Research Article

Gap between Prediction and Truth: A Case Study of False-Positives in Leakage Detection

Pengbo Wang,1,2 Ming Tang,1,2 Shoukun Xiang,3 Yaru Wang,1 and Botao Liu1

1Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China
2State Key Laboratory of Cryptology, P.O. Box 5159, Beijing 100878, China
3Wuhan Maritime Communication Research Institute, Wuhan 430079, China

Correspondence should be addressed to Ming Tang; m.tang@126.com

Received 12 May 2021; Accepted 12 January 2022; Published 12 February 2022

Academic Editor: Fazlullah Khan

Copyright © 2022 Pengbo Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Since leakage detection was introduced as a popular side-channel security assessment, it has been plagued by false-positives (a.k.a. type I errors). To fix this error, the previous solutions set detection thresholds based on an assumption-based prediction of false-positive rate (FPR). However, this study points out that such a prediction (of FPR) may be inaccurate. We notice that the prediction in EuroCrypt2016 is much smaller than (approximately 1/779 times) the true FPR. The gap between prediction and truth, called underpredicted false-positives (UFP), leads to severe false-positives in leakage detection. Then, we check the statistical distribution of test statistics to analyze the cause of UFP. Our analysis indicates that the overlap between cross-validation (CV) blocks gives rise to an assumption error in the distribution of the CV-based estimates of \( \rho \)-statistics, which is the root cause of UFP. Therefore, we tackle the UFP by eliminating the overlap between blocks. Specifically, we propose a profiling-shared validation (PSV) and utilize this validation to improve the detection of any-variate any-order leakages. Our experiments show that the PSV solves the UFP and saves more than 75% of the test time costs. In summary, this article reports a potential flaw in leakage detection and provides a complete analysis of the flaw for the first time.

1. Introduction

Side-channel attack (SCA) utilizes the physical leakages (such as execution time [1], power consumption [2], and electromagnetic radiation [3]) of a running device to retrieve some secrets (e.g., the private key) inside the device. Since it was proposed by Kocher [1], such an attack has seriously threatened the security of cryptographic modules, including smart cards [2, 3] and FPGAs [4, 5]. Thus, side-channel security assessments have been developed to evaluate the security of these modules against SCA [6].

Leakage detection is a side-channel security assessment that determines whether the leakages depend on the data (e.g., plaintext or ciphertext) accessible to an attacker [7]. In most cases, it uses a hypothesis test to identify the data-dependent leakages by comparing the test statistics with a certain threshold [8–10]. If no statistics exceed the threshold, the module is allowed to pass the test; otherwise, the module is rejected. In contrast to the attack-based assessment [6], leakage detection exploits the dependency between leakage and data rather than the key recovery. Hence, it has advantages in computational complexity and works well in black-box environments [10]. A typical example is the test vector leakage assessment (TVLA) [8]. This assessment utilizes Welch’s \( t \)-test to compare \( t \)-statistic with a threshold of 4.5 and identifies the leakages dependent on the plaintext. In EuroCrypt2016, Durvaut et al. found that TVLA failed to detect the plaintext-dependent leakages [10]. Then, they put forward a correlation-based leakage detection (\( \rho_{CV} \)-test) to identify the hard-to-detect leakages for TVLA. The \( \rho_{CV} \)-test takes advantage of cross-validation (CV) to obtain a well-estimated \( \rho \)-statistic \( \bar{\rho}_{CV} \) and then compares the \( \bar{\rho}_{CV} \) with a threshold of 5.0 to assess any-variate any-order leakages. Based on the \( \rho_{CV} \)-test, more assessments have emerged, such
as the \( \chi^2 \)-test [11], \( T^2 \)-test (and its simplification D-test) [12], KS-test [13], ANOVA [14], and DL-based test [15].

However, as discussed in [10], some of the identified plaintext-dependent leakages are in fact not related to the plaintext. In other words, leakage detection is challenged by false-positives (a.k.a. type I errors). To address this challenge, the frequently used solutions carefully set the threshold for leakage detection according to an acceptable false-positive rate (FPR) [16, 17]. Such solutions rely on an assumed distribution of the test statistics to make a fast prediction of FPR. For example, Durvaux et al. at EuroCrypt2016 chose 5.0 as the threshold of the \( \rho_{CV} \)-test based on the assumption that \( \bar{\tau}_{sz} \) follows the standard normal distribution (when \( \rho \)-statistic \( r_{\rho} = 0 \)) [10]. This choice corresponds to a theoretical \( p \) value (i.e., the predicted FPR of a single test) below \( 10^{-5} \) and reduces the predicted FPR (PFPR) of \( 10^4 \) tests to 0.0057 [17].

Obviously, the above solutions demand that the assumption must be close enough to the true distribution of the test statistics; otherwise, an assumption error (AE) in the distribution may lead to an inaccurate prediction of FPR. As far as we know, the potential risk of AE has not been studied in previous work.

This work studies the false-positives in the \( \rho_{CV} \)-test and focuses on the potential AE in [10]. We first propose an estimation algorithm that statistically approximates the true FPR of the \( \rho_{CV} \)-test. After running the algorithm, we notice the underpredicted false-positives (UFP) that the true FPR (of \( \rho_{CV} \)-test) is about 779 times the prediction (in [10] at the threshold of 5.0). The UFP implies that the previous prediction of FPR may be unreliable. Second, we discover an AE threshold of 5.0). Then, we introduce the underpredicted false-positives (UFP) that the true FPR may be unreliable. Second, we discover an AE threshold of 5.0). Then, to reduce the prediction in [10] is not accurate. In this section, we first review the details of the \( \rho_{CV} \)-test. Then, we introduce our estimation algorithm and describe the underpredicted false-positives.

2. Underpredicted False-Positives

In [10], FPR of the \( \rho_{CV} \)-test was predicted based on an assumed standard normal distribution. This section shows that the prediction in [10] is not accurate. In this section, we first review the details of the \( \rho_{CV} \)-test. Then, we introduce our estimation algorithm and describe the underpredicted false-positives.

2.1. Correlation-Based Leakage Detection. The correlation-based leakage detection (\( p \)-test) takes advantage of a \( \rho \)-statistic \( r_{\rho} \) to identify the plaintext-dependent leakage by comparing the absolute value \( |r_{\rho}| \) to a detection threshold \( T_{det} \). In [10], Durvaux et al. utilized the \( k \)-fold CV to estimate the statistic \( r_{\rho} \) at time point \( \tau \) as follows.

First, an assessor randomly splits the acquired traces \( \mathcal{L} \) into \( k \) nonoverlapping subsets \( \mathcal{L}^{(i)} \) of approximately the same size.

Then, for \( j^{th} \) (\( 1 \leq j \leq k \)) fold of CV, the assessor defines the profiling block \( \mathcal{L}_p^{(j)} : = \mathcal{L} / \mathcal{L}^{(j)} \) and the test block \( \mathcal{L}_t^{(j)} : = \mathcal{L}^{(j)} \). Thus, a CV-based model \( \rho_{CV} \) is profiled from \( \mathcal{L}^{(j)}_p \):

\[
\rho_{CV}(X) 
\]

The sample mean is \( \bar{\rho}_{CV}(X) \), where \( X \) is the associated data.

