Holistic Power Side-Channel Leakage Assessment:

Towards a Robust Multidimensional Metric

Alric Althoff, Jeremy Blackstone, Ryan Kastner
University of California, San Diego
{aalthoff,jblackst,kastner}@eng.ucsd.edu

ABSTRACT

For many devices, power side-channel attacks are an effective means of obtaining secret keys from cryptographic algorithms. Recently, methods have been proposed to assess the vulnerability of devices to these attacks. While existing approaches effectively evaluate device vulnerability to attacks at specific points during execution, they do not consider the power measurement vectors holistically, using all time points in the measurement. This is necessary in order to accurately assess resistance to multi-target attacks. In this work, we identify characteristics of an ideal holistic side-channel security metric and develop a metric under these criteria. We demonstrate that our approach correctly ranks different FPGA implementations of AES with respect to attack difficulty.

1 INTRODUCTION

Attacks on cryptographic hardware using side-channel information are often fast and simple to execute. These side-channel attacks (SCAs) are well-studied and used widely across both industry and academia. Conversely, mitigation strategies are expensive and time consuming since they rely on specialized hardware design techniques or complex algorithmic modifications [16, 24]. Unfortunately, even after the implementation of a mitigation scheme, it is difficult to assess its effectiveness and correctness. This has motivated NIST and ISO/IEC to solicit recommendations for side-channel leakage assessment (SCLA) metrics and propose standards such as ISO/IEC 17825:2016 [7, 11].

Mean Traces to Disclosure (MTD) [24] is one commonly used metric. This metric is fundamentally tied to the employed attack method(s). Attacks are improving rapidly and becoming less expensive to execute. This means that a robust SCLA metric should not be tailored to detect vulnerabilities based on a specific attack type—or even a group of attack types. Instead, as pointed out in [23], a metric should indicate whether information about secret data is leaking through the tested side-channel for a particular device under test (DUT) in a way not linked to a particular power model or type of attack.

The Test Vector Leakage Assessment (TVLA) from Cryptography Research Inc. [1] makes strides in this direction by removing the requirement for a model of power consumption. However, it is univariate, i.e., it only considers one sample at a time. Multivariate, or “higher-order,” modes of the TVLA exist, but these violate the assumptions of the underlying statistic and provide uncertain conclusions [22]. While many attack techniques are univariate (DPA, CPA, template attacks), very powerful multivariate attacks have been developed [4, 9, 14] and we expect this trend to continue. A metric that detects vulnerability to these multi-target attacks must be *holistic* in the sense that it should consider all time points *and their interdependencies.* Additionally, TVLA assumes measurements are Gaussian and independent. Making assumptions about the underlying distribution of the power trace data may be statistically invalid when these assumptions don’t hold. Thus, an ideal metric would avoid such assumptions.

Such a metric should also provide a numeric score so that hardware vendors and end users can rank devices with finer granularity and make more informed decisions. This means we would prefer methods giving comparable numeric values to a “pass/fail” qualitative indicator of vulnerability. Additionally, a practical metric would have modest data requirements and be quick to execute so that tests, and testing labs, can be less expensive to run. In line with this requirement, a metric should make *valid* confidence intervals available so that a quick estimate can be differentiated from a thorough analysis, thus allowing security grades to be qualitatively compared.

In this work, we discuss at more depth the limitations of current tests. We provide characteristics of a robust and holistic SCLA metric. Then, we develop a SCLA metric that addresses each of these factors. Our metric, the Holistic Assessment Criterion (HAC), is nonlinear, holistic, and as assumption-free as practically possible, and comes with strong mathematical guarantees of consistency.

In summary, this work

- Intuitively describes and experimentally demonstrates the limitations of existing single-dimensional and higher order SCLA metrics.
- Provides a framework for robust multidimensional metrics.
- Introduces a new robust and model-free holistic SCLA metric within this framework.
- Evaluates our metric on power traces from different AES implementations, and shows that it correlates with TVLA and attack results.

This paper is organized as follows, Section 2 defines the power SCA threat model and implications for SCLAs, Section 3 discusses current SCLAs, Section 4 gives theoretical, synthetic, and real-life examples of situations that holistic SCLAs should be sensitive to. In Section 5, we introduce our framework for an ideal holistic SCLA, and in Section 6 we develop an algorithm within this framework. In Section 7 we verify our metric experimentally. Section 8 concludes the paper.

2 THREAT MODEL FOR POWER SCA

We assume an attacker can cause security critical programs to run with arbitrary inputs and collect detailed measurements $\mathcal{M}$ of device power use from a specific device under test (DUT). For example, the attacker could write a program which calls either a library function, or cryptographic hardware, to encrypt their chosen data while gathering voltage measurements over time with a connected

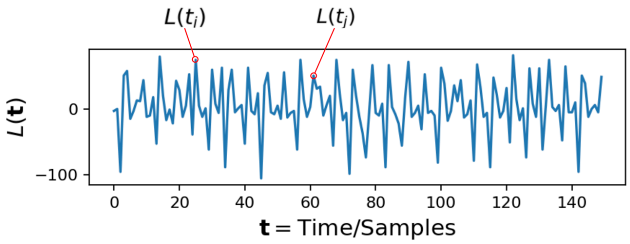

Figure 1: An example power trace from DPA Contest v4.2 [8]. $L(t_i)$ and $L(t_j)$ are leakage at different time indices. Univariate SCLAs (and attacks) only use leakage at one time index, multivariate SCLAs (and multi-target attacks) use more than one, while a holistic test (our approach) uses any subset of the trace, and can use the full trace. Practical attacks often use at least several hundred or thousand traces.

oscilloscope. We additionally assume that the attacker can synchronize these measurements so that all times $\mathbf{t} = [t_1, t_2, \ldots, t_n]$ during computation are aligned across runs, and that she can attack any or all of these points of interest (POI). She also knows precisely when the start of the computation occurs, e.g., by using a simple power analysis [12]. These are common assumptions across power analysis attacks.

