International Standard

ISO/IEC 17825

Second edition
2024-01

Information technology — Security techniques — Testing methods for the mitigation of non-invasive attack classes against cryptographic modules

Technologie de l'information — Techniques de sécurité — Méthodes de test pour la protection contre les attaques non intrusives des modules cryptographiques

Reference number
ISO/IEC 17825:2024(en)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

ISO and IEC draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO and IEC take no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO and IEC had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents and https://patents.iec.ch. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 27, *Information security, cybersecurity and privacy protection*.

This second edition cancels and replaces the first edition (ISO/IEC 17825:2016), which has been technically revised.

The main changes are as follows:

— test methods have been updated as per research trends;

— an introduction has been added which states the expectations in terms of security level of this document;

— requirements have been numbered to ensure their traceability.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Introduction

Testing requires defined constants, which are derived from an axiomatic analysis of the security problem. The security assurance levels are bound to the testing and remaining risks. The testing approach can be characterized as follows:

a) Testing soundness

   1) A formal description of empirical closed-box testing provides the soundness, in the context of the attack, because the testing adheres to an accepted methodology.

   2) The application of the methodology does not ensure that all possible attacks are covered. Testing allows for weakness detection in a system; hence, it increases the confidence in a system's ability to withstand a set of simulated attacks. The implemented formalism allows to detect weaknesses, and the outcome is a reasonable level attested by tests.

   3) The level of assurance that can be reached with the methodology in this document is a "controlled" level of "reasonable" confidence level, which is the level low to medium. Level high is not reachable due to the closed-box approach. The meaning of "reasonable" is determined by the customer's risk threshold. The tester is defining the level of reasonability, in accordance with a security level target.

   4) Testing is guided by a strategy, which allows for transparency in the methodology and outcomes.

   5) The methodology is device-class specific. The pass/fail criteria should take into account the class of devices under test. For example, the criteria for devices with a deterministic behaviour (i.e. bare metal), and for devices with a complex software stack should be different.

   6) Security testing is an "estimation" when based upon noisy measurements, or when the tester does not have full control of the implementation under test (IUT).

b) Repeatability (as per ISO/IEC 17025:2017, 7.2.2.4)

   Repeatability means similar results from the same (i.e. repeated) methodology, while reproducibility means similar results from similar methodology. Security evaluation is an estimation based on noisy measurements, on IUT whose behaviour is probably not in full control of the tester. In this document, there is a prerequisite that the IUT is closed-box, which can behave in a non-deterministic manner (at least, its internals – owing to some intentional randomization used as a protection). Furthermore, the test can only be carried out based on external observations and findings. As a result, the objective is to document a formal and transparent process of testing, where independent tests can be reproduced with similar expected results (as much as possible, within reasonable bounds). The methodologies are similar (e.g. executed by two testers) in that they yield similar outcome.

c) Cost of testing

   1) The objective is to devote the right amount of effort for the testing of a given assurance level. Cost effectiveness of the testing has a direct implication on assuring a certain level of security. Cost of testing includes, but is not limited to:

      i) Level of expertise and experience: Consequence/implication of using an already formalized process (agnostic in the IUT). The testers require skills and competencies.

      ii) Time: Elapsed time for data acquisition, even though the procedure is automated.

      iii) Equipment: The cost impact of equipment is covered in ISO/IEC 20085-1:2019 (requirements) and ISO/IEC 20085-2:2020 (calibration).

   2) This document aims to keep cost moderate. A threshold is reached in the assurance level up to a certain number of traces captured. The level of assurance does not increase significantly more beyond the threshold. The prescribed methodology cannot exceed a certain level of assurance by its design.

The following statements apply as an artefact of the methodology used:

d) Closed-box testing limits this methodology to exclusively test for leakage that does not account for specific features of a given algorithm's implementation (e.g. implementation specificities, such as parallel execution of unrelated cryptographic operations, or countermeasures, such as random masking, implementation of field arithmetic in elliptic curve cryptography).

e) Testing only considers leakage during tested cryptographic operations using keys. By design the process does not look for other potential sources of leakage (e.g. emissions during transit of keys over internal bus).

f) Results are dependent on the data sets and quality of equipment used during acquisition. Attackers with larger resources can still exploit attack paths tested by this methodology, even if they had passed the test based on increased resources and effort.

g) More sophisticated attacks can be applied and succeed. More sophisticated attacks refer to attacks other than conventional ones, for example the attacks that are particular to asymmetric ciphers (see 9.2).

h) Each specific application/cryptographic module API instance also requires a delta evaluation on top of the generic tests in this document. Such areas of assessment should include application-specific non-parametric module usage threats, such as traffic analysis, manipulation of logical order or scope of external operations.

