the multiplication over G.F.(2^8) for masking and it has been the basis for much of the research further for non-linear masking. The authors have applied the masks on the S-boxes of the DES and AES with significantly higher overhead with respect to execution time and memory requirement. With time complexity almost doubled in case of DES and almost 3 times in case of AES. The subsequent work focused more on either reducing the time or space complexity. In the paper [21], the multiplicative masking on G.F.(2^8) proved to be a complex one given the implementation on the circuits and additionally using the only one Galois Field could render it vulnerable. Therefore, the authors have come up with "tower field" meaning instead of using that large multiplicative field, the S-box fields were divided into smaller fields of G.F.(2^4) or G.F.(2^2) and using the dynamic strategies based on the random input provided robust masking with lesser overhead and less complex architecture. In [22], the authors have further tried to make the multiplicative masking to linear masking using the

G.F.(4) and thus simplifying the masking with efficiency for the defense against the first order attack with ease of implementation. The paper [23] discussed uses of the composite field arithmetic and using the G.F. (4) and G.F.(16) for implementing the S-boxes to make them further compact and efficient.

C. First Order Masking

The text and the key is applied with the mask that is shared in 2 parts is called the first order masking. The d order masking has d+1 shares. Without masking, during the execution of the algorithm, at any given point of interest, there is a leakage profile using which the adversary uses the Probability Distribution Function to reduce the key space. This can be distorted by neutralizing the point of interest and making two points of interest by applying masking and then again applying unmasking which makes the properties of the single point of interest distributed over multiple points. In this case, the adversary applies ML and DL algorithms to discover even the multiple points of interest to remove the masking. Such attacks are Second Order DPA(SODPA) and Higher Order DPAs (HODPA). This further requires the masking algorithms to use more than 2 shares to further distribute and increase the points of interests that further complicates the power analysis.

D. Second and Higher Order Masking

The paper [26] discussed that as the number of points of interest increases with first order masking, the adversary can use the advanced techniques of unsupervised learning combined with ML and DL to remove the mask. There is a need for second or higher order masking. Nth order masking required N+1 shares. For the second order masking, the formula for the linear parts for the random number r is distributed over 3 numbers namely r1, r2 and r3. For the S-box computation that is a non-linear component, the function is defined in such a way that the share values are stored in multiple state tables that increases the space complexity. The paper [25] discussed the table compression scheme to reduce the size of the S-box state tables. The paper [27] uses a different view to achieve compression by converting Multiplicative Masking (MM) to Additive Masking (AD) and vice versa. Due to which the authors claim to use the same tables to compute the values of different shares for S-boxes in the Galois Field G.F.((2^m)*..)^n where m and n are variables based on the number of shares needed for any particular order of the masking. Therefore, by adding the algorithmic overhead, this scheme is useful for the space constraint applications as it can use the memory only for storing a few shares at a time and remaining can be computed as needed.

The paper [28] uses the similar concept as shown in paper [27] with the slight modification on how the exponential function is implemented. The authors have divided the types of exponential variables into either odd or even. In case of the odd exponent, the exponential function is implemented in terms of multiplication. This multiplication scheme is time consuming. On the other hand, if the exponent is even, the exponential function can be deduced to a square function and squaring function is faster compared to the multiplication function. Hence, the modified higher order algorithm is more efficient than the earlier paper.

The paper [29] questions the use of linear masking and claims to be vulnerable. Instead of the linear masking, the paper proposed to use the affine masking scheme over the G.F.(256) over G.F.(2) for the linear addition and multiplication with some overhead. The same can be used for the non-linear masking also and that saves the different additional overhead for the different scheme. Instead of masking at the bit level, the masking is applied at word level and with using higher size registers, the process is faster and even the linear parts are difficult to provide any additional information that could be useful for unmasking the non-linear parts.

E. Masking Comparisons

Based on the above discussion on masking, following is the comparison table 1 for various schemes. The comparison table is aimed towards identifying the qualitative value addition a researcher can follow keeping the trade-offs in their mind. In all the cases, it is assumed that the masking operations are secure against the side channel analysis. Also, it is observed that for higher order masking, the algorithmic intervention is towards reducing the computing complexity of multiplication over the Galois field. Based on the comparative analysis, the following parameters have been discovered that can be taken into consideration while designing, developing and evaluating the masking schemes.