With respect to SCLA, we assume that the equipment used to record the power traces, or "leakage", $L(\mathbf{t}) \in \mathcal{M}$ during a leakage assessment is at least as good as that available to an attacker, i.e., with comparable or greater sampling rate, bit depth, and noise characteristics. We note that while relatively low-cost equipment is often used for power analysis attacks [12], a leakage assessment should endeavour to equal or supersede the attacker's capabilities. We also assume the assessor has sampled traces in a representative manner. Our sampling method is discussed in Section 7. Figure 1 shows an example power trace that would be collected for attack.

Our technique focuses on leakage assessment for power SCA. Other side-channels (timing, acoustic, RF, etc.) are outside of our threat model, although this technique is likely to be much more broadly useful.

3 RELATED WORK

The goal of any empirical assessment of leakage is to determine if it is statistically possible to correctly classify disjoint groups of traces, $D_0 \subset \mathcal{M}$ and $D_1 \subset \mathcal{M}$, corresponding to different secret keys. Many techniques [2, 6, 15, 25] exist to detect information leakage, but all of these approaches are either univariate (testing a single time points one-by-one), require explicit selection of a small subset of time points, have onerous data requirements, or offer indefinite conclusions. We will discuss these points in Section 4.

The focus of recent work in univariate SCLAs [19, 25], with multivariate extensions [20], is the TVLA [1]. It is an attractive alternative to MTD because it doesn't emphasize a particular attack or power model. In this section, we describe it and discuss its benefits and drawbacks. While variations on the TVLA test have been proposed, (other univariate hypothesis tests in particular,) these share many or all of the shortcomings of the TVLA, making it a good example for discussion.

3.1 Test Vector Leakage Assessment (TVLA)

A typical TVLA [1] for use with the Advanced Encryption Standard (AES) begins with the collection of many power traces from the device under test for either fixed or random plaintexts[1] divided into two different groups, D_0 and D_1, described in [1]. Then the assessor will perform a two-tailed Welch's t-test on particular points of interest (POIs) in time to determine if there is a statistically significant difference between the traces in D_0 and those in D_1 at those POIs. If $\bar{x}_i$ is the sample mean of traces at a single POI, s_i is a sample standard deviation, and N_i is the number of traces taken in D_i, the t-statistic is

$$t = \frac{\bar{x}_0 - \bar{x}_1}{\sqrt{s_0^2/N_0 + s_1^2/N_1}} \tag{1}$$

Assuming that the data are independently drawn from Gaussian distributions with unknown variance, we may use t to determine the probability that these two Gaussian distributions have equal means μ_1 and μ_2. This probability is computed under the Student's t distribution parameterized by the degrees of freedom d.f.. The formula commonly used to approximate d.f. is well-known [20], and available in many statistical software packages, so we will not reproduce it here.

The TVLA assumes that the DUT is secure at a particular time index if the t-test fails to reject the null hypothesis for a confidence level α equivalent to 99.999%. The assessor is free to check any time index in the sampled power traces where secret data may be used, with algorithm-specific recommendations such as during S-Box access in the first or last round of AES.

It is also used to validate higher order statistical moments, i.e., variance, skewness, kurtosis, etc. However, estimation of these higher order moments is very sensitive to noise, time-consuming, and rather numerically unstable. Our approach negates this issue by avoiding higher moment-based statistics entirely. They are not necessary in non-parametric holistic testing, and as we will show in Section 4, may even lead to incorrect conclusions about security of the DUT.

3.2 Multivariate Combining

An extension to multiple variables is explored for attacks in [17] and used in the TVLA assessment regimen of [1, 20]. The method combines leakage from multiple time indices via a combination function and then performs a univariate assessment as before with a somewhat higher threshold. The optimal combiner analyzed in [17] computes the centered product of the samples at the POI, that is

$$L(t') = \prod_{t \in \mathrm{POI}} L(t) - \bar{x}(t) \tag{2}$$

such that the resulting collection of $L(t')$ acts as pseudo-traces for the collection of times in the set of POI. The corresponding test then uses moments computed over this pseudo-trace set.

[1] It may seem counterintuitive that keys do not always differ between groups, see Section 7 for details.

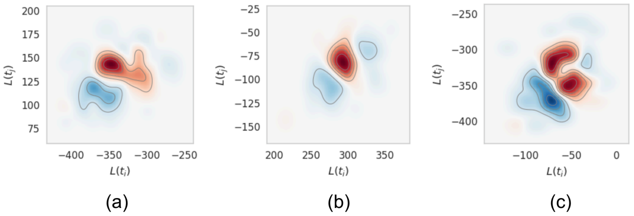

Figure 2: Three examples of nonlinear bivariate leakage in real measurements. $L(t_i)$ **and** $L(t_j)$ **of (a), (b), and (c) are from power traces where times** t_i **and** t_j **are rated secure by TVLA and the multivariate extension, but a nonlinear SVM classifies the traces into correct key distributions with over 70% accuracy. Red corresponds to the distribution of key 1 and blue to key 2. Such examples are not uncommon.**

In order to combine time indices we must first identify the indices we wish to test. This requires another stage of analysis resembling what in machine learning and statistics is called *feature selection*. POI determination for SCLA has a critical difference: where feature selection *minimizes* the size of the set of indices that we need to use, it discards redundant features, in the context of power SCA an attacker would find redundant indices equally exploitable. Therefore, we must consider time indices that are redundant to be equally leaky. Intuitively, this means that our identification of POI should also allow us to label redundant time indices, and we are unlikely to know this if we choose POI a priori for a previously un-analyzed piece of hardware. So multivariate assessments that *do not* operate holistically, on every sample in the trace, should include a technique for selecting these POI. The examples in Section 4 suggest that a POI selection algorithm for a multivariate SCLA should also be multivariate, and this is a complex problem in itself.

This being said, **our process for holistic testing does not require identifying POI at all**. Because holistic testing checks all POI at once, we never need to solve the problem of identifying time indices to test. However, that doesn't mean that this is a limitation of holistic testing; a set of POI could certainly be tested using a holistic test regime, but we believe it is important to create security metrics that require *minimal* a priori assumptions, even assumptions about where an attacker may strike.