In this document, requirements are numbered. By convention, the requirements are labelled as [CC.NN], where CC represents the clause number (e.g. 06 means Clause 6), and NN represents the requirement position within the Clause (e.g. the first requirement of Claude 6 is referred to as [06.01]). The purpose of labelled requirements is to ease the generation of documents showing compliance with this document, and their traceability for testers.

Information technology — Security techniques — Testing methods for the mitigation of non-invasive attack classes against cryptographic modules

1 Scope

This document specifies the non-invasive attack mitigation test metrics for determining conformance to the requirements specified in ISO/IEC 19790:2012 for security levels 3 and 4. The test metrics are associated with the security functions addressed in ISO/IEC 19790:2012. Testing is conducted at the defined boundary of the cryptographic module and the inputs/outputs available at its defined boundary.

This document is intended to be used in conjunction with ISO/IEC 24759:2017 to demonstrate conformance to ISO/IEC 19790:2012.

NOTE ISO/IEC 24759:2017 specifies the test methods used by testing laboratories to assess whether the cryptographic module conforms to the requirements specified in ISO/IEC 19790:2012 and the test metrics specified in this document for each of the associated security functions addressed in ISO/IEC 19790:2012.

The test approach employed in this document is an efficient "push-button" approach, i.e. the tests are technically sound, repeatable and have moderate costs.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 19790:2012, *Information technology — Security techniques — Security requirements for cryptographic modules*

ISO/IEC 24759:2017, *Information technology — Security techniques — Test requirements for cryptographic modules*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 19790 and the following apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

3.1
advanced side-channel analysis
ASCA
advanced exploitation of the instantaneous side-channels emitted by a cryptographic device that depends on the data it processes and on the operation it performs to retrieve secret parameters

3.2
correlation power analysis
CPA
analysis where the correlation coefficient is used as the statistical method

3.3
critical security parameter class
CSP class
class into which a *critical security parameter* (3.3) is categorised

EXAMPLE Cryptographic keys, authentication data such as passwords, PINs, biometric authentication data.

3.4
differential electromagnetic analysis
DEMA
analysis of the variations of the electromagnetic field emanated from a cryptographic module, using statistical methods on a large number of measured electromagnetic emanations values for determining whether the assumption of the divided subsets of a secret parameter is correct, for the purpose of extracting information correlated to security function operation

3.5
differential power analysis
DPA
analysis of the variations of the electrical power consumption of a cryptographic module, for the purpose of extracting information correlated to cryptographic operation

3.6
electromagnetic analysis
EMA
analysis of the electromagnetic field emanated from a cryptographic module as the result of its logic circuit switching, for the purpose of extracting information correlated to security function operation and subsequently the values of secret parameters such as cryptographic keys

3.7
implementation under test
IUT
implementation which is tested based on non-invasive methods

3.8
power analysis
PA
analysis of the electric power consumption of a cryptographic module, for the purpose of extracting information correlated to the security function operation and subsequently the values of secret parameters such as cryptographic keys

3.9
side-channel analysis
SCA
exploitation of the fact that the instantaneous side-channels emitted by a cryptographic device depends on the data it processes and on the operation it performs to retrieve secret parameters

3.10
side-channel collision attack
powerful category of *side-channel analysis* (3.9) that usually combines leakage from distinct points in time, making them inherently bivariate

3.11
simple electromagnetic analysis
SEMA
direct (primarily visual) analysis of patterns of instruction execution or logic circuit activities, obtained through monitoring the variations in the electromagnetic field emanated from a cryptographic module, for the purpose of revealing the features and implementations of cryptographic algorithms and subsequently the values of secret parameters

3.12
simple power analysis
SPA
direct (primarily visual) analysis of patterns of instruction execution (or execution of individual instructions), in relation to the electrical power consumption of a cryptographic module, for the purpose of extracting information correlated to a cryptographic operation

3.13
timing analysis
TA
analysis of the variations of the response or execution time of an operation in a security function, which can reveal knowledge of or about a security parameter such as a cryptographic key or PIN