Next, compute the Pearson correlation coefficient between the model \( \rho_{CV}^{(j)}(X) \) and the test leakages \( L^{(j)}(\tau) \in \mathcal{L}^{(j)}_t \):

\[
\rho_{CV}(\tau) = \bar{\rho}
\]

where \( X^{(j)}_t \) is the data associated with the test leakages \( L^{(j)}_t \). An unbiased estimate \( \hat{\rho}_{CV}(\tau) \) of the correlation coefficient is obtained by combining the \( k \) coefficients \( \rho_{CV}^{(j)} \).

\[
\hat{\rho}_{CV}(\tau) = \frac{1}{k} \sum_{j=1}^{k} \rho_{CV}^{(j)}(\tau).
\]

Finally, after Fisher’s \( z \)-transformation and normalization, the CV-based estimate of \( \tau_{\rho} \) is

\[
\hat{\tau}_{\rho}(\tau) = \frac{\sqrt{N} - 3}{2} \ln \frac{1 + \hat{\rho}_{CV}(\tau)}{1 - \hat{\rho}_{CV}(\tau)}.
\]

where \( N \) is the size of the set \( \mathcal{L} \). Based on the assumption that \( \hat{\tau}_{sz} \) can be interpreted by a normal distribution \( \mathcal{N}(\tau_{\rho}, 1) \), the FPR of (a single) the \( \rho_{CV} \)-test can be predicted by

\[
\hat{\alpha}_{FPR} = 2 \times \left( 1 - \text{CDF}_{\mathcal{N}(0,1)} \left( |\hat{\tau}_{\rho}| \right) \right).
\]

where CDF_{\mathcal{N}(0,1)}(\cdot) is the cumulative function of the standard normal distribution \( \mathcal{N}(0,1) \).

2.2. Statistical Estimation of FPR. In statistical estimation, FPR is calculated as the ratio of the number \( N_{FP} \) of false-positives to the number \( N_{AN} \) of actual negatives:

\[
\alpha = \frac{N_{FP}}{N_{AN}}.
\]

However, neither \( N_{FP} \) nor \( N_{AN} \) is known to the evaluators in leakage detection. Therefore, we define a data-independent leakage set (DILSET) to statistically approximate the true FPR of the \( \rho_{CV} \)-test. DILSET is a leakage set, where each leakage \( L \) is independent of the associated data \( X' \), \( L^L \). Thus, \( \tau_{\rho}(\tau) = 0 \), and all the \( L \) are negatives for the
$\rho_{CV}$-test. By rewriting (6), FPR of the $\rho_{CV}$-test can be estimated:

$$\hat{\alpha}_{T_{\text{det}}} = \frac{N_{r_i>T_{\text{det}}}}{N_{\rho}},$$

where $N_{\rho}$ is the number of $\rho_{CV}$-tests on DILSET, and $N_{r_i>T_{\text{det}}}$ is the number of (false) positives $|r_i|>T_{\text{det}}$ in the $N_{\rho}$ test results.

The statistical estimation is formally described in Algorithm 1. Taking the trace set $\mathcal{L}$ and the data vector $X$ as input, the algorithm first creates a DILSET $\{\mathcal{L}, X_t\}$, where the $\rho$-statistic $r_z \equiv 0$. In each loop of Algorithm 1, the function GenLeak($\mathcal{L}$) generates a leakage vector $L$ based on $\mathcal{L}$, and $\text{GFRn} d(X)$ produces a random plaintext vector $X_t$ on the Galois field $GF(X)$, so that $X_t \parallel L$. Then, the function CorrTest(·) performs the $\rho_{CV}$-test on $\{\mathcal{L}, X_t\}$ and returns an estimate $\hat{\tau}_{\mathcal{L}, \text{CV}}$ to the vector $r_z$. After $N_{\rho}$ loops, the algorithm counts the number of $|\hat{\tau}_{\mathcal{L}, \text{CV}}|>T_{\text{det}}$ in $r_z$ and outputs the ratio $\hat{\alpha}$ as the estimated FPR (EFPR) of the $\rho_{CV}$-test. According to the law of large numbers, $\hat{\alpha}$ will converge to the true FPR of the $\rho_{CV}$-test as $N_{\rho}$ increases.

2.3. Assumption-Based Prediction vs. Statistical Estimation.

In this subsection, we run Algorithm 1 on three different DILSETs to check the prediction of FPR in [10] (i.e., equation (5)).

2.3.1. Data-Independent Leakage Set. The three DILSETs used to check the prediction are constructed as follows.

DILSET_1 is a simulated DILSET where all the leakages are Gaussian noise. Accordingly, in each loop of Algorithm 1, the function GenLeak(·) generates a leakage vector $L$ composed of 5,000 random leakages $l \sim N(0,1)$, and $\text{GFRn} d(\cdot)$ randomly produces 5,000 8-bit variables to form the associated data vector $X', X' \parallel L$.

DILSET_2 and DILSET_3 are two measured DILSETs where all the leakages come from DPA contests [18]. In each loop of Algorithm 1, the function GenLeak(·) randomly selects a time point $\tau$ from $\mathcal{L}$ and returns the leakages $L(\tau)$ at $\tau$ as $L$, and $\text{GFRn} d(\cdot)$ produces 5,000 8-bit random variables to form $X'$. The DPA contests are detailed.

DPA contest v2 targets an unprotected FPGA implementation of the advanced encryption standard (AES) algorithm and provides three databases for side-channel evaluation [19]. In this study, we use the first 5,000 traces in the template base to build DILSET_2.

DPA contest v4.2 uses a rotating SBOX masking to protect the AES implementation on a smart card and collects 5,000 traces in each zip file [20]. This study decompresses the first zip file, corresponding to the SHA1sums: f711206b41b8d02f595d5861996ff61a1711f3d, to build DILSET_3.

2.3.2. Parameter Setting. The number of folds $k$ has an impact on the bias and variance of the CV-based estimation. A larger value of $k$ gives a less biased estimate but means a higher procedure’s variance. In [21], James et al. showed that $k = 5$ and $k = 10$ make a good trade-off between the bias and the variance. We choose 5, 10, 15, and 20 as the candidates for $k$ to support the universality of our work.

The detection threshold $T_{\text{det}}$ determines the false-positives in leakage detection. Goodwill et al. suggested a threshold of 4.5 for TVLA [22]. However, as the number of points (or tests) increases, the test statistics may be so large that a leak-free device cannot pass the test with $T_{\text{det}} = 4.5$. Therefore, $T_{\text{det}} = 5.0$ was recommended for longer traces in [9] and adopted as the threshold of the $\rho_{CV}$-test in [10]. In our experiments, we set $T_{\text{det}}$ to the empirical values of 4.5 and 5.0, respectively.