As an example of how such assumptions may harm attack mitigation efforts, consider S-Box masking [16] for AES. Masking has been designed to protect the most common attack POI in AES, however, as we can see in the results of DPA Contest v4.2 [8], which invites researchers to attack AES with a masked S-Box implementation, a successful attack can be performed by relocating the attack target to a different POI. One attack shown in the contest results does just that, moving their POI to ShiftRows and recovering the entire key.

4 SENSITIVITY REQUIREMENTS FOR ROBUST SCLA

In this section we will identify what types of information leakage an SCLA should be able to detect, and what statistical pitfalls an SCLA should avoid. First, we demonstrate that any SCLA that is *robust* to attacker ingenuity or future algorithmic and hardware developments must consider multivariate leakage *jointly*.

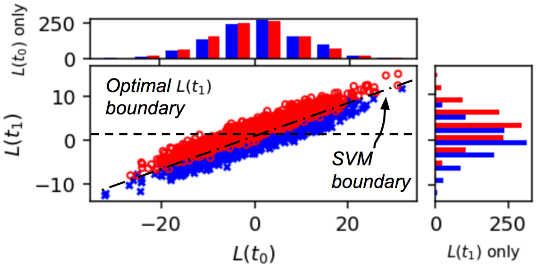

Figure 3: A case of underestimation for univariate security metrics. Considering the $L(t_0)$ **and** $L(t_1)$ **axes jointly is more powerful than either one alone. We may not have known that** $L(t_0)$ **was relevant without testing all pairs of samples. See Section 4 for details.**

4.1 Motivating Examples

Multivariate leakage exists. This is evidenced by the increased effectiveness of multivariate attacks such as [4, 9, 14]. What is less clear is that this type of leakage is also *quite likely*. In this section we will show some examples of multivariate leakage, and explain why two (or more) points can be considerably more powerful when considered together.

Figure 2 shows distributions from power traces from an FPGA AES core. Figure 2(a), 2(b), and 2(c) are bivariate kernel densities of voltage measurements taken from different pairs of time points (i, j). Colors correspond to different keys. Each of these bivariate distributions is rated secure[2] by TVLA and also by the multivariate combining method of [1, 20], but a nonlinear SVM classifies them each with a cross-validated accuracy greater than 70%. So, if we attack one of either time point t_i or t_j, with power traces $L(t_i)$ or $L(t_j)$, or even if we consider them at the same time, but independently, we would have a harder time attacking them than if we consider them jointly.

In statistics and machine learning this type of interaction is called *variable complementarity* and, as we can see, certainly exists in trace samples at different points in time. Additionally, it is possible for complementarity to exist and yet be *statistically impossible to detect* using any moment-based statistics, or even those based off of statistical independence between keys and traces!

This may seem surprising, but consider the following example: for Boolean x and y, x XOR $y = z$, but knowledge of x alone does not reveal any information about z (x and z are statistically independent), and neither does knowledge of y, however, knowing both completely determines z.

Replace z by the secret key, and x and y by power trace measurements, and we can see that this example implies that it is theoretically possible to have a power measurement $L(t_i)$ which under the *best* univariate metric at *all statistical moments* will indicate

[2] 4.5σ implies a p-value $< 10^{-5}$, 2(a), 2(b), and 2(c) have t-test p-values > 0.001 for both TVLA and TVLA with multivariate combining.

that there is little to no information in the traces about the secret key, but by considering additional samples at some other time t_j all key bits may be recovered. This example also applies to univariate tests of univariate statistical independence and mutual information. This implies that neither of these criteria are sufficient to detect all forms of *clearly exploitable* multivariate leakage.

Figure 3 shows another synthetic example that is closer to what we might see in practice. Assume that the red Os and blue Xs are leakage measurements $L(t_0)$ and $L(t_1)$ at two time indices t_0 and t_1. Further, assume that the blue Xs and red Os are measurements taken during a cryptographic operation where only the key differs. The univariate t-test on $L(t_1)$ between the distribution of Xs and Os differ significantly, with a p-value of 2×10^{-53} that the distributions have means that are equal. A Bayes-optimal classification using only $L(t_0)$ data (top histogram) has a success rate of 64%. A similar t-test on time index $L(t_1)$ gives a p-value of 0.07, which passes—i.e. is not flagged as "leaky"—due to the TVLA detection threshold of $p < 10^{-5}$ recommended in [1]. **However, *if* $L(t_0)$ *and* $L(t_1)$ *are used together,* a decision boundary learned by a linear support vector machine (SVM) achieves a classification accuracy of 91%.** This implies that a multi-target or higher order attack using power trace data from two time indices with a joint probability distribution resembling $\mathbb{P}(L(t_0), L(t_1))$ would be considerably faster and have a higher success rate than if the $L(t_0)$ data were used alone. It is possible for this to happen in higher dimensions than just two, e.g., nine time points could seem individually useless, but when combined with a tenth, it could increase the attackability of the DUT by a large amount.

Our security metric (Section 6) successfully detects information leakage in these scenarios.

4.2 Common SCLA Issues

Multiple Comparisons. Many tests use a threshold α for the p-value that is not changed regardless of the number of tests conducted. While this leads to an issue that applies to any hypothesis test conducted multiple times, we will use the TVLA as an example.

To see the problem, consider that the TVLA requires us to conduct a separate t-test for each POI, and each of these POI have probability 10^{-5} of exceeding the threshold *if the null hypothesis is true*, i.e. the means of the presumptive underlying Gaussian distributions are the same. However, if we test each sample in the entire trace, then for traces more than 10^5 samples long we should expect for the test to reject the null hypothesis at least once by chance alone. For this reason an uncorrected test procedure is vulnerable to false "insecure" ratings when multiple POIs are tested.