4 Symbols and abbreviated terms

ASCA	advanced side-channel analysis
AES	advanced encryption standard
CPA	correlation power analysis
CSP	critical security parameter
DEMA	differential electromagnetic analysis
DES	data encryption standard
DLC	discrete logarithm cryptography
DPA	differential power analysis
DSA	digital signature algorithm
ECC	elliptic curve cryptography
ECDSA	elliptic curve digital signature algorithm
EM	electromagnetic
EMA	electromagnetic analysis
HMAC	keyed-hashing message authentication code
IFC	integer factorization cryptography
IUT	implementation under test
MAC	message authentication code
PA	power analysis
PC	personal computer
PCB	printed circuit board
PKCS	public-key cryptography standards
RBG	random bit generator
RNG	random number generator

RSA	Rivest Shamir Adleman
SCA	side-channel analysis
SEMA	simple electromagnetic analysis
SHA	secure hash algorithm
SNR	signal to noise ratio
SPA	simple power analysis
USB	universal serial bus
TA	timing analysis
·	multiplication symbol

5 Document organization

Clause 6 specifies the non-invasive attack methods that a cryptographic module shall mitigate against for conformance to ISO/IEC 19790:2012.

Clause 7 specifies the non-invasive attack test methods.

Clause 8 specifies the test methods for side-channel analysis of symmetric-key cryptosystems.

Clause 9 specifies the test methods for side-channel analysis of asymmetric-key cryptosystems.

This document shall be used together with ISO/IEC 24759:2017 to demonstrate conformance to ISO/IEC 19790:2012.

6 Non-invasive attack methods

This clause specifies the non-invasive attack methods that shall [06.01] be addressed to ensure conformance with ISO/IEC 19790:2012.

The non-invasive attacks use side-channels (information gained from the physical implementation of a cryptosystem) emitted by the implementation under test (IUT), such as:

— the power consumption of the IUT,

— the electromagnetic emissions of the IUT,

— the computation time of the IUT.

The number of possible side-channels can increase in the future (e.g. photonic emissions,[49] acoustic emanations).

In order to be more formal in the taxonomy of the attacks, a formalism allows the relationships to be highlighted between the different attacks and to have a systematic way to describe a new attack.

An attack is described in the following way:

<KKK>-<YYY>-<XXX>-<ZZZ>-<TTT>

KKK refers to the order of the attack (e.g. “2O” for second order attack).

YYY refers to the statistical treatment used in the attack (e.g. “S” for simple, “C” for correlation, “MI” for mutual information, “ML” for maximum likelihood, “D” for difference of means, “LR” for linear regression, etc.).

NOTE 1 Other statistical treatments can be inserted like “dOC” which corresponds to a correlation treatment exploiting *d*th order moments (obtained for instance, by raising each targeted point in the traces to a power *d*, or by combining *d* points per trace before processing the correlation).

XXX refers to the kind of observed side channel: e.g. “PA” for power analysis, “EMA” for electromagnetic analysis, “TA” for timing analysis, etc.

ZZZ can refer to the profiled (“P”) or unprofiled (“UP”) characteristic of the attack. This is optional and the default value is “UP”.

TTT refers to the direction of the attack (e.g. “V” for vertical, “H” for horizontal,[43] “R” for rectangle).