- Masking Order
- Timing Overhead
- Space Overhead
- Masking Scheme (e.g. Boolean, Multiplicative, Affine)

Table 1: Masking Schemes

Algorithm	Novelty	Trade-off
Akkar [20]	First Order Masking	Time and Space Overhead increase
Canright[21]	Improved algorithm for first and second order	Time Overhead increased with space overhead marginally increase
IAIK[22]	Simplifying Multiplication	Reduce space overhead
Zakeri[23]	Composite Field Arithmetic	Reduce space and time overhead for complex design
Valivati[25]	S-box compression scheme for 2 order	Compact S-box with additional computing
Coron[26]	Higher order Masking Tables	Less efficient for AES better in DES
Genelle[27]	Squaring over multiplication for even orders	Efficient than multiplication
Rivain[28]	Provably secure software implementation	Efficient for hardware not for software
Fumaroli[29]	Affine masking over Linear	Secure with overhead

V. RE-KEYING TECHNIQUES

Masking schemes have been prevalent since the introductions of the block ciphers. Since the introduction of the block ciphers, the modes of operations have been in use and are in a way masking techniques. Modes of operations like ECB, CBC, CFB, OFB and CTR etc. These modes are a way to mask the data before the cipher is processed. Masking schemes mask the data and operations both which adds to the processing overhead. Masking tries to cover the operation data such that if an adversary reads the operation being performed, the adversary reads the masked data and not the actual one. However, the adversaries have improved their attack using DPA with ML and DL techniques such that, given sufficient number of power traces, even the masking can be removed. Re-keying tries to overcome using the method in which the adversary does not get the enough traces to discover the main key even if the operation is not protected against collection of the power traces, and even if the temporary key is discovered. Similar to protecting the masking operations in a secure way, re-keying also assumes the temporary session key is generated in the secured environment.

The paper [31] discussed that using the re-keying approach, the lifetime of the master key is increased. In paper [32], the authors have classified the re-keying into internal and external re-keying where, for the external re-keying the key needs to be shared while internal re-keying is done by manipulating the same session key to derive multiple keys to be used to prolong the life of the session key. The authors also claim that using the combined approach of both types of re-keying called composite method, the lifetime of the shared session key and the master key is significantly increased. The internal re-keying majorly uses the sequential mode and hence the encryption and decryption process should happen in the sequence and can be useful only if there is no change in the ciphertext during the execution and the data is processed in the synchronous way. Following is the generic structure of any re-keying scheme[33].

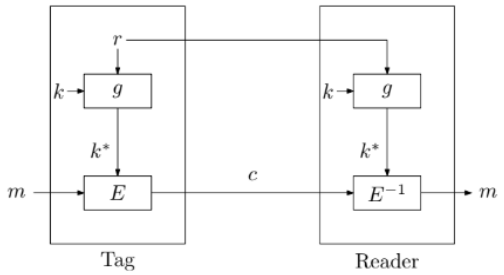

Figure 4: Structure of a general re-keying scheme[33]

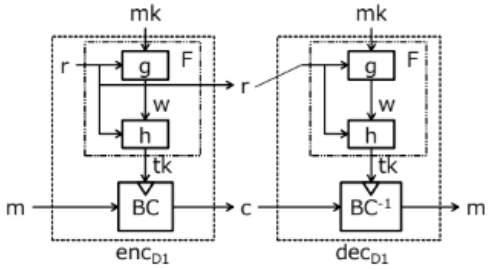

Figure 5: Re-keying scheme as in [35]

As in the above figure, the first scheme [34] was introduced in which the function g was used and assumed to be implemented securely. Here, the cipher text along with the random r is also shared publicly. As the k* is dependent on the r, and k* can be recovered using the Side Channel Attack from the c, and the function g is reversible, using the birthday attack, the master key k can be recovered. Hence, in [35], a new scheme was proposed where instead of one invertible function, two non-invertible functions were used as shown in figure 5.
In this scheme, the function is not invertible instead, the session key calculation involves 2 functions instead of 1. Therefore, if the functions g and h are sufficiently resistant, the adversary can only recover w only from the session key. However, this scheme is also not useful against the known plaintext attack to recover the session key and using that subsequent message m. In this case, another version with birthday security is provided by the same author with using the internal re-keying system to thwart the birthday attack. The authors of [36] have proposed the concept of tweakable block cipher which is similar to the mode of operations CFB and OFB, where we use the Initialization Vector along with the key in a synchronous mode, similar to that, the cipher is tweaked where such additional tweaking value is used in the sequential synchronous mode to thwart the birthday attack. Based on that, the tweakable block cipher is used.