Recently researchers have made mathematically sound efforts to address this. For example, [25] applies a goodness of fit test to the p-values from the t-test at each POI in the trace, and rejects the null hypothesis of DUT security if this p-value distribution is non-uniform. Unfortunately, even though a few recent works have begun mentioning this issue e.g. [15], it remains unimplemented in the analysis. We stress that if correction is not used during multiple testing, the p-values should not be considered correct; they no longer represent the probability of a false reject.

Assumption Violations when Testing Hypotheses. The multivariate combining technique in Section 3.2 is an assumption violation for the t-test: Even if the samples for all POIs are Gaussian and independent the product distribution of these random variables is non-Gaussian. As written, Eqn. (2) does not correct the distribution of $L(t')$ such that the p-value returned by a t-test is the probability of a false rejection of the null hypothesis; because t-statistics derived from these data will not have a Student's t distribution, derived p-values will be heuristics. This is a particular concern when using these "p-values" as a *comparable* indicator of certainty in the result.

Consider too that Eqn. 2 was analyzed in [17] in the context of second order CPA, where a combination function is part of the attack method. It is optimal in the sense that it is the best in terms of a Hamming weight power model when using Pearson's correlation to distinguish between differing keys when a combining function is required. This does not imply optimality or suitability in general.

Poorly Defined Test Criteria. If the SCLA is sensitive to criteria that are too weak to indicate security, then we run the risk of falsely labeling a device secure when it is not. The opposite can be damaging as well: A test that is sensitive to factors *beyond* DUT security can cause practitioners to waste time and effort attempting to fix issues that are artifacts of the SCLA method but unrelated to security of the DUT.

To make this point more clear, consider a bivariate Gaussian distribution with random variables (X, Y) where X and Y taken alone, (i.e. their marginals,) have equal means and standard deviations. If we are given samples of X and Y and asked to discern the most likely Y given X, we might inspect their correlation to determine a likely region for Y. However, if we replace Y with plaintext inputs, and X with power traces, it seems odd that we should be able to determine the *key* from the relationship of X and Y.

Yet this is exactly the protocol applied by many tests [1]. The reasoning for this fixed-vs.-random input (plaintext) protocol is that certain attacks (e.g. CPA) require knowledge of a known set of plaintexts or ciphertexts, and predictable variations in the traces due to these values enables the attack. However, in the general case, dependence on the plaintexts does not matter (e.g. [4] represents a family of attacks that do not require knowledge of plaintexts), and we do not want holistic testing to be tied to certain attack techniques.

This example implies that it is entirely possible to develop an *effective* SCA countermeasure that alters statistical dependence between traces and plaintexts arbitrarily. SCLAs relying on tests of these quantities would inappropriately label a design as insecure after the mitigation is implemented. The formal definitions in Section 5.1 and metric framework we supply in this paper do not have this limitation.

While the work of [22] mentions, with specific reference to the t-test, several of the issues that we have pointed out, e.g. there is an inappropriate risk of false positives and negatives, and notes the multiple comparison issue, in the end, many of their points and assumptions are specific to AES S-Box masking. Side-channel attacks, hardware architectures, and new algorithms are being developed rapidly, and so we would like to address the overall problem. Therefore, in this work we are not developing a framework or an algorithm that focuses on any implementation specifically. Instead, we propose that metrics address the overarching problem:

measuring the complexity of the hypothesis class necessary to handle particular massively multi-class classification problems using feature vectors with very high dimension.

5 A FRAMEWORK FOR HOLISTIC SCLA

In this section we will introduce a framework for holistic—considering the entire trace at once—and robust SCLA. The main components of this framework are a null hypothesis which assumes vulnerability given a compact but well-founded definition of exploitable leakage, non-parametric confidence intervals, and a minimum of implicit assumptions influencing the resulting scores.

5.1 Baseline of Vulnerability

We do not assume that a device is secure prior to having tested it, and this is rarely, if ever, a reasonable assumption for a DUT pulled off a shelf at random. Thus, a holistic SCLA should begin in the state of "insecure/unmitigated" with respect to an unknown DUT and work to collect evidence to the contrary. This is a philosophical, rather than mathematical, departure from other SCLAs, including the TVLA, for which the null hypothesis is that the device is secure[3]. This testing philosophy should lead us to a more stable and repeatable application of a device security grade.

In order to formalize our intuition of "vulnerability" as key distinguishability given measurements, we need to define the type of leakage we are testing for:

Definition 1 (Exploitable Leakage). *If there exists for some key distributions $\mathcal{K}_0$ and $\mathcal{K}_1$, and measurements $X_0 \sim \mathcal{K}_0$ and $X_1 \sim \mathcal{K}_1$ where $(X_0, X_1) \overset{d}{\neq} (X_1, X_0)$ then the pair of key distributions $\mathcal{K}_0$ and $\mathcal{K}_1$ have exploitable leakage > 0 where $\overset{d}{=}$ denotes identical probability distribution.*

In other words, devices leak secret information if the random variables underlying the power traces are not *exchangeable* with respect to different secrets.

Definition 2 (Exchangeable Random Variables [5]). *If a tuple of random variables $(X_0, \ldots, X_n) \overset{d}{=} (X_{\pi(0)}, \ldots, X_{\pi(n)})$ for arbitrary permutations π then the random variables X_i, $1 \leq i \leq n$, are exchangeable.*

This means that for n exchangeable random variables, all $n!$ permutations have the same joint distribution. With respect to power traces and secret keys, this means that we cannot organize traces into bins corresponding to their most likely key with an accuracy better than chance. Exchangeability is a strictly stronger property than identical distribution, (i.e. all exchangeable random variables are i.d., but not the reverse,) and so detecting this property also indicates when measurements are classifiable by most likely distribution It is also strictly weaker than independence, which we have already shown by counterexample is too strong.

This logic allows us to state that **the baseline state, or null hypothesis H_0, of a holistic SCLA should be that traces are not exchangeable over secrets**. This stems directly from the definition; if it is usually possible to differentiate power traces measured when using different secret keys, then the DUT is vulnerable to power attacks.