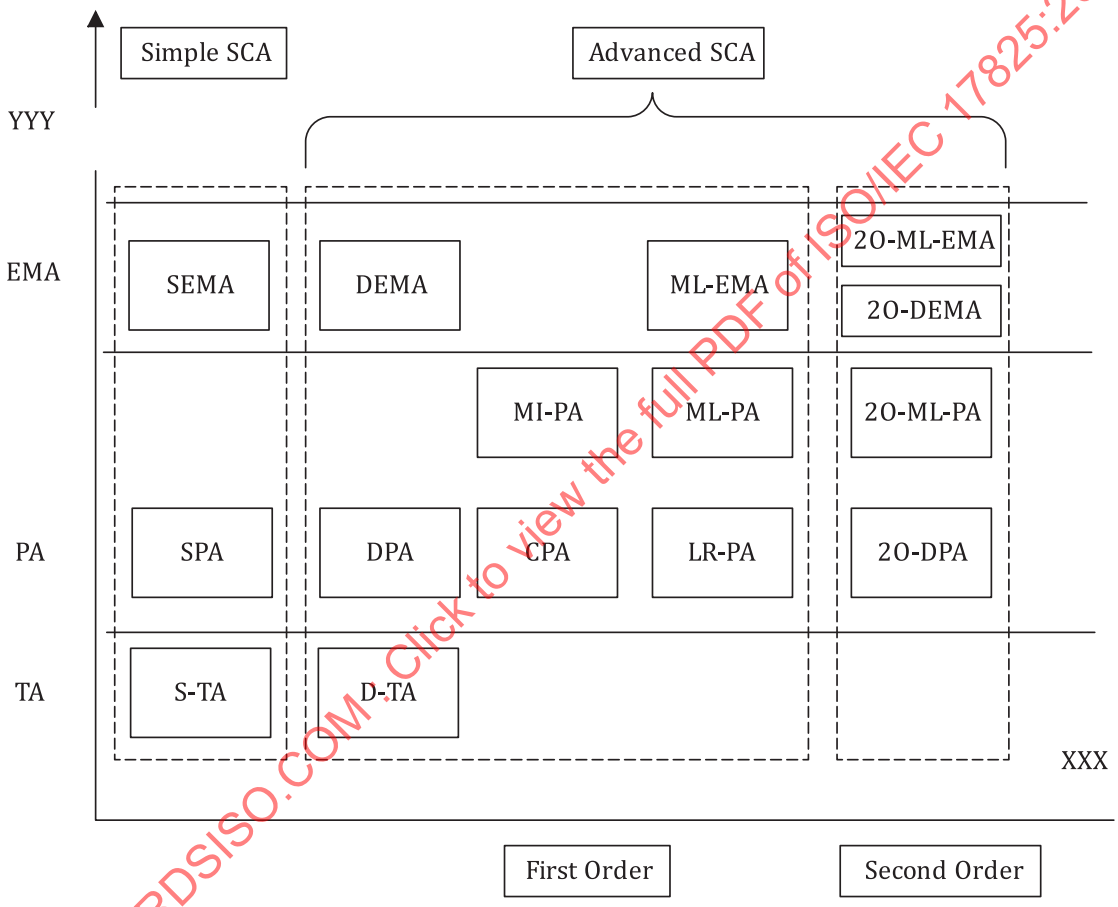

Figure 1 — Taxonomy of non-invasive attacks

NOTE 2 Instead of just splitting advanced side-channel analysis (ASCA) into univariate and multivariate cases, the classification can still be refined by separating attacks based on “variable distinguishers” (which focus on a particular moment of the distribution of the target variable) from those based on “pdf distinguishers” (non-invasive analysis distinguisher which requires as input an estimation of the leakage probability density function knowing the secret key). The first category includes ASCA based on correlation or on the linear regression techniques. The second one includes maximum likelihood and mutual information attacks for instance.

NOTE 3 The simple power analysis (SPA) and simple electromagnetic analysis (SEMA) attack methods include some extensions to basic SPA and SEMA attacks (i.e. template attack). The differential power analysis (DPA) and differential electromagnetic analysis (DEMA) attack methods include some extensions to basic DPA and DEMA attacks [i.e. correlation power analysis (CPA) and higher-order DPA attacks]. It is not mandatory to test them in this document.

The taxonomy of non-invasive attacks is illustrated in Figure 1. The scope of this document focuses on first-order attacks, i.e. the first two columns of Figure 1. Emerging non-invasive attacks and side-channels are described in Annex D but are not applicable currently as required test method in this document.

The variables used in the description of ASCA are:

A	cryptographic processing
C	observation processing
D	number of predictions
d_C	multivariate degree
d_D	multivariate degree
d_o	dimension of observation
F	function, i.e. manipulation
h	observation
i	index
K	secret key
k_1	sub key 1
k_2	sub key 2
M	model of leakage
N	number of observations
o_i	observation interval
$(o\_i)\_i$	observation interval number i
$pred\_i$	prediction
t_i	i iteration of time
x_1_i	i iteration of x_1
x_2_i	i iteration of x_2
X	known data

ASCA is described in the following steps:

1) Measure N observation intervals o_i related to a cryptographic processing A parameterized by a known input X and a secret key K.

2) (Optional) Choose a model of leakage M for the device leakage.

3) (Optional) Choose an observation processing C (by default C is set to the identity function).

4) Make all hypothesis h on the value of K or a subpart of it.

5) Select as the most likely key the hypothesis with the largest statistical test.

NOTE 4 The observations o_i can be univariate or multivariate. In the latter case, each coordinate of o_i, viewed as a vector, corresponds to a different time t_i. The dimension of o_i is denoted by d_o in the rest of this note.