In the above method, if the operations are not secured with guarding, the masks can be applied that are sufficient for providing a tweakable cipher with masking as in the figure 6 as proposed in [30]. This scheme is resistant against first order attacks.

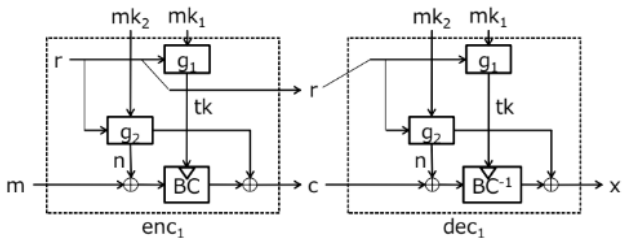

Figure 6: SCA resistance tweakable cipher[30]

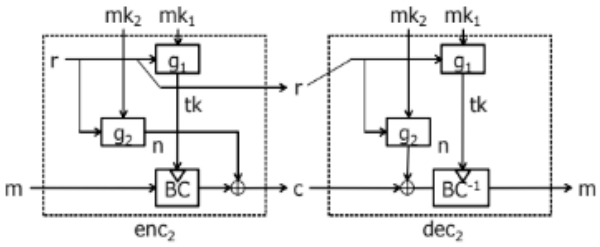

Figure 7: Efficient SCA resistance tweakable cipher[30]

As in the figure 6, we can see that the mask n is used and is dependent on the random r, which is shared on both the side, the authors have come up with more efficient scheme where the masking is applied on the encryption side while unmasking is done on the decryption side as shown in figure 7. These 2 schemes of [30] have an overhead of sharing of randomness r and using 2 master keys. Additionally, to make them secure against the higher order attack, they have also proposed a scheme that uses 4 keys which is an overhead while refreshing the master key.

Table 2: Re-keying techniques

Algorithm	Novelty	Trade-off
Medwed[34]	Re-keying scheme on basis of function F and randomness r	Function is vulnerable due to reversible nature. Randomness Sharing is overhead
Dobraunig[35] - First Scheme	Reversible function converted to 2 irreversible functions	Added a layer against the basic DPA but vulnerable against birthday attack
Dobraunig[35] - Second Scheme	Above scheme with tweakable block cipher	Secure against birthday attacks. Needs synchronous mode
Liskov[36]	First Scheme on tweakable block cipher	Synchronous mode required. Decryption erroneous if bit changes
Yuichi[30] - First Scheme	SCA resistant tweakable block cipher against First Order DPA	Two master keys required. Two variants, of which in masking application and removal done on both sides
Yuichi[30] - Second Scheme	Efficient SCA resistant tweakable block cipher against First Order DPA	In above method, masking applied on encryption and removed on decryption

Comparison of all these re-keying techniques have been given in table 2 for the researchers to identify the open problems and the directions of the further research. The next section discusses the common grounds for evaluation

when discussing the SCA resistant security of masking against the re-keying. As both schemes use totally different approaches to solve the same problem, qualitative analysis is provided.

VI. Common Grounds for Comparison

A. Qualitative Analysis

So far, all the studies have focused their approach on either of the masking or re-keying techniques as a countermeasure against the Power SCA. Additionally, each of the techniques have their own overhead and advantages. Moreover, the efficiency of the various techniques has been studied in isolation and not as an overall efficiency. While the re-keying scheme seems to be more efficient than the masking scheme, the sharing overhead needs to be considered. Additionally, the frequency of the sharing of randomness is also needed to be considered.