A larger number of $\rho_{CV}$-tests $N_{\rho}$ improves the estimation accuracy of FPR, but means more running time of Algorithm 1. Because $\hat{\alpha}_{4.5} = 6.7953 \times 10^{-4}$ and $\hat{\alpha}_{5.0} = 5.733 \times 10^{-5}$, we set $N_{\rho} = 4^k$ to make a trade-off between the estimation accuracy and the time overhead.

2.3.3. Comparison Experiments. We run Algorithm 1 on DILSETs and record EFPRs of the $\rho_{CV}$-test, as given in Table 1. By comparing EFPRs to PFPRs, we notice that all EFPRs are much higher than PFPRs. For example, in the case of DILSET_1 and $k = 10$, EFPRs $\hat{\alpha}_{4.5} = 1.5255 \times 10^{-3}$ and $\hat{\alpha}_{5.0} = 4.465 \times 10^{-4}$, whereas PFPRs $\alpha_{4.5} = 6.7953 \times 10^{-6}$ and $\alpha_{5.0} = 5.733 \times 10^{-7}$, $\hat{\alpha}_{4.5}/\alpha_{4.5} \approx 224$ and $\hat{\alpha}_{5.0}/\alpha_{5.0} \approx 779$. The gap between EFPR and PFPR reports an underpredicted false-positives (UFP) that the FPR of the $\rho_{CV}$-test was underpredicted in [10].

3. Root Cause Analysis

Root cause analysis (RCA) identifies the root cause of UFP, which helps prevent the underprediction from recurring. This section uses hypothesis test tools to determine the root cause of UFP. In this section, we first introduce the test tools and then analyze the UFP and the AE in [10].

3.1. Hypothesis Test Tools. The tests used for our root cause analysis (RCA) are given in Table 2.

3.1.1. One-Sample $t$-Test. The one-sample $t$-test compares the sample mean to a prespecified value to test for the deviation from this value. The null hypothesis of the test assumes that no difference exists between the true mean $\mu$ and the comparison value $\bar{r}$, while the alternative assumes that the difference exists. The test statistic for a one-sample $t$-test is denoted by the letter $t$, which is calculated using the following formula:

$$t = \frac{\bar{r} - \mu}{\sigma/\sqrt{n}}$$

$\bar{\sigma}$ is the sample standard deviation, and $n$ is the sample size. Then, the $p$ value of the $t$-test can be computed:
Input: Trace set $\mathcal{X}$, Data vector $X$, Number of CV folds $k$, Detection threshold $T_{\text{det}}$, Number of $\rho$-tests $N_\rho$.

Output: Estimate of FPR $\hat{\alpha}$.

\begin{algorithm}
for $i \leftarrow 1$: $N_\rho$

\begin{align*}
L & \leftarrow \text{GenLeak} (\mathcal{X}); \\
X_1 & \leftarrow \text{GFRR} (d(X)); \\
r_2 [1: N_\rho] & \leftarrow \text{CorrTest} (L, X_1, k);
\end{align*}

end for

$N_{\text{FP}} \leftarrow \text{CountPositives} (r_2); \\
\hat{\alpha} = N_{\text{FP}} / N_\rho$;

end for

\end{algorithm}

Table 1: The FPFR vs. EFPR of the $\rho_{\text{CV}}$-test, $N_\rho = 10,000,000$.

| FPFR | DILSET | $k$-fold CV | $T_{\text{det}} = 4.5$ | $T_{\text{det}} = 5.0$
|------|--------|-------------|-----------------|----------------|
|      | DILSET_1 |             |                 |                 |
|      | k = 5   | $6.7953 \times 10^{-6}$ | $5.733 \times 10^{-7}$ |
|      | k = 10  | $1.4371 \times 10^{-3}$ | $4.000 \times 10^{-4}$ |
|      | k = 15  | $1.5255 \times 10^{-3}$ | $4.465 \times 10^{-4}$ |
|      | k = 20  | $1.5472 \times 10^{-3}$ | $4.494 \times 10^{-4}$ |
|      | k = 5   | $1.5666 \times 10^{-3}$ | $4.643 \times 10^{-4}$ |
|      | k = 10  | $1.4147 \times 10^{-3}$ | $4.115 \times 10^{-4}$ |
|      | DILSET_2 |             |                 |                 |
|      | k = 10  | $1.5335 \times 10^{-3}$ | $4.391 \times 10^{-4}$ |
|      | k = 15  | $1.5368 \times 10^{-3}$ | $4.564 \times 10^{-4}$ |
|      | k = 20  | $1.5814 \times 10^{-3}$ | $4.612 \times 10^{-4}$ |
|      | k = 5   | $1.4099 \times 10^{-3}$ | $3.935 \times 10^{-4}$ |
|      | k = 10  | $1.4958 \times 10^{-3}$ | $4.267 \times 10^{-4}$ |
|      | DILSET_3 |             |                 |                 |
|      | k = 15  | $1.5354 \times 10^{-3}$ | $4.524 \times 10^{-4}$ |
|      | k = 20  | $1.5727 \times 10^{-3}$ | $4.454 \times 10^{-4}$ |

where $\Gamma (\cdot)$ denotes the gamma function [23].

3.1.3. Kolmogorov–Smirnov Test. The Kolmogorov–Smirnov test (KS-test) is a nonparametric test that can be used to compare a sample with a reference probability distribution. Its null hypothesis assumes that the samples come from a specified distribution $\mathcal{F}$. In this test, an empirical cumulative function $F_n$ for $n$ independent and identically distributed ordered observations $X_i$ is defined as

$$F_n (x) = \frac{1}{n} \sum_{i=1}^{n} I_{[-\infty, x]} (X_i).$$

where $I_{[-\infty, x]} (X_i)$ is the indicator function which returns 1 if $X_i \leq x$ and returns 0 otherwise. The KS statistic $D_n$ for a given function $F(x)$ is

$$D_n = \sup \{|F_n (x) - F (x)|\},$$

where $\sup \cdot$ is the supremum of the set of distances. If the sample comes from $\mathcal{F}$, $D_n$ converges to zero in the limit when $n$ goes to infinity. Its $p$ value, that is, the probability that the null hypothesis holds can be calculated regarding [24].

3.1.4. Pearson’s Correlation Test. Pearson’s correlation test is used to test whether there is a relationship between two variables. Its null hypothesis assumes no correlation between the observed phenomena $\rho = 0$. Given an estimate $\hat{\rho}$ of $\rho$, the test statistic can be estimated:

$$t = \frac{\hat{\rho}}{\sqrt{1 - \hat{\rho}^2} \sqrt{n - 2}}.$$  

The $p$ value of the test is determined by Student’s $t$-distribution with degrees $n - 2$ of freedom [25]. Note that Pearson’s correlation test is different in definition from the $\rho$-test in Section 2.1.