NOTE 5 In side-channel collision attacks against block ciphers, the second step is skipped and the third step simply consists in a point selection in the traces o_i. Then, the hypothesis h typically corresponds to a hypothesis between the difference ($k1$-$k2$) of two parts of the targeted key K (e.g. two sub-keys in a block cipher implementation). Eventually, the predictions are deduced from the observations (o_i)_i and the difference h. If for instance the attack targets the manipulation of a value $F(x1\_i+k1)$ [i.e. $C(o\_i)$ corresponds to the part of the observation related to the manipulation of $F(x1\_i+k1)$], then the attack will extract from the o_i the observations during the manipulation of another values $F(x2\_i+k2)$. Those observations will be re-arranged such that $x2\_i - x1\_i = h$. Then h_i corresponds to the part of the observation related to the manipulation of $F(x2\_i+k2) = F(x1\_i+k1)$ if h is correct. To validate the hypothesis, a correlation coefficient is usually used for D. Additionally, all the attacks described in Clause 6 can be vertical, horizontal or rectangle (i.e. horizontal and vertical). An attack is said to be vertical if each observation o_i corresponds to a different algorithm processing. If all the o_i correspond to a same algorithm processing, then the attack is said to be horizontal. If some o_i share the same algorithm processing while some other o_i do not, then the attack is said to be rectangle. The classical attacks specified in literature are vertical and this modus operandi will hence be defined as the default one. Examples of attacks performed in the horizontal mode can be found in References [43] and [44].

NOTE 6 An approval authority can modify, add or delete non-invasive attack methods, the association with security functions (see Table C.1) and non-invasive attack mitigation test metrics specified in this document.

7 Non-invasive attack test methods

7.1 General

This clause presents an overview of the non-invasive attack test methods for the corresponding non-invasive attack methods specified in Clause 6.

7.2 Test strategy

The goal of non-invasive attack testing is to assess whether a cryptographic module utilizing non-invasive attack mitigation techniques can provide resistance to attacks at the desired security level. No standardized testing programme can guarantee complete protection against attacks. Rather, effective programmes validate that sufficient care was taken in the design and implementation of non-invasive attack mitigations.

Non-invasive attacks exploit a bias latent in the physical quantities which are non-invasively measured on or around the IUT. Such a bias is induced from and depends on the secret information that the attacks target. For further details, see Reference [16]. The bias can be subtle but is generally persistent. In this document, the biased information that depends on the secret information is referred to as leakage hereinafter. A device can fail one or more tests if experimental evidence suggests that leaking information exceeds permitted leakage thresholds. This implies that leakage demonstrates a potential vulnerability. Conversely, attacks fail and the test passes unless leakage is observed. The test of existence of leakage is called leakage analysis (leak analysis) hereinafter.

The goal is to collect and analyse measurements within certain test limitations such as maximum waveforms collected, elapsed test time, and to determine the extent of the CSP information leakage. The test limitations and leakage thresholds constitute the test criteria. The maximum acquisition time shall [07.01] also be bounded. The values for security Level 3 and Level 4 are detailed in Annex A.

Consider timing the attack testing. If the test reveals that the computation time is biased relative to the CSP, the IUT fails. For DPA, if the test reveals that the power consumption during CSP-related processes is biased relative to the CSP, the IUT fails. The testing approach uses statistical hypothesis testing to determine the likelihood that a bias is present. Thus, this document provides a leakage threshold in terms of statistical significance. The test fails if a bias exceeds the leakage threshold. The pass/fail conditions for the desired security level are given in Annex A.

7.3 Side-channel analysis workflow

7.3.1 Core test flow

The tester collects measurement data from the IUT and applies a suite of statistical tests on the collected data. See Annex B for the requirements for measurement apparatus. Core test refers to testing for a single security function with a single critical security parameter (CSP) class, where CSP classes include cryptographic keys, biometric data or PINs. If some security functions deal with more than one CSP class, leakage analysis for every applicable CSP class is performed for each security function. The test method requires repeating core tests with different CSP classes until the first fail of test occurs or all the CSP classes pass. If a core test is unable to continue if the IUT limits the number of repeated operations, the result is a pass and the core test is continued with the next CSP class. The core test is shown in Figure 2. The side-channel resistance test framework is depicted in Figure 3. Leakage analysis for TA is shown in Figure 4, SPA/SEMA in Figure 5 and DPA/DEMA in Figure 6.