5.2 Nonparametric Confidence Intervals

Statistically derived values always have an associated uncertainty. This can either be due to noise or non-representative sampling. Any security metric should be accompanied by a confidence interval such that—for a fixed probability α—the true value of the statistic lie within the range. This will allow us to say something about the range in which the true value of the statistic may lie.

In addition, an ideal confidence interval would not make assumptions about measurements, models, or the relationship between measurements and latent variables unless this is justified. Specifically, when deriving confidence bounds we should not assume that measurement noise is Gaussian, or even that it is "noise" that cannot be perfectly deconvolved using known processor state.

5.3 Lack of a Priori Assumptions

t-tests assume that the data are Gaussian and that samples are independent—things that are very unlikely to be true for power traces in general [21]. In the case of a holistic SCLA, we would like our test to be valid across all known—and ideally, unknown—hardware types. This prohibits us from using tools that rely on untested assumptions about the nature of the measurements. A holistic SCLA metric should be as **assumption-free (nonparametric) as practically possible** in order to be comparable across devices and robust to changing technologies and attack vectors.

While it is common practice in statistics to take advantage of assumptions that have *empirical support given the sample in question*, there is a danger of false inference when imposing these assumptions on situations where they do not apply. We are not making a case against assumptions in general, but we do feel that we should have high *empirical* confidence that they are true for specific data before relying on them.

6 OUR PROPOSED SCLA METRIC

Now that we have identified the criteria for a holistic SCLA, we are in position to derive a metric in accordance with these principles.

6.1 A Holistic Assessment Criterion

Consider two sets, D_0 and D_1, consisting of N traces each, where each trace is M samples long. Our algorithm, the Holistic Assessment Criterion (HAC), is motivated by the fact that if the sampled traces in D_0 and D_1 are identically distributed then the k^{th} nearest neighbor of any trace picked at random from D_0 (resp. D_1) will have an equal probability of being from D_0 as D_1. To formalize the intuition,

Theorem 6.1. *Let A and B be composed of samples from the random variables X and Y respectively. If X and Y are identically distributed, then the nearest neighbor $y \in A \cup B$ of any $x \in A$ is a member of A or B with equal probability.*

Proof (informal). Identically distributed random variables X and Y, with samples x and y, obey $\mathbb{P}(x \in C) = \mathbb{P}(y \in C)$ for all subsets C of the sample space. This implies that, for some open region of the sample space, the probability of a sample landing

[3] The null hypothesis, H_0, of the t-test is that the means of the of the two groups are identical, which would imply that the DUT is secure.

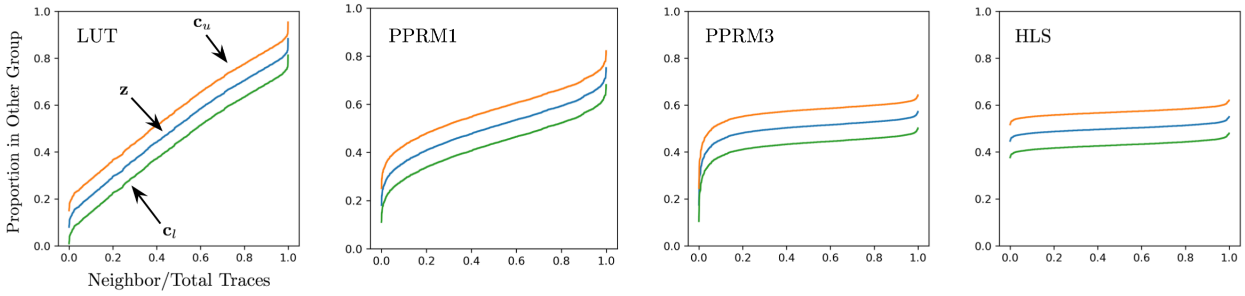

Figure 4: HAC plots of the four FPGA architectures tested in Section 7. The horizontal axis of each plot is the fraction of the total number of columns in Q (x-axis 0.0 representing the first neighbors, and 1.0 representing the farthest neighbors), and the vertical axis is the proportion of joint traces from $\mathbf{L}_0\|\mathbf{L}_1$ whose k^{th} neighbors are in $\mathbf{L}_1\|\mathbf{L}_0$ instead of $\mathbf{L}_0\|\mathbf{L}_1$. If the blue/center line (z from Algorithm 1) has a greater average slope, then the groups of traces in D_0 and D_1 are less exchangeable, and the DUT is less secure. The orange/upper and green/lower lines are the $\alpha = 10^{-4}$ confidence interval $[\mathbf{c}_l, \mathbf{c}_u]$ about the center line. That is, we may be sure that the true proportion lies within the bounded region with probability $1 - 10^{-4}$.

in that region is *identical* for X and Y. Further assume that this region is a δ-thin band at distance ρ from another sample, that is, $(\rho - \delta, \rho + \delta)$, where $\lim_{\delta \to 0}$. Earlier definitions imply that if X and Y have identical distributions, then a sample of either X or Y will have equal probability of being in this band. Notice that this is true regardless of the distance function we use, so long as it is a well-defined distance metric on all C. □

This means that if sets A and B of cardinality n are identically distributed, then a Boolean indicator of membership in B for the list of $1 \leq k^{th} \leq 2n - 1$ nearest neighbors to all points in $A \cup B$ would have a Bernoulli($p = 1/2$) distribution, and so the sum of these indicators would have a binomial($2n - 1, p = 1/2$) distribution.

This yields a powerful test that indicates whether two distributions are equal. Moreover, it does not make any assumptions about the distributions, and is fast and robust to data with high dimension M.

With this technique in place, the only thing left to do to verify exchangeability directly from Def. 2 is to consider the traces from D_0 and D_1 *jointly*, that is, with the traces appended end-to-end, then swap them, and check to see if the joint traces are identically distributed.