3.2. Error in Assumed Distribution. As shown in Figure 1, the UFP implies a nonnegligible error between the true distribution of $\tilde{F}_{z, \text{CV}}$ and the assumed distribution $\mathcal{N} (0, 1)$. Regarding the description in [26], we name this error the assumption error in leakage detection and use the $\rho$ values of the tests on $\tilde{F}_{z, \text{CV}}$ to quantify the AE—the smaller the $\rho$ value,
the stronger the null hypothesis should be rejected and the more significant the AE in the $\rho_{z,CV}$-test. The results of the hypothesis tests are given in Table 3. At the significance level of 0.01 (an accepted threshold in [27]), the $p$ values of $t$-tests exceed 0.01, which means the null hypothesis $\mu = 0$ is accepted. In contrast, the $p$ values of $\chi^2$-tests and KS-tests are less than 0.01, that is, the null hypotheses $\sigma^2 = 1$ and $\Theta \sim \mathcal{N}(0, 1)$ are rejected. Hence, it is confirmed that the assumed $\mathcal{N}(0, 1)$ is not the true distribution of $\hat{r}_{z,CV}$.

3.3. Correlation between Cross-Validation Blocks. To determine the root cause of a problem, RCA establishes an event timeline from the normal situation up to the time the problem occurs. Based on equations (1)–(4), we create the timeline from $\mathcal{S}$ input to UFP (or AE), as shown in Figure 2. To facilitate RCA, intrablock estimate $\hat{r}_{z,CV}^{(k)}$ is obtained by performing Fisher’s $z$-transform and normalization on $\hat{r}_{z,CV}$.

$$\hat{r}_{z,CV}^{(j)}(\tau) = \frac{\sqrt{N/k - 3}}{2} \ln \left( \frac{1 + \hat{r}_{z,CV}^{(j)}(\tau)}{1 + \hat{r}_{z,CV}^{(j)}(\tau)} \right),$$

(16)

where $N/k$ is the number of traces in the test block $\mathcal{S}^{(j)}$.

We first check the distribution of $\hat{r}_{z,CV}$ to locate the source of AE. Table 4 provides the $p$ values of the tests on $\hat{r}_{z,CV}$ (when $k = 5$). Since all the $p$ values exceed the level of 0.01, the intrablock estimates $\hat{r}_{z,CV}^{(j)}$ pass the tests for $\mathcal{N}(0, 1)$, which excludes (1) and (2) from the possible source of AE. In [28], Bengio et al. proved that between-block correlation leads to a biased variance of the CV-based estimation (in equation (3)). Therefore, we utilize Pearson’s correlation test to examine the correlation between $(\hat{r}_{z,CV}^{(j_1)}, \hat{r}_{z,CV}^{(j_2)})$ ($j_1 \neq j_2$).

In the case of DILSET_1 and $k = 5$, the results of correlation tests are given in Table 5. Since $p \equiv 0$, the null hypothesis $\rho = 0$ is rejected. The correlation between CV blocks is confirmed and may be the cause of the AE in the $\rho_{z,CV}$-test.

| Hypothesis test tool | The null hypothesis in this study |
|----------------------|----------------------------------|
| One-sample $t$-test  | The mean $\mu = 0$.              |
| Chi-squared test     | The variance $\sigma^2 = 1$.     |
| Kolmogorov–Smirnov test | The samples come from a standard normal distribution $\mathcal{N}(0, 1)$. |
| Pearson’s correlation test | The correlation coefficient $\rho = 0$. |

**Table 2: Test tools in this study.**

**Table 3: The $p$ values of the hypothesis tests on $\hat{r}_{z,CV}$.**

| DILSET | $k$-fold CV | $t$-test | $\chi^2$-test | KS-test |
|--------|-------------|----------|---------------|---------|
| k = 5  | 0.2527      | 0        | $7.1675 \times 10^{-36}$ |         |
| k = 10 | 0.9720      | 0        | $1.4230 \times 10^{-29}$ |         |
| k = 15 | 0.2741      | 0        | $2.4038 \times 10^{-36}$ |         |
| k = 20 | 0.9630      | 0        | $1.6397 \times 10^{-30}$ |         |
| k = 5  | 0.9785      | 0        | $3.5003 \times 10^{-30}$ |         |
| k = 10 | 0.1445      | 0        | $3.4179 \times 10^{-34}$ |         |
| k = 15 | 0.3034      | 0        | $3.3193 \times 10^{-33}$ |         |
| k = 20 | 0.9335      | 0        | $7.3287 \times 10^{-32}$ |         |
| k = 5  | 0.2746      | 0        | $5.5938 \times 10^{-32}$ |         |
| k = 10 | 0.7406      | 0        | $3.2925 \times 10^{-32}$ |         |
| k = 15 | 0.8433      | 0        | $3.1929 \times 10^{-33}$ |         |
| k = 20 | 0.2075      | 0        | $3.0201 \times 10^{-36}$ |         |
Figure 2: Event timeline used for the RCA of UFP. The solid line is the timeline from \( \mathcal{E} \) input to UFP occurrence (i.e., \( r_{z,CV} \) output), and the dashed line is the operation to obtain the intrablock estimate \( \bar{r}_{z,CV} \).

Table 4: The \( p \) values of the hypothesis tests on \( \bar{r}_{z,CV}^{(j)} (k = 5) \).

| DILSET_1 | DILSET_2 | DILSET_3 |
|----------|----------|----------|
| \( \bar{r}_{z,CV}^{(1)} \) | t-test | \( \chi^2 \)-test | KS-test | t-test | \( \chi^2 \)-test | KS-test |
| \( \bar{r}_{z,CV}^{(2)} \) | 0.1458 | 0.1405 | 0.2098 | 0.8268 | 0.9626 | 0.9957 | 0.8209 | 0.6617 | 0.9697 |
| \( \bar{r}_{z,CV}^{(3)} \) | 0.8336 | 0.5541 | 0.9989 | 0.5929 | 0.1430 | 0.2672 | 0.2606 | 0.0310 | 0.1622 |
| \( \bar{r}_{z,CV}^{(4)} \) | 0.4676 | 0.3257 | 0.7521 | 0.6638 | 0.9034 | 0.9294 | 0.7215 | 0.4373 | 0.2853 |
| \( \bar{r}_{z,CV}^{(5)} \) | 0.8638 | 0.7702 | 0.8815 | 0.2165 | 0.6410 | 0.6507 | 0.6913 | 0.5691 | 0.7465 |
| \( \bar{r}_{z,CV} \) | 0.4571 | 0.2662 | 0.6768 | 0.7206 | 0.3444 | 0.5294 | 0.7058 | 0.1246 | 0.6759 |

Table 5: The \( p \) values of Pearson’s correlation test between \( \bar{r}_{z,NOV}^{(j)} \) (DILSET_1, \( k = 5 \)).

| \( \bar{r}_{z,NOV}^{(1)} \) | \( \bar{r}_{z,NOV}^{(2)} \) | \( \bar{r}_{z,NOV}^{(3)} \) | \( \bar{r}_{z,NOV}^{(4)} \) | \( \bar{r}_{z,NOV}^{(5)} \) |
|----------------|----------------|----------------|----------------|----------------|
| \( \bar{r}_{z,NOV}^{(1)} \) | — | 0 | 0 | 0 | 0 |
| \( \bar{r}_{z,NOV}^{(2)} \) | 0 | — | 0 | 0 | 0 |
| \( \bar{r}_{z,NOV}^{(3)} \) | 0 | 0 | — | 0 | 0 |
| \( \bar{r}_{z,NOV}^{(4)} \) | 0 | 0 | 0 | — | 0 |
| \( \bar{r}_{z,NOV} \) | 0 | 0 | 0 | 0 | — |

3.4. Overlapping Cross-Validation Blocks. According to Corollary 3 in [28], the between-block correlation in cross-validation stems from the overlap between the training and the test blocks. In this subsection, we present a nonoverlapping validation (NOV) and compare the NOV-based \( \rho \)-test (\( \rho_{NOV} \)-test) with the \( \rho_{CV} \)-test to check the impact of the overlap on FPR. The \( \rho_{NOV} \)-test works as follows.