Some of the parameters are qualitative and depend on the qualitative nature of the algorithm. To elaborate this claim, it is evident that some re-keying schemes need to be executed in synchronous manner as their temporary key depends on the value of the previous block. In practice, the packets and data may travel through various paths and they may not reach in a synchronous manner to the destination. Hence, the destinations need to wait till all the sequences reach in order before the decryption can take place.

This requirement will improve the security and efficiency when taken into isolation but when checked holistically, it is slower. Therefore, the differentiating idea of this paper is to check the qualitative approach and discover the additional parameters to focus on to achieve the practical comparison of various methods. The trade-off parameters along with the application environment can determine the appropriateness and practical applicability of the algorithms instead of focusing only on individual parameters like efficiency. Moreover, both the masking and re-keying techniques aim to thwart the Power SCA attacks, however when the choice is to be made between the two, the studies have not yet undertaken to either decide the common grounds for evaluations nor the parameters have been discovered to determine the parameters to prepare the comparative study across the different approaches. The following section discusses the parameters to compare and evaluate the masking and re-keying approaches that are not only on the basis of the efficiency of the algorithm but also considers the additional overhead those algorithms carry along with the practicality of the approach. Therefore, for comparison of various differing schemes, the parameters are classified into quantitative parameters and qualitative analysis for further studies.

B. The parameters for the comparison

As discussed above, we divide the comparison parameters into qualitative and quantitative as follows:

Quantitative Parameters: Researchers so far have focused on the time and space complexities for evaluating their algorithms with others. Additionally, the security against the order of DPA and guessing entropy has also been the parameters to consider when a comprehensive study is to be taken. However, when implemented, its overall execution impact is not evaluated and hence, the paper proposes the following evaluating parameters for evaluation and comparison.

1) Guessing Entropy: The basis of any security algorithm can be determined using its guessing entropy function.

2) Total Timing Overhead: The total timing overhead function T is defined with respect to the Masking overhead (m), Randomness /Key sharing overhead (s), Re-keying refresh frequency (f), Complete cycle (encryption/decryption) overhead (y) and some algorithm specific additional factors related to the particular (u). The function may add more timing parameters as it may not be limited to that.

$T=f(m,s,f,y,u,*)$

3) Total Space Overhead: The total space overhead S is defined with respect to the additional S-box table space (sb), additional key space overhead (k), additional buffer space need for synchronous mode (b), additional network payload space (p) and any other algorithm specific additional factors (us). The function may add more space constraint parameters as it may be beyond these as well.

$S=f(sb,k,b,p,us,*)$

4) Attack order protected against: To compare and evaluate various schemes and to ascertain its practical applicability in the real world with respect to the computational security of the algorithm.

Qualitative Parameters: In addition to the quantitative parameters, some of the parameters are related to applicability for the particular applications. Even if one scheme is efficient, to check if that is useful in a particular application setting, these parameters are needed.

- Modes of Operations
- Processing Mode (Synchronous / Asynchronous)
- Possible Vulnerable to attack type
- Assumptions

VII. Research Direction Forward And Trade-offs between Masking and Re-keying

The higher order masking techniques thwart the lower order attacks but with less computational efficiency. However, if compromised, the subsequent messages are compromised. On the other hand, the re-keying scheme claims to be sufficient to discover the master key, the re-keying function and sharing overhead is substantial. Both these parameters are independent and due to which, the combined strategy of:

- Applying masking on the Re-keying scheme can potentially prolong the life of session key and increase overall efficiency
- Using internal or external re-keying techniques with existing masking algorithms can thwart the adversary by rendering insufficient traces that helps to lower the guessing entropy.