To restate this in precise terms, first, we will "stack" all the traces from D_x into two $N \times M$ matrices $\mathbf{L}_x$, where $x \in \{0, 1\}$. Then, we will concatenate these matrices, to form two $N \times 2M$ "joint trace" matrices $\mathbf{J}_{01} = \mathbf{L}_0\|\mathbf{L}_1$, and $\mathbf{J}_{10} = \mathbf{L}_1\|\mathbf{L}_0$, where $\|$ in this case signifies matrix concatenation. Note that if $\mathbf{J}_{01} \overset{d}{=} \mathbf{J}_{10}$, then the sets D_0 and D_1 are exchangeable. To this end, we can make use of Thm. 6.1.

Let $d(\mathbf{A}, \mathbf{B})$ be the matrix of pairwise distances between each row of $\mathbf{A}$ and all rows of $\mathbf{B}$. So $\mathbf{T} = d(\mathbf{A}, \mathbf{B})$ implies $\mathbf{T}_{i,j}$ is the distance between row i of $\mathbf{A}$, (which we write $\mathbf{A}_{i,:}$) and row j of $\mathbf{B}$, ($\mathbf{B}_{j,:}$). In our algorithms and experiments we use the Euclidean distance $d(x, y) = \|x - y\|_2$, but this is, as we can see in the proof of Thm. 6.1, one of many possible statistically equivalent choices. We use this distance function to compute the joint pairwise distance matrices,

$$\mathbf{T}_{\text{self}} = d(\mathbf{J}_{01}, \mathbf{J}_{01}) \tag{3}$$

$$\mathbf{T}_{\text{other}} = d(\mathbf{J}_{01}, \mathbf{J}_{10}) \tag{4}$$

$$\mathbf{C} = \mathbf{T}_{\text{self}}\|\mathbf{T}_{\text{other}} \tag{5}$$

We then identify the nearest neighbors in order of distance and mark the membership of these neighbors with respect to one of the joint matrices. That is, we sort each row of the concatenated distance matrix $\mathbf{C}$ in ascending order and form a new Boolean matrix $\mathbf{Q}$, letting $\mathbf{Q}_{i,k}$ be the entry of $\mathbf{Q}$ at the i^{th} row and k^{th} column, $\mathbf{Q}_{i,k} = 0$ if the k^{th} nearest neighbor of row i of $\mathbf{J}_{01}$ is some row of $\mathbf{J}_{01}$, and $\mathbf{Q}_{i,k} = 1$ if it is from $\mathbf{J}_{10}$.

This is a direct implementation of Def. 2, and Thm. 6.1: We now have joint variables in two permutations ($\mathbf{J}_{01}$ and $\mathbf{J}_{10}$), if they are identically distributed, then the variables are exchangeable. More specifically, we may infer that **the DUT is insecure** if the proportion, $\mathbf{z} = N^{-1} \sum_{i=1}^{N} \mathbf{Q}_{i,:}$ when sorted, has an average (arithmetic mean) slope greater than that predicted by the quantile function of a binomial distribution with parameters N = Number of traces in D_0, $p = 1/2$—by symmetry of distance, it is not necessary to form the "full" distance matrix. As we will demonstrate in Section 7, this slope is an indicator of overall attack difficulty. We describe our technique step-by-step in Algorithm 1.

Using this technique of examining the proportion of nearest neighbors we are able to detect exchangeability, up to limits imposed by noise, numerical issues, and the number of data points. While statistically drawn conclusions cannot constitute mathematical proof, we can be certain enough for comfort. To this end we propose the confidence intervals in Section 6.1.1.

6.1.1 Confidence Intervals on $\mathbf{z}$. We certify our result with a probabilistic bound on $\mathbf{z}$ for the reasons discussed in Section 5.2. Hoeffding's inequality [10] implies that a two-sided bound on the deviation of an empirical proportion $\hat{p}$ from the true proportion p decreases with the number of measurements N according to the rule,

$$\mathbb{P}(|\hat{p} - p| > \epsilon) \leq 2\exp(-2N\epsilon^2) \tag{6}$$

Algorithm 1: Holistic Assessment Criterion (HAC)

```
Data: Matrices $\mathbf{L}_0, \mathbf{L}_1 \in \mathbb{R}^{N \times M}$ of $N$ measurements, each $M$
      samples long. $\mathbf{L}_0$ contains traces from $D_0$, and $\mathbf{L}_1$ from
      $D_1$, and $\alpha$, specifying the confidence region, such that
      $\mathbf{z}$ will be within $[\mathbf{c}_l, \mathbf{c}_u]$ with confidence $1 - \alpha$.
Result: A sorted vector of proportions $\mathbf{z}$, with upper and
        lower confidence intervals $\mathbf{c}_u$ and $\mathbf{c}_l$, and the HAC
        slope (average slope of $\mathbf{z}$), for which larger values
        indicate more vulnerability of the DUT.
// Concatenate the trace matrices in both orders; 01 and 10
$\mathbf{J}_{01} \leftarrow \mathbf{L}_0 || \mathbf{L}_1$
$\mathbf{J}_{10} \leftarrow \mathbf{L}_1 || \mathbf{L}_0$
// Pairwise distances between rows of $\mathbf{J}_{01}$ other rows in $\mathbf{J}_{01}$
$\mathbf{T}_{\text{self}} \leftarrow d(\mathbf{J}_{01}, \mathbf{J}_{01})$
// Pairwise distances between rows of $\mathbf{J}_{01}$ and those of $\mathbf{J}_{10}$
$\mathbf{T}_{\text{other}} \leftarrow d(\mathbf{J}_{01}, \mathbf{J}_{10})$
// Concatenate the pairwise distance matrices
$\mathbf{C} \leftarrow \mathbf{T}_{\text{self}} || \mathbf{T}_{\text{other}}$
// Form the set-membership matrix $\mathbf{Q}$
forall $i \in [N]$ do
    $\mathbf{s} \leftarrow$ sortAscending($\mathbf{C}_{i,:}$)
    forall $k \in [2N]$ do
        if $s_k$ is from $T_{self}$ then
            $\mathbf{Q}_{i,k} \leftarrow 0$
        end
        else if $s_k$ is from $\mathbf{T}_{other}$ then
            $\mathbf{Q}_{i,k} \leftarrow 1$
        end
    end
end
// Sum up the columns of $\mathbf{Q}$
forall $k \in [2N]$ do
    $\mathbf{z}_k \leftarrow \frac{1}{N} \sum_{i=1}^{N} \mathbf{Q}_{i,k}$
end
// See Fig. 4 for plots of sorted $\mathbf{z}$ and confidence intervals
$\mathbf{z} \leftarrow$ sortAscending($\mathbf{z}$)
// By Hoeffding's inequality [10]
$\epsilon \leftarrow \sqrt{\log(2/\alpha)} / \sqrt{(2N)}$
$[\mathbf{c}_l, \mathbf{c}_u] \leftarrow [\mathbf{z} - \epsilon, \mathbf{z} + \epsilon]$
// HAC slopes are reported in Table 1
HAC slope $\leftarrow N/(N-1) \sum_{i=2}^{N} \mathbf{z}_i - \mathbf{z}_{i-1}$
```