First, the traces \( \mathcal{E} \) are split into \( k \) nonoverlapping blocks \( \mathcal{E}^{(j)} \) of approximately the same size. For each block \( \mathcal{E}^{(j)} \), \( 1/m \) of the traces are randomly selected as the test subset \( \mathcal{L}^{(j)} \), and the others are left as the profiling subset \( \mathcal{L}^{(j)} \). Then, the model \( \hat{\varphi}_{NOV}^{(j)} \) is profiled:

\[
\hat{\varphi}_{NOV}^{(j)}(X) = \mathcal{L}^{(j)}.
\]

Finally, an NOV-based estimate \( \bar{r}_{z,NOV} \) of \( \rho \)-statistic can be calculated by equations (18)-(20):

\[
\bar{r}_{z,NOV}^{(j)}(\tau) = \hat{\varphi}(\mathcal{L}^{(j)}(\tau), \hat{\varphi}_{NOV}^{(j)}(\mathcal{X}^{(j)})), \tag{18}
\]

\[
\bar{r}_{z,NOV}^{(j)}(\tau) = \sqrt{\frac{\sqrt{N/m \times k - 3}}{2}} \ln \left( \frac{1 + \bar{r}_{z,NOV}^{(j)}(\tau)}{1 - \bar{r}_{z,NOV}^{(j)}(\tau)} \right), \tag{19}
\]

We run Algorithm 1 on an extended DILSET_1 (ExDILSET_1) to approximate the true FPR of the \( \rho_{NOV} \)-test. Specifically, in each loop of the algorithm, the function \( \text{GenLeak}() \) produces \( 50,000 \) random leakages \( \mathcal{L} \sim \mathcal{N}(0, 1) \) to form the leakage vector \( L \), and \( \text{GFRn}d() \) randomly generates a vector \( \tilde{X} \) composed of \( 50,000 \) 8 bit variables, so that \( \tilde{X} \) is \( L \). Then, the function \( \text{CorrTest}() \) performs the \( \rho_{NOV} \)-test on \( \{\mathcal{E}, \mathcal{X}\} \) and returns an NOV-based estimate \( \bar{r}_{z,NOV} \). After \( 10,000,000 \) loops, \( 10,000,000 \bar{r}_{z,NOV}^{(j)} \) (for each \( j \)) and \( 10,000,000 \bar{r}_{z,NOV} \) are obtained. Finally, Algorithm 1 counts the number of false-positives and outputs the EFPR of the \( \rho_{NOV} \)-test.
Table 6: The \( p \) values of Pearson’s correlation test between \( \tau_{z,NOV}^{(j)} \) (ExDILSET_1, \( m = 10, k = 5 \)).

| \( j_1 \)  | \( j_2 \)  | \( j_3 \)  | \( j_4 \)  | \( j_5 \)  |
|------------|------------|------------|------------|------------|
| 1          | 0.4215     | 0.5670     | 0.2791     | 0.0839     |
| 2          | 0.5670     | 0.3591     | 0.0372     | 0.7828     |
| 3          | 0.2791     | 0.7718     | 0.0372     | 0.9380     |
| 4          | 0.0839     | 0.0463     | 0.7828     | 0.9380     |

4. Improved \( \rho \)-Test

Section 3 has shown that NOV is an effective solution to the UFP in the \( \rho_{CV} \)-test, but requires more samples to construct the \( k \) nonoverlapping blocks. In this section, we introduce the profiling-shared validation to efficiently solve the UFP.

4.1. Profiling-Shared Validation. Profiling-shared validation shares the same profiling samples and the same profiling model for all the test blocks. In a PSV-based \( \rho \)-test (\( \rho_{PSV} \)-test), the evaluator randomly splits the traces into \( m + k \) nonoverlapping subsets \( \mathcal{D}^{(0)} \) with approximately same size. Choosing the first \( m \) subsets as the profiling block \( \mathcal{D}_r^{(0)} = \{ \mathcal{D}^{(0)} \mid 1 \leq i \leq m \} \), the model at point time \( r \) can be profiled:

\[
\text{model}_{\rho_{PSV}}(X) \sim \mathcal{D}_r.
\]  

Then, for \( j \)th test block \( \mathcal{D}_r^{(j)} = \mathcal{D}^{(j+m)}, 1 \leq j \leq k \), the Pearson correlation coefficient between the model and the test leakages \( L_i^{(j)}(r) \in \mathcal{D}_r^{(j)} \) is computed:

\[
\hat{\tau}_{PSV}^{(j)}(r) = \rho\left(L_i^{(j)}(r), \text{model}_{\rho_{PSV}}(X_i^{(j)})\right).
\]  

Next, after Fisher’s \( z \)-transformation and normalization, the intrablock estimates \( \hat{\tau}_{z,PSV}^{(j)} \),

\[
\hat{\tau}_{z,PSV}^{(j)}(r) = \frac{\sqrt{N/m+k-3}}{2} \ln \left(\frac{1 + \hat{\tau}_{PSV}^{(j)}(r)}{1 - \hat{\tau}_{PSV}^{(j)}(r)}\right).
\]  

Finally, averaging the intrablock estimates \( \hat{\tau}_{z,PSV}^{(j)} \) and then normalizing the mean to obtain a PSV-based estimate \( \hat{\tau}_{z,PSV}(r) \) of \( \rho \)-statistic,

\[
\hat{\tau}_{z,PSV}(r) = \frac{1}{\sqrt{k}} \sum_{j=1}^{k} \hat{\tau}_{z,PSV}^{(j)}(r).
\]

4.2. Effectiveness and Efficiency

4.2.1. Effectiveness. We perform the \( \rho_{PSV} \)-test on DILSET_1, 2, and 3 and then test the estimates \( \hat{\tau}_{z,PSV} \) and \( \hat{\tau}_{PSV} \). Since each PSV block is independent of other blocks, \( \rho\left(\hat{\tau}_{z,PSV}^{(j)}, \hat{\tau}_{PSV}^{(j)}\right) = 0, j_1 \neq j_2 \). For example, in the case of DILSET_1 and \( (m,k) = (10,5) \), the obtained results of correlation tests between \( \hat{\tau}_{z,PSV} \) are given in Table 8. At the significance level of 0.01, the null hypothesis \( \rho = 0 \) holds. Table 9 provides the \( p \) values of the distribution tests on \( \hat{\tau}_{z,PSV} \) and EFPRs of the \( \rho_{PSV} \)-test. Since all the \( p \) values exceed the acceptable level of 0.01, the PSV-based estimates \( \hat{\tau}_{z,PSV} \) pass the test for \( \mathcal{F} \) (0, 1). In addition, compared to EFPRs of the \( \rho_{CV} \)-test (Table 1), EFPRs of \( \rho_{PSV} \)-tests are much closer to PFPRs. Hence, PSV is an effective solution to the AE and UFP.