VIII. Conclusion

The SCA attacks are real and bypasses the mathematical security of the algorithms and exploits the implementation vulnerabilities. The Power and EM Side channel attacks are the major threat based on the possibility and practicality of the attack. The DPA is a widely studied attack by the researchers and the major countermeasure used is masking. As guarding involves the architectural change, masking is useful for existing software and some hardware implementations of the cryptographic modules. Masking relies heavily on adding the randomness and reversibly removing it before transmitting the data to thwart the adversary to ascertain the actual data and key being processed. Machine Learning and Deep Learning techniques are used to remove the masking to render them ineffective. Thus, researchers have proposed second order and higher order masking schemes that again are prone to be threatened by the ML and DL techniques. Re-keying is another dimension to hide the master key by using the derived key for actual encryption only for a specified period of time to make the ML and DL DPA technique efforts futile by limiting the number of traces available for the given derived key. However, this scheme involves sharing some random number or secret before changing the key over the secure channel. Hence, there is an overhead of sharing through a secure channel. On the other side, the masking techniques have overhead of adding and removing the masks at both the sender and receiver end. After careful qualitative analysis, both techniques wish to extend the guessing entropy values. As a novel intervention, re-keying avalanche has been discussed in this paper that can become a basis for statistically securing the master key in the re-keying approach.

Theoretically, combining both the techniques can help the masking with re-keying reduce the overhead in terms of frequency of sharing the new secret and the masking reduce the order of masking to decrease the guessing entropy as both techniques are independent of each other. The future direction for this research can be towards examining the combined strategy quantitatively to measure the effectiveness with respect to the parameters guessing entropy, total timing overhead, total space overhead, and the attack order protected against.

References

[1] R. Spreitzer, V. Moonsamy, T. Korak, and S. Mangard, “Systematic Classification of Side-Channel Attacks: A Case Study for Mobile Devices,” IEEE Communications Surveys and Tutorials, vol. 20, no. 1, pp. 465–488, Jan. 2018, doi: 10.1109/COMST.2017.2779824.

[2] Y. Zhou and D. Feng, “Side-Channel Attacks: Ten Years After Its Publication and the Impacts on Cryptographic Module Security Testing ♠.”

[3] M. Randolph and W. Diehl, “Power side-channel attack analysis: A review of 20 years of study for the layman,” Cryptography, vol. 4, no. 2. MDPI AG, pp. 1–33, Jun. 01, 2020. doi: 10.3390/cryptography4020015.

[4] P. Kocher, J. Jaaee, and B. Jun, “Diaeerential Power Analysis.” [Online]. Available: http:èèwww.cryptography.com

[5] Sayakkara, N. A. Le-Khac, and M. Scanlon, “A survey of electromagnetic side-channel attacks and discussion on their case-progressing potential for digital forensics,” Digital Investigation, vol. 29. Elsevier Ltd, pp. 43–54, Jun. 01, 2019. doi: 10.1016/j.diin.2019.03.002.

[6] G. Camurati, S. Poeplau, M. Muench, T. Hayes, and A. Francillon, “Screaming channels: When electromagnetic side channels meet radio transceivers,” in Proceedings of the ACM Conference on Computer and Communications Security, Association for Computing Machinery, Oct. 2018, pp. 163–177. doi: 10.1145/3243734.3243802.

[7] G. Camurati, A. Francillon, and F.-X. Standaert, “Understanding Screaming Channels: From a Detailed Analysis to Improved Attacks,” vol. 2020, no. 3, pp. 358–401, 2020, doi: 10.13154/tches.v2020.i3.358-401.

[8] P. Kocher, J. Jaffe, B. Jun, and P. Rohatgi, “Introduction to differential power analysis,” Journal of Cryptographic Engineering, vol. 1, no. 1, pp. 5–27, Apr. 2011, doi: 10.1007/s13389-011-0006-y.

[9] K. Ramezanpour, P. Ampadu, and W. Diehl, "SCAUL: Power Side-Channel Analysis with Unsupervised Learning," Jan. 2020, [Online]. Available: http://arxiv.org/abs/2001.05951

[10] R. Benadjila, E. Prouff, R. Strullu, E. Cagli, and C. Dumas, "Study of Deep Learning Techniques for Side-Channel Analysis and Introduction to ASCAD Database-Long Paper."

[11] D. Guo, K. Chen, X. Hu, Y. Wei, and J. Li, "A survey of prototype side-channel attacks based on machine learning algorithms for cryptographic chips," in Journal of Physics: Conference Series, Institute of Physics Publishing, Mar. 2019. doi: 10.1088/1742-6596/1176/3/032005.

[12] O. Choudary and M. G. Kuhn, "Efficient template attacks," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Springer Verlag, 2014, pp. 253–270. doi: 10.1007/978-3-319-08302-5_17.