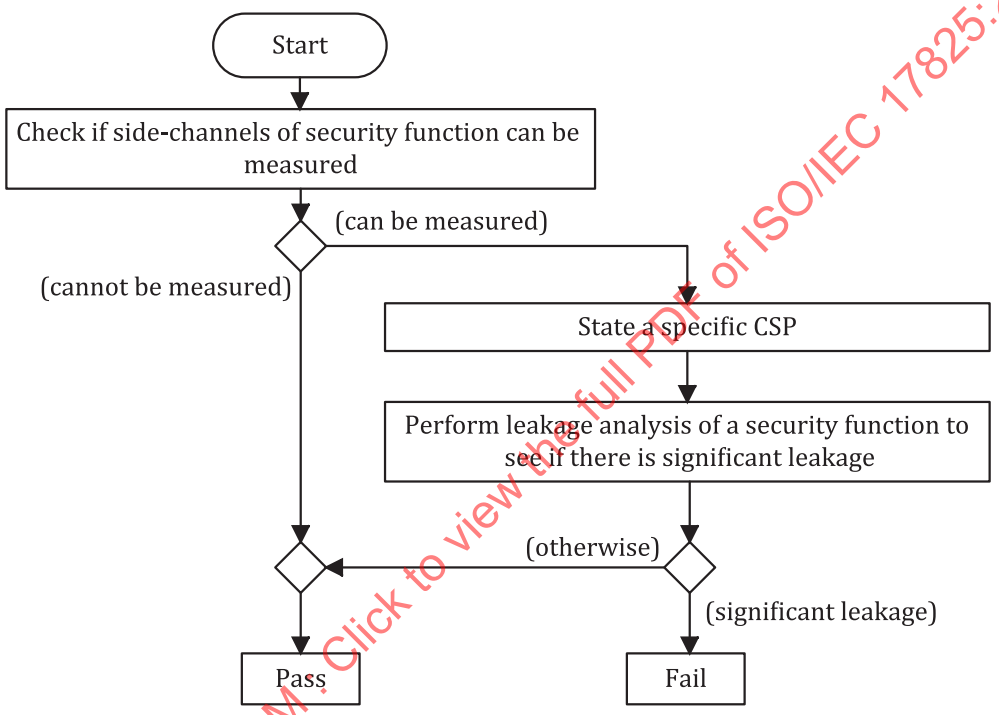

Figure 2 — Core test flow

Figure 2 shows the flow of a core test. First, the vendor document is verified for the specified CSP class. Second, the practicality of measuring the physical characteristics is determined. If the measurement cannot be made, the test result is pass. The testing laboratory shall [07.02] provide a reason why the side-channel is not measurable. A list of accepted reasons for a laboratory to assess a side-channel as not measurable is given in Annex G. Third, a set of CSPs determined by the testing laboratory is configured into the IUT. Finally, the essential part of the core test, the analysis, which is shown in Figures 3, 4, 5, 6, 7, 8, and D.1, is performed and significant leakage is either observed or not.

7.3.2 Side-channel resistance test framework

As explained in 7.3.4, 7.3.5 and 7.3.6, a testing laboratory shall [07.03] check the security of IUTs against TA, SPA, and DPA.

The sequential test of the three attacks leads to the attack framework depicted in Figure 3. The testing laboratory should follow the order of the operations. For example, the SPA can be tested only if TA passed.