4.2.2. Efficiency. Different from the CV which requires \( k \)-profiled models for the validation, PSV shares the same model for all test blocks, which means the \( \rho_{PSV} \)-test spends less time in the profiling phase than the \( \rho_{CV} \)-test. We compare the execution time of the \( \rho_{PSV} \)-test and \( \rho_{CV} \)-test on an HP EliteBook 735 G6 (AMD Ryzen 5 PRO 3500U CPU @ 2.1 GHz, 8 GB RAM) with Windows 10 as the operating system. The time overhead of 1,000,000 tests on DILSET_1 is shown in Figure 3. Compared with the time cost of 1,000,000 \( \rho_{CV} \)-tests of at least 676.94 seconds when \( k = 5 \), the maximum time cost of 1,000,000 \( \rho_{PSV} \)-tests is 144.98 seconds when \( (m,k) = (10,5) \), saving about 79% of the time costs. In short, the \( \rho_{PSV} \)-test is more time-efficient than the \( \rho_{CV} \)-test.

5. Higher-Order Leakage Detection

Higher-order leakage detection evaluates the security of the protected implementation against higher-order side-channel analysis. In this section, we analyze the performance of the PSV method in detecting higher-order leakages.

5.1. Combining Function. In higher-order side-channel analysis, a combining function maps the leakages of multiple shares to a simple univariate leakage. The work of Prouff et al. has demonstrated the central product function:

\[
C(r) = \prod_{j=1}^{d} \left( L(r_j^i) \frac{1}{N} \sum_{i=1}^{N} L_i(r_j^i) \right),
\]

where \( L(r_j^i) \) is the \( j \)th leakage at time point \( r_j^i \) in trace set \( \mathcal{L} \) and is optimal for the Hamming weight leakage scenario represented by the smart card platform [29]. In our experiments, the central product function is selected to preprocess the leakages of the masking. According to the classification in [30], we analyze the performance of PSV under the following combining functions:
Table 7: The test results for the AE and UFP in the $\rho_{NOV}$-test (ExDILSET_1, $m = 10$).

| DILSET       | $k$-fold NOV | $t$-test | $\chi^2$-test | KS-test | $T_{det} = 4.5$ | $T_{det} = 5.0$ |
|--------------|--------------|----------|----------------|---------|----------------|----------------|
| ExDILSET_1   | $k = 5$      | 0.6346   | 0.0342         | 0.2767  | $6.4 \times 10^{-6}$ | $6 \times 10^{-7}$ |
|              | $k = 10$     | 0.0816   | 0.1658         | 0.1716  | $5 \times 10^{-6}$  | $1 \times 10^{-7}$ |
|              | $k = 15$     | 0.4912   | 0.1817         | 0.1957  | $6.8 \times 10^{-6}$ | $1.0 \times 10^{-6}$ |
|              | $k = 20$     | 0.8411   | 0.5330         | 0.8187  | $6.0 \times 10^{-6}$ | $8 \times 10^{-7}$ |

Table 8: The $p$ values of Pearson’s correlation test between $\hat{\rho}^{(j)}_{PSV}$ (DILSET_1, $m = 10$, $k = 5$).

| $j_1$ | $j_2 = 1$ | $j_2 = 2$ | $j_2 = 3$ | $j_2 = 4$ | $j_2 = 5$ |
|-------|-----------|-----------|-----------|-----------|-----------|
| 1     | —         | 0.3471    | 0.6185    | 0.9131    | 0.9447    |
| 2     | 0.3471    | —         | 0.7603    | 0.0151    | 0.7986    |
| 3     | 0.6185    | 0.7603    | —         | 0.0139    | 0.8772    |
| 4     | 0.9131    | 0.0151    | 0.0139    | —         | 0.6937    |
| 5     | 0.0447    | 0.7986    | 0.8772    | 0.6937    | —         |

Table 9: The test results for the AE and UFP in the $\rho_{PSV}$-test.

| DILSET       | $(m, k)$-PSV | $t$-test | $\chi^2$-test | KS-test | $T_{det} = 4.5$ | $T_{det} = 5.0$ |
|--------------|--------------|----------|----------------|---------|----------------|----------------|
| DILSET_1     | (5, 5)       | 0.3663   | 0.1737         | 0.5475  | $6.6 \times 10^{-6}$ | $5 \times 10^{-7}$ |
|              | (5, 10)      | 0.1969   | 0.8244         | 0.6305  | $7.5 \times 10^{-6}$ | $9 \times 10^{-7}$ |
|              | (10, 5)      | 0.9605   | 0.6395         | 0.8367  | $7.6 \times 10^{-6}$ | $6 \times 10^{-7}$ |
|              | (10, 10)     | 0.6426   | 0.5536         | 0.3759  | $5.9 \times 10^{-6}$ | $7 \times 10^{-7}$ |
| DILSET_2     | (5, 5)       | 0.4444   | 0.1916         | 0.0366  | $6.7 \times 10^{-6}$ | $5 \times 10^{-7}$ |
|              | (5, 10)      | 0.8794   | 0.3818         | 0.9811  | $8.8 \times 10^{-6}$ | $9 \times 10^{-7}$ |
|              | (10, 5)      | 0.1102   | 0.5267         | 0.0623  | $6.5 \times 10^{-6}$ | $1.1 \times 10^{-6}$ |
|              | (10, 10)     | 0.7468   | 0.3270         | 0.9834  | $6.1 \times 10^{-6}$ | $5 \times 10^{-7}$ |
| DILSET_3     | (5, 5)       | 0.3469   | 0.4118         | 0.6994  | $6.9 \times 10^{-6}$ | $6 \times 10^{-7}$ |
|              | (5, 10)      | 0.1937   | 0.5068         | 0.3475  | $8.0 \times 10^{-6}$ | $5 \times 10^{-7}$ |
|              | (10, 5)      | 0.8339   | 0.7577         | 0.9392  | $8.1 \times 10^{-6}$ | $7 \times 10^{-7}$ |
|              | (10, 10)     | 0.9565   | 0.1542         | 0.6326  | $7.0 \times 10^{-6}$ | $5 \times 10^{-7}$ |

(1) The central product function with the number of variables $d = 1$ and the order of statistical moment $o = 2$,
\[
C_1^2(r) = \left( L(r) - \frac{1}{N} \sum_{i=1}^{N} L_i(r) \right)^2.
\]

(2) The central product function with $d = 1$ and $o = 3$,
\[
C_1^3(r) = \left( L(r) - \frac{1}{N} \sum_{i=1}^{N} L_i(r) \right)^3.
\]

(3) The central product function with $d = 2$ and $o = 2$,
\[
C_2^2(r) = \prod_{j=1}^{2} \left( L(r) - \frac{1}{N} \sum_{i=1}^{N} L_i(r_j) \right).
\]