[13] O. Choudary and M. G. Kuhn, "Template attacks on different devices," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Springer Verlag, 2014, pp. 179–198. doi: 10.1007/978-3-319-10175-0_13.

[14] "Efficient, Portable Template Attacks," IEEE Transactions on Information Forensics and Security, vol. 13, no. 2, pp. 490–501, Feb. 2018, doi: 10.1109/TIFS.2017.2757440.

[15] M. A. Elaabid and S. Guilley, "Portability of templates," Journal of Cryptographic Engineering, vol. 2, no. 1, pp. 63–74, May 2012, doi: 10.1007/s13389-012-0030-6.

[16] H. Maghrebi, T. Portigliatti, and E. Prouff, "Breaking Cryptographic Implementations Using Deep Learning Techniques."

[17] B. Hettwer, S. Gehrer, and T. Güneysu, "Applications of machine learning techniques in side-channel attacks: a survey," Journal of Cryptographic Engineering, vol. 10, no. 2, pp. 135–162, Jun. 2020, doi: 10.1007/s13389-019-00212-8.

[18] E. Prouff and M. Rivain, "Masking against Side-Channel Attacks: a Formal Security Proof."

[19] O. Reparaz, "Analysis and Design of Masking Schemes for Secure Cryptographic Implementations," 2016.

[20] M.-L. Akkar and C. Giraud, "An Implementation of DES and AES, Secure against Some Attacks."

[21] D. Canright and L. Batina, "A Very Compact 'Perfectly Masked' S-Box for AES (corrected)."

[22] E. Oswald, S. Mangard, N. Pramstaller, and V. Rijmen, "A Side-Channel Analysis Resistant Description of the AES S-box."

[23] B. Zakeri, M. Salmasizadeh, A. Moradi, M. Tabandeh, and M. T. M. Shalmani, "Compact and Secure Design of Masked AES S-Box."

[24] S. Chari, C. S. Jutla, J. R. Rao, P. Rohatgi, and I. B. M. Thomas, "Towards Sound Approaches to Counteract Power-Analysis Attacks."

[25] Valiveti and S. Vivek, "Second-order masked lookup table compression scheme," IACR Transactions on Cryptographic Hardware and Embedded Systems, vol. 2020, no. 4, pp. 129–153, 2020, doi: 10.13154/tches.v2020.i4.129-153.

[26] J.-S. Coron, "LNCS 8441 - Higher Order Masking of Look-Up Tables," 2014.

[27] L. Genelle, E. Prouff, and M. Quisquater, "Thwarting Higher-Order Side Channel Analysis with Additive and Multiplicative Maskings."

[28] M. Rivain and E. Prouff, "Provably Secure Higher-Order Masking of AES."

[29] G. Fumaroli, A. Martinelli, E. Prouff, and M. Rivain, "Affine Masking against Higher-Order Side Channel Analysis."

[30] Y. Komano and S. Hirose, "Re-keying scheme revisited: Security model and instantiations," Applied Sciences (Switzerland), vol. 9, no. 5, 2019, doi: 10.3390/app9051002.

[31] M. Abdalla and M. Bellare, "Increasing the Lifetime of a Key: A Comparative Analysis of the Security of Re-Keying Techniques," Springer-Verlag, 2000.

[32] E. K. Alekseev, L. R. Akhmetzyanova, I. B. Oshkin, and S. v Smyshlyaev, "On internal re-keying."

[33] C. Dobraunig, M. Eichlseder, S. Mangard, and F. Mendel, "On the Security of Fresh Re-keying to Counteract Side-Channel and Fault Attacks."

[34] M. Medwed, F.-X. Standaert, J. Großschädl, and F. Regazzoni, "Fresh Re-Keying: Security against Side-Channel and Fault Attacks for Low-Cost Devices."

[35] C. Dobraunig, F. Koeune, S. Mangard, F. Mendel, and F.-X. Standaert, "Towards Fresh and Hybrid Re-Keying Schemes with Beyond Birthday Security."

[36] M. Liskov, R. L. Rivest, and D. Wagner, "Tweakable Block Ciphers."