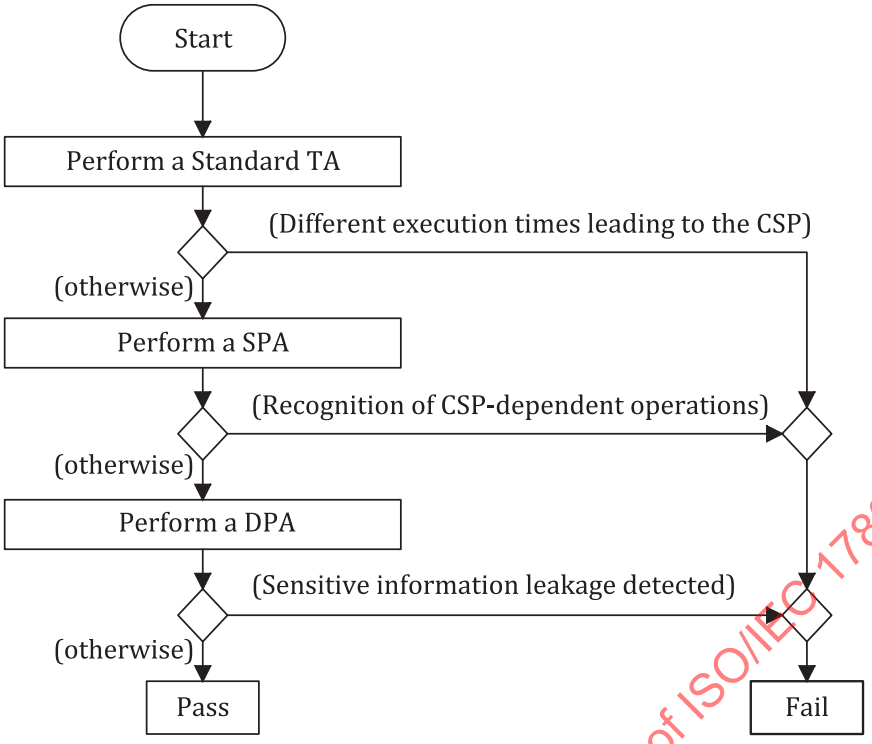

Figure 3 — Side-channel resistance test framework

The proposed methodology for side-channel resistance assessment does not require full key extraction to fail a device: an IUT can fail if significant sensitive information leakage can be demonstrated. Nevertheless, it should be noted that the primary purpose of the pass/fail criteria is that the IUT can be failed only if there is a risk of revealing the CSP/sensitive information.

7.3.3 Required vendor information

The vendor shall [07.04] provide the following information about the algorithms and countermeasures implemented in the IUT:

a) implemented cryptographic algorithms;

b) design of the implementation;

c) the conditions/mode(s) of usage where the IUT is susceptible to side-channel analysis.

Moreover, the testing laboratory shall [07.05] be able to modify CSPs and cipher text when performing side-channel testing.

When performing side-channel analysis, it is common to perform signal alignment so that different traces can be compared at the same point during the cryptographic calculation. For the purposes of side-channel testing, the vendor should provide the testing laboratory with the best synchronization signal for the start of the cryptographic operation. For example, in testing mode the device can provide an external trigger point to indicate when the cryptographic operation starts or stops. If such start and stop information is not available, the testing laboratory should adopt standard signal processing- and matching-based techniques to perform alignment. In cases where traces are well aligned at the start of the cryptographic operation, the laboratory can be required to use standard signal matching to perform better alignment on specific internals of the algorithm; the number and locations of these alignment points are specified by the testing laboratory.

The vendor should then provide a function that allows the testing laboratory to:

d) synchronize its measurements,

e) check the quality of its measurements (see 7.3.6 for more details).