(4) The central product function is with $d = 3$ and $o = 3$,
\[
C_3^3(r) = \prod_{j=1}^{3} \left( L(r) - \frac{1}{N} \sum_{i=1}^{N} L_i(r_j) \right).
\]

Figure 3: Time cost of the $\rho_{PSV}$-test vs. $\rho_{CV}$-test. The solid line is the time cost of the $\rho_{PSV}$-test, and the dashed line is the time cost of the $\rho_{CV}$-test.
5.2. Effectiveness and Efficiency

5.2.1. Effectiveness. As described in Section 3, the \( p \)-values of the distribution tests on \( \tau_z \) quantify the AE and the UFP in the \( \rho \)-test. Thus, we test the distribution of \( \tau_{z,PSV} \) to demonstrate the effectiveness of PSV in higher-order leakage detection, where the \( \tau_{z,PSV} \) is calculated from the combined DILSETs with different \((o, d)\). For example, when \((o, d) = (2, 2)\), we compute the estimates \( \tau_{z,CV} \) and \( \tau_{z,PSV} \) as follows. In addition to the vector \( X' \), two sets of leakages \( l \sim N(0, 1) \) are randomly generated, so that \( L(\tau_1) \parallel L(\tau_2) \parallel X' \). Then, the leakages \( L(\tau_1) \) and \( L(\tau_2) \) are combined by the central product function (28). Finally, we perform the \( \rho_{CV} \)-test or \( \rho_{PSV} \)-test on the combined leakages to obtain \( \tau_{z,CV} \) or \( \tau_{z,PSV} \). For each \((o, d)\), Table 10 provides the results of the distribution tests on \( \tau_{z,CV} \) and \( \tau_{z,PSV} \). At the significance level of 0.01, \( \tau_{z,CV} \) is rejected by the test of \( \sigma^2 = 1 \) and \( N(0, 1) \), while \( \tau_{z,PSV} \) passes. It is proved that the AE and UFP in the higher-order \( \rho_{CV} \)-test can be solved by the PSV method.

5.2.2. Efficiency. The time costs of 1,000,000 \( 2^{nd} \)-order \( \rho \)-tests are shown in Figure 5. As shown in Figure 5(a), compared with the time cost of 1,000,000 univariate \( 2^{nd} \)-order \( \rho_{CV} \)-tests of at least 743.85 seconds, the maximum time cost of the 1,000,000 \( \rho_{PSV} \)-test is 161.31 seconds, saving about 78% of the time overhead of detections. In Figure 5(b), compared with the minimum time cost of the bivariate \( 2^{nd} \)-order \( \rho_{CV} \)-test of 767.80 seconds, the \( \rho_{PSV} \)-test takes at most 187.23 seconds, saving about 76% of the time overhead. In other words, the higher-order \( \rho_{PSV} \)-test is more time-efficient than the higher-order \( \rho_{CV} \)-test.
Table 10: Test results for the AE and UFP in the higher-order $\rho$-test.

| Validation  | 2nd-order univariate $\rho$-test | 2nd-order bivariate $\rho$-test |
|-------------|-----------------------------------|----------------------------------|
|             | $t$-test | $\chi^2$-test | KS-test | $t$-test | $\chi^2$-test | KS-test |
| 5-fold CV   | 0.1420  | 0            | $7.416 \times 10^{-28}$ | 0.1190  | 0            | $3.6605 \times 10^{-29}$ |
| 10-fold CV  | 0.8306  | 0            | $3.2023 \times 10^{-32}$ | 0.2635  | 0            | $2.3100 \times 10^{-36}$ |
| 15-fold CV  | 0.5540  | 0            | $2.2630 \times 10^{-37}$ | 0.1769  | 0            | $3.3470 \times 10^{-40}$ |
| 20-fold CV  | 0.1899  | 0            | $3.0929 \times 10^{-37}$ | 0.5986  | 0            | $3.5635 \times 10^{-33}$ |
| (5,5)-PSV   | 0.5719  | 0.1914       | 0.8120  | 0.3307  | 0.2494       | 0.2121  |
| (5,10)-PSV  | 0.4067  | 0.0477       | 0.5899  | 0.0234  | 0.3518       | 0.0916  |
| (10,5)-PSV  | 0.2351  | 0.3899       | 0.3563  | 0.0555  | 0.6004       | 0.1146  |
| (10,10)-PSV | 0.3265  | 0.2440       | 0.4941  | 0.0213  | 0.0994       | 0.0212  |
|             | $3rd$-order univariate $\rho$-test | $3rd$-order 3-variate $\rho$-test |
| Validation  | $t$-test | $\chi^2$-test | KS-test | $t$-test | $\chi^2$-test | KS-test |
| 5-fold CV   | 0.6974  | 0            | $2.3462 \times 10^{-34}$ | 0.6372  | 0            | $8.8311 \times 10^{-35}$ |
| 10-fold CV  | 0.9003  | 0            | $1.1018 \times 10^{-28}$ | 0.1725  | 0            | $6.6531 \times 10^{-35}$ |
| 15-fold CV  | 0.3149  | 0            | $8.2116 \times 10^{-36}$ | 0.8509  | 0            | $2.6746 \times 10^{-33}$ |
| 20-fold CV  | 0.9979  | 0            | $1.6713 \times 10^{-30}$ | 0.7414  | 0            | $1.2006 \times 10^{-33}$ |
| (5,5)-PSV   | 0.6206  | 0.8854       | 0.6493  | 0.6512  | 0.9185       | 0.9123  |
| (5,10)-PSV  | 0.1166  | 0.7295       | 0.3953  | 0.7761  | 0.0327       | 0.4982  |
| (10,5)-PSV  | 0.7967  | 0.1901       | 0.4832  | 0.4607  | 0.3087       | 0.5690  |
| (10,10)-PSV | 0.5551  | 0.8565       | 0.7678  | 0.9965  | 0.2055       | 0.2865  |

Figure 5: Time cost of the 2nd-order $\rho_{PSV}$-test vs. 2nd-order $\rho_{CV}$-test. The solid line is the time cost of the $\rho_{PSV}$-test, and the dashed line is the time cost of the $\rho_{CV}$-test. (a) $\alpha = 2$, $d = 1$. (b) $\alpha = 2$, $d = 2$.

5.3. Measured Experiments. We use (26) to preprocess DPA contest v4.2 and run the $\rho_{PSV}$-test on the preprocessed traces. Figure 6 shows the results of the $\rho_{PSV}$-test between the traces and the first 4 plaintext bytes. When $T_{det} = 5.0$, the first 4 bytes are leaked in the intervals $[379.97 \mu s, 381.00 \mu s]$, $[380.81 \mu s, 381.81 \mu s]$, $[381.65 \mu s, 382.69 \mu s]$, and
which means that the univariate 2nd-order leakages in the measured power consumption are identified by the $\rho_{PSV}$-test.