this means that we can solve for a region where p will be with probability $1 - \alpha$ given $\hat{p}$ by setting

$$\epsilon = \sqrt{\log(2/\alpha)/(2N)} \tag{7}$$

and so

$$[\mathbf{c}_l, \mathbf{c}_u] = [\mathbf{z} - \epsilon, \mathbf{z} + \epsilon] \tag{8}$$

is a valid confidence interval for each p given the empirical proportions $\hat{p}$ in $\mathbf{z}$. $\mathbf{c}_u$ and $\mathbf{c}_l$ are shown for our experiments in the upper and lower lines of the Fig. 4 plots.

6.1.2 *Interpretation of HAC Results.*

First things first: For a DUT to be secure, the slope of $\mathbf{z}$ should be small, and confidence intervals should be narrow. Confidence interval width depends on measurement count.

We anticipate that the reader will question our approach. Why not use p-values from a test of $\mathbf{z}$ against the binomial distribution? We believe that a single p-value is too reductive. In our HAC, we leverage the fact that if the probability of set-membership is $1/2$, then the variance of the proportion will narrow for N measurements at a rate of $1/(4N)$. This implies that the slope of the non-decreasing sequence $\mathbf{z}$, the empirical quantile function of the proportion of other-set membership, should shrink at a rate of $1/(4N)$ as well, but will never quite reach **zero—the "perfect score" for a secure DUT**. We take the view that it is very likely *impossible* to *empirically prove* that a design is perfectly secure, however, we can have an overwhelming confidence in such a result by observation, and our approach supports this view.

Additionally, as we collect more data, the confidence intervals around $\mathbf{z}$ will narrow, and so we can use these to say when "enough is enough," either with regards to our confidence that a design is secure, or that it is insecure to a certain degree. This enables us to make informed statements about the **HAC ranking between designs**, which is the central goal of this work.

Also, our approach avoids the problems of multiple comparison, unverified model assumptions, inappropriate metric criteria, and hypothesis tests which only give only qualitative pass/fail indicators without a degree of precision. This is true *even though it does not reveal the most vulnerable points of interest (POI).* Though it is possible to apply HAC to sub-sections of the traces and conduct a search for highly vulnerable regions, we will leave that for future work.

7 EXPERIMENTS

To validate our metric we compare correlated power analysis (CPA) attack [3] results on four different FPGA architectures for AES-128 encryption. Fig. 4 shows HAC results $\mathbf{z}$, $\mathbf{c}_l$, and $\mathbf{c}_u$ from these designs (further details are in the caption). Table 1 gives a comparison of the HAC slope (the mean slope of $\mathbf{z}$ in Fig. 4) with CPA trace complexity (MTD) and TVLA $\max(-\log(p))$ values, which are accepted standards.

These four FPGA designs differ mainly in their method of obtaining S-Box values during the SubBytes step.

LUT stores the S-Box values in on-chip Block RAM. Each value is retrieved as-needed from memory. Retrieving key-dependent values from RAM is a well-known source of information leakage.

PPRM1 implements positive polarity Reed-Muller AND-XOR logic to compute the S-Box values at run time.

PPRM3 the same principles as PPRM1, but using three AND-XOR stages. Because this was designed for low-power systems, we should expect the signal-to-noise ratio (SNR) to be smaller (i.e. more noise) due to power efficiency of S-Box computations.

HLS was implemented in hardware via high-level synthesis from C++. It is pipelined and loops are all unrolled. S-Box values are stored in registers near the SubBytes functional

Table 1: Comparison of CPA mean trace count (MTD) for a successful attack, t-statistic (TVLA) $\max(-\log p\text{-values})$, and HAC slope (Alg. 1) values across AES implementations.

Impl.	CPA MTD	TVLA	HAC†
LUT	4K	96.5	0.80
PPRM1	30K	30.7	0.57
PPRM3	38K	32.3	0.40
HLS*	>100K	11.7	0.10

* Design remains unbroken even after several attack types.

† This work. Value is the mean slope of the HAC plots in Fig. 4.

units and accessed concurrently. It has very low SNR, and pipelining adds significantly to the noise.

Trace Collection Protocol. We acquired voltage-drop traces using a National Instruments PXIe-5186 oscilloscope from a SAKURA-G evaluation board [13], at a sampling rate of 1 GHz for all FPGA designs. The minimum trace length is over 8k samples, and the max is 32k samples. Lengths vary to ensure that the entire AES computation is captured. Triggering is controlled by an independent controller FPGA on the SAKURA-G board.

HAC scores are computed via Algorithm 1 run on 2k traces, 1k in each of the sets D_0 and D_1. The set D_0 corresponds to the "fixed-key" and D_1 to the "random-key" collected according to the fixed-vs.-random key regimen of the data collection protocol released in a report [18] by RAMBUS.

The protocol defined in [18] specifies that one set of data are gathered with a fixed key, and one with a pseudorandom key, with pseudorandom plaintexts generated for each set. The reason that we do not use the fixed-vs.-random input versions is because these do not necessarily reveal *key-dependent* information leakage. As demonstrated in Table 1, our method is predictive of attack complexity when using the fixed-vs.-random key protocol.