7.3.4 TA leakage analysis

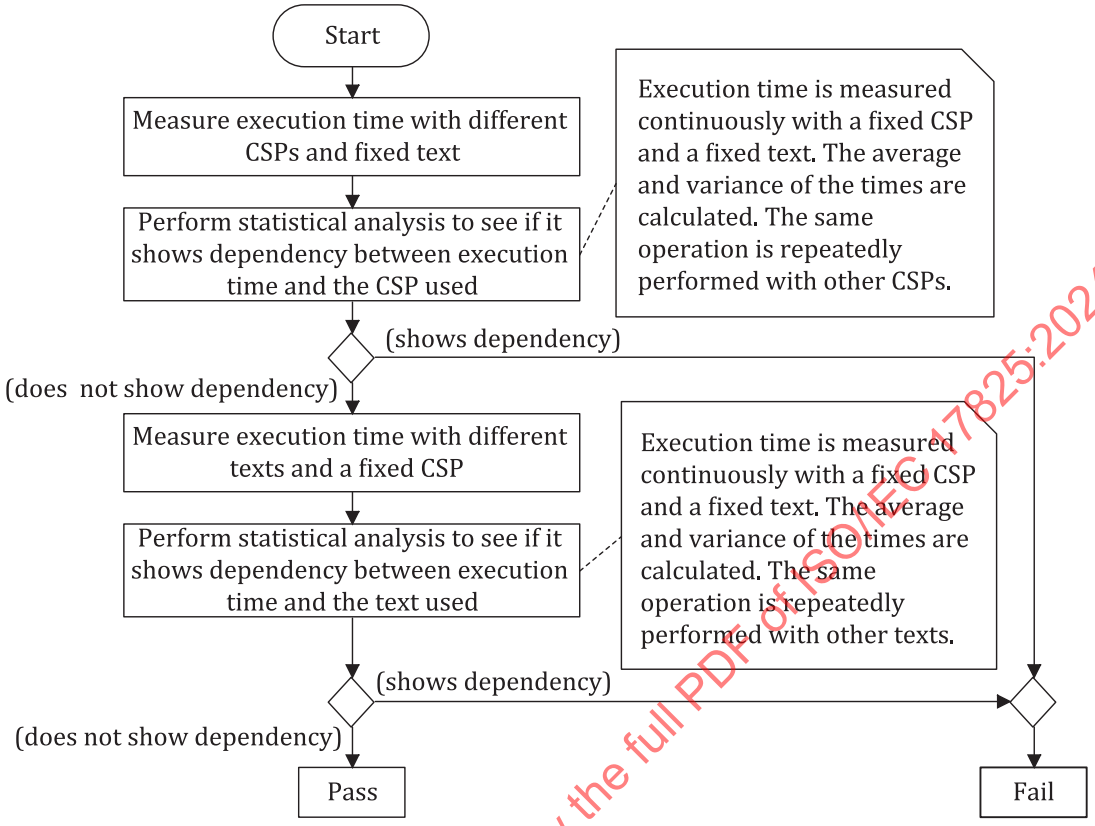

Figure 4 — Leakage analysis for timing attacks

Figure 4 shows the leakage analysis flow for timing attacks. The flow can be divided into two stages. For the first stage, execution times with several different CSPs and fixed text are measured. If the measured execution time does not show dependency with the CSP used through statistical analysis, then the test continues to the second stage. Otherwise, the test fails. For the second stage, execution times with several different texts and a fixed CSP are measured. If the measured execution time does not show dependency with the text used, the test passes. Otherwise, the test fails. If the execution time is difficult to measure, a tolerance value ε which equals a clock cycle related to the algorithm co-processer of the targeted chip should be used. To compare the two time values (or two average time values) T_1 and T_2, the test passes if $|T_1 - T_2| < \varepsilon$, and fails otherwise. Timing analysis shall [07.07] be performed with a sufficient number of measurements. Detailed requirements for security level 3 and level 4 are given in Annex A.

Not only the difference of means, but also of variances, shall be computed, so as to detect second-order timing leakage. Indeed, high-order timing attacks are practical threats.[58]

7.3.5 SPA/SEMA leakage analysis

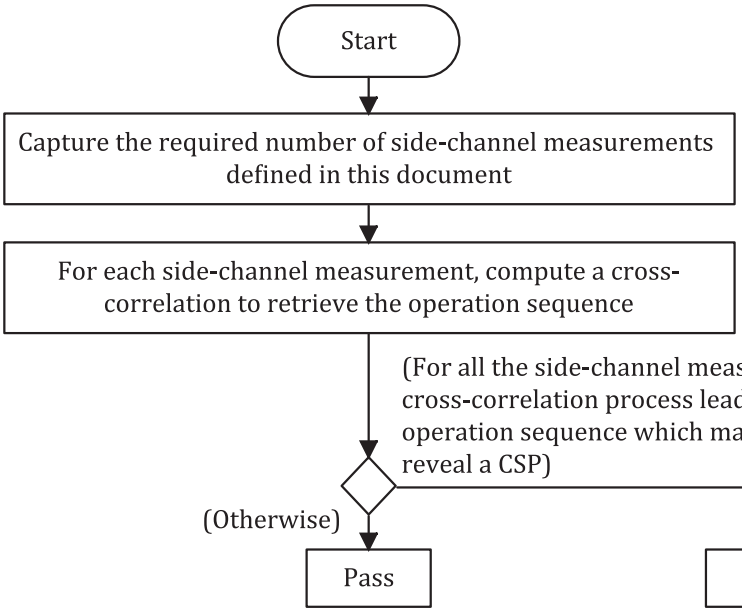

Figure 5 — SPA (SEMA) leakage analysis

Figure 5 shows the SPA/SEMA leakage analysis flow. The flow can be divided into two stages.

First, the testing laboratory shall [07.08] capture the number of side-channel measurements related to the desired security level, as specified in Annex A.

Asymmetric cryptography repeatedly uses elementary operations. For RSA these are modular square (denoted "S") and multiply (denoted "M") operations. For ECC, these are point doubling and addition operations. Since the key can be derived from the order of operations, it is important for the testing laboratory to distinguish these operations. As side-channel measurements can be noisy (see Annex E for quality criteria for measurement setups), it can be difficult to recognize these operations visually. A good method to identify a repeating operation is called "cross-correlation". This method also helps to remove subjective assessment from the testing laboratory. When the correlation is so weak that no definite statement can be taken, the testing laboratory can mount a cluster analysis.

For all the side-channel measurements, if the cross-correlation process leads to a non-regular operation sequence which leads to the CSP, the test result is fail. It is important to note that the IUT can only be failed if the CSP has been revealed from the non-regular operation sequence.

7.3.6 DPA/DEMA leakage analysis