### Figure 6: Univariate 2nd-order $\rho_{PSV}$-tests on DPA contest v4.2

The dashed line is the detection threshold 5.0. (a) $m = 5$, $k = 5$. (b) $m = 5$, $k = 10$. (c) $m = 10$, $k = 5$. (d) $m = 10$, $k = 10$.

which means that the univariate 2nd-order leakages in the measured power consumption are identified by the $\rho_{PSV}$-test.

### 6. Conclusions

Assumption error (AE) invalidates side-channel security assessment. This study finds that the false-positive rate of leakage detection might be mispredicted due to potential errors between the assumed distribution and the true distribution of the estimated test statistics. We notice underpredicted false-positive (UFP) in [10]. This under-prediction, interpreted as the AE in the statistical distribution of the estimates of $\rho$-statistics, is caused by the overlap between the training and the test blocks in cross-validation. In addition, we propose the profiling-shared validation (PSV) to improve the detection of any-variate any-order leakages and show that the UFP and AE can be addressed by eliminating the between-block overlap. Compared with the $\rho_{CV}$-test, our $\rho_{PSV}$-test overcomes the UFP and only takes less than 25% of the time cost. To the best of our knowledge, this article is the first empirical study of the false-positives in leakage detection. In future work, we will refine our tools, including the estimation algorithm and the distribution tests, to preevaluate other security assessments.

### Data Availability

The datasets and codes used to support the findings of this study are available from the corresponding author upon request.
Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (61972295), the Wuhan Science and Technology Project Application Foundation Frontier Special Project (2019010701011407), and the Foundation Project of Wuhan Maritime Communication Research Institute (2018)-11.

References

[1] P. C. Kocher, "Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems," in Proceedings of the Annual International Cryptology Conference, pp. 104–113, Springer, California, CI, USA, August 1996.
[2] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in Proceedings of the Annual international cryptology conference, pp. 388–397, Springer, California, CA, USA, August 1999.
[3] D. Agrawal, B. Archambeault, J. R. Rao, and P. Rohatgi, "The EM side—channel (s)," International Workshop on Cryptographic Hardware and Embedded Systems, Springer, Berlin, Germany, 2002.
[4] S. Mangard, N. Pramstaller, and E. Oswald, "Successfully attacking masked AES hardware implementations," in Proceedings of the International workshop on cryptographic hardware and embedded systems, pp. 157–171, Springer, Edinburgh, UK, August 2005.
[5] C. Archambeau, E. Peeters, and F.-X. Standaert, "Template attacks in principal subspaces," International Workshop on Cryptographic Hardware and Embedded Systems, Springer, Berlin, Germany, 2006.
[6] F. X. Standaert, T. G. Malkin, and M. Yung, "A unified framework for the analysis of side-channel key recovery attacks," in Proceedings of the Annual international conference on the theory and applications of cryptographic techniques, pp. 443–461, Springer, Cologne, Germany, April 2009.
[7] K. Chatzikokolakis, T. Chothia, and A. Guha, "Statistical measurement of information leakage," in Proceedings of the International Conference on Tools and Algorithms for the Construction and Analysis of Systems, pp. 390–404, Springer, Paphos, Cyprus, March 2010.
[8] B. J. Gilbert Goodwill, J. Jaffe, and P. Rohatgi, "A testing methodology for side-channel resistance validation," in Proceedings of the NIST non-invasive attack testing workshop, vol. 7, pp. 115–136, Nara, Japan, September 2011.
[9] J. Balasch, B. Gierlichs, V. Grosso, and O. Reparaz, "On the cost of lazy engineering for masked software implementations," in Proceedings of the International Conference on Smart Card Research and Advanced Applications, pp. 64–81, Springer, November 2014.
[10] F. Durvaux and F. X. Standaert, "From improved leakage detection to the detection of points of interests in leakage traces," in Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 240–262, Springer, May 2016.
[11] A. Moradi, B. Richter, T. Schneider, and F.-X. Standaert, "Leakage detection with the x2-test," IACR Transactions on Cryptographic Hardware and Embedded Systems, pp. 209–237, 2018.
[12] O. Bronchain, T. Schneider, and F. X. Standaert, “Multi-tuple leakage detection and the dependent signal issue,” IACR Transactions on Cryptographic Hardware and Embedded Systems, pp. 318–345, 2019.
[13] X. Zhou, K. Qiao, and C. Ou, "Leakage detection with Kolmogorov-smirnov test," IACR Cryptol. ePrint Arch., 2019.
[14] W. Yang and A. Jia, "side-channel leakage detection with one-way analysis of variance," Security and Communication Networks, vol. 2021, Article ID 6614702, 13 pages, 2021.
[15] T. Moos, F. Wegener, and A. Moradi, "DI-la: deep learning leakage assessment: a modern roadmap for sca evaluations," IACR Transactions on Cryptographic Hardware and Embedded Systems, pp. 552–598, 2021.
[16] F. Bache, C. Plump, and T. Güneyusu, "Confident leakage assessment—a side-channel evaluation framework based on confidence intervals," in Proceedings of the Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 1117–1122, IEEE, Dresden, Germany, March 2018.
[17] A. A. Ding, L. Zhang, F. Durvaux, and F. X. Standaert, “Towards sound and optimal leakage detection procedure,” in Proceedings of the International Conference on Smart Card Research and Advanced Applications, pp. 105–122, Springer, Lugano Switzerland, November 2017.
[18] Dpacocontest, “Telecom ParisTech,” 2021, http://www.dpacocontest.org/home/.
[19] C. Clavier, J. L. Danger, G. Duc, and M. A. Elaabid, “Practical improvements of side-channel attacks on AES: feedback from the 2nd DPA contest,” Journal of Cryptographic Engineering, vol. 4, no. 4, pp. 259–274, 2014.
[20] Z. Martinasek, F. Iglesias, L. Malina, and J. Martinasek, “Crucial pitfall of DPA Contest V4. 2 implementation,” in Proceedings of the 2nd DPA contest, vol. 4, no. 4, 2014.
[21] J. A. Cornell, Introductory Mathematical Statistics: Principles and Methods, Wiley, New York, NY, USA, 1971.
[22] A. Jaffe, F. DeMulder, G. Goodwill, J. Jaffe, G. Kenworthy, and P. Rohatgi, Eds., in Proceedings of the International Catholic Migration Commission, Gaithersburg, MD, USA, September 2013.
[23] J. A. Rice, Mathematical Statistics and Data Analysis, Cengage Learning, Massachusetts, MA, USA, 2006.
[28] Y. Bengio and Y. Grandvalet, “No unbiased estimator of the variance of k-fold cross-validation,” *Journal of Machine Learning Research*, vol. 5, no. Sep, pp. 1089–1105, 2004.

[29] E. Prouff, M. Rivain, and R. Bevan, “Statistical analysis of second order differential power analysis,” *IEEE Transactions on Computers*, vol. 58, no. 6, pp. 799–811, 2009.

[30] A. Moradi and O. Mischke, “On the simplicity of converting leakages from multivariate to univariate,” in *Proceedings of the International Conference on Cryptographic Hardware and Embedded Systems*, pp. 1–20, Springer, California, CA, USA, August 2013.