As we mention in Section 4.2, tests using groups with differing inputs (plaintexts) instead of keys do so under the assumption that you cannot conduct an attack unless plaintexts significantly alter the trace. However, consider that Gaussian template attacks [4] require *no* knowledge of the plaintexts used in the attack. For the purposes of constructing Gaussian templates, inputs are considered "noise" and generated pseudorandomly.

8 CONCLUSION

In this work we demonstrate the value of holistic SCLA testing. We additionally lay out criteria for an ideal holistic test and develop a nonlinear, nonparametric statistical metric, HAC, within this framework. To evaluate our approach, we test our technique on traces from four AES implementations and show successful vulnerability detection in accordance with both t-test and CPA attack results. As a bonus, our technique can be applied quite generally to assess the difficulty of massively multi-class high dimensional classification problems, and is novel to the best our knowledge.

It is our hope that this work will inspire further research, and lead to certification efforts examining the benefits of incorporating holistic testing in standardized SCLA procedures.

REFERENCES

[1] Becker, G., Cooper, J., DeMulder, E., Goodwill, G., Jaffe, J., Kenworthy, G., Kouzminov, T., Leiserson, A., Marson, M., Rohatgi, P., et al. Test vector leakage assessment (tvla) methodology in practice. In *International Cryptographic Module Conference* (2013), vol. 1001, p. 13.

[2] Bhasin, S., Danger, J.-L., Guilley, S., and Najm, Z. Nicv: normalized inter-class variance for detection of side-channel leakage. In *Electromagnetic Compatibility, Tokyo (EMC'14/Tokyo), 2014 International Symposium on* (2014), IEEE, pp. 310–313.

[3] Brier, E., Clavier, C., and Olivier, F. Correlation power analysis with a leakage model. In *International Workshop on Cryptographic Hardware and Embedded Systems* (2004), Springer, pp. 16–29.

[4] Chari, S., Rao, J. R., and Rohatgi, P. Template attacks. In *International Workshop on Cryptographic Hardware and Embedded Systems* (2002), Springer, pp. 13–28.

[5] Diaconis, P., and Freedman, D. Finite exchangeable sequences. *The Annals of Probability* (1980), 745–764.

[6] Durvaux, F., Standaert, F.-X., and Veyrat-Charvillon, N. How to certify the leakage of a chip? In *Annual International Conference on the Theory and Applications of Cryptographic Techniques* (2014), Springer, pp. 459–476.

[7] Easter, R. Side channel testing requirements in 19790. In *International Cryptographic Module Conference* (2016).

[8] et al., S. B. Dpa contest v4.2, 2015.

[9] Gierlichs, B., Batina, L., Preneel, B., and Verbauwhede, I. Revisiting higher-order dpa attacks. In *Cryptographers Track at the RSA Conference* (2010), Springer, pp. 221–234.

[10] Hoeffding, W. Probability inequalities for sums of bounded random variables. *Journal of the American statistical association 58*, 301 (1963), 13–30.

[11] ISO/IEC. 17825 - information technology - security techniques - non-invasive attack mitigation test metrics for cryptographic modules.

[12] Kocher, P., Jaffe, J., Jun, B., and Rohatgi, P. Introduction to differential power analysis. *Journal of Cryptographic Engineering* (2011), 1–23.

[13] Lab, S. Sakura hardware security project, 2014.

[14] Mather, L., Oswald, E., and Whitnall, C. Multi-target dpa attacks: Pushing dpa beyond the limits of a desktop computer. In *International Conference on the Theory and Application of Cryptology and Information Security* (2014), Springer, pp. 243–261.

[15] Moradi, A., Richter, B., Schneider, T., and Standaert, F.-X. Leakage detection with the χ^2-test. *IACR Transactions on Cryptographic Hardware and Embedded Systems 2018*, 1 (2018).

[16] Oswald, E., Mangard, S., Pramstaller, N., and Rijmen, V. A side-channel analysis resistant description of the aes s-box. In *Fast Software Encryption* (2005), Springer, pp. 199–228.

[17] Prouff, E., Rivain, M., and Bevan, R. Statistical analysis of second order differential power analysis. *IEEE Transactions on computers 58*, 6 (2009), 799–811.

[18] RAMBUS. Test vector leakage assessment (TVLA) derived test requirements (DTR) with AES. https://www.rambus.com/test-vector-leakage-assessment-tvla-derived-test-requirements-dtr-with-aes/, 2015.

[19] Reparaz, O., Gierlichs, B., and Verbauwhede, I. Fast leakage assessment. In *International Conference on Cryptographic Hardware and Embedded Systems* (2017), Springer, pp. 387–399.

[20] Schneider, T., and Moradi, A. Leakage assessment methodology. In *International Workshop on Cryptographic Hardware and Embedded Systems* (2015), Springer, pp. 495–513.

[21] Schneider, T., Moradi, A., Standaert, F.-X., and Güneysu, T. Bridging the gap: advanced tools for side-channel leakage estimation beyond gaussian templates and histograms. In *International Conference on Selected Areas in Cryptography* (2016), Springer, pp. 58–78.

[22] Standaert, F.-X. How (not) to use welch's t-test in side-channel security evaluations. *Cryptology ePrint Archive, Report 2017/138* (2017).

[23] Standaert, F.-X., Malkin, T., and Yung, M. A unified framework for the analysis of side-channel key recovery attacks. In *Eurocrypt* (2009), vol. 5479, Springer, pp. 443–461.

[24] Tiri, K., Hwang, D., Hodjat, A., Lai, B.-C., Yang, S., Schaumont, P., and Verbauwhede, I. Prototype ic with wddl and differential routing–dpa resistance assessment. In *International Workshop on Cryptographic Hardware and Embedded Systems* (2005), Springer, pp. 354–365.

[25] Zhang, L., Ding, A. A., Durvaux, F., Standaert, F.-X., and Fei, Y. Towards sound and optimal leakage detection procedure. *IACR Cryptology ePrint Archive 2017* (2017), 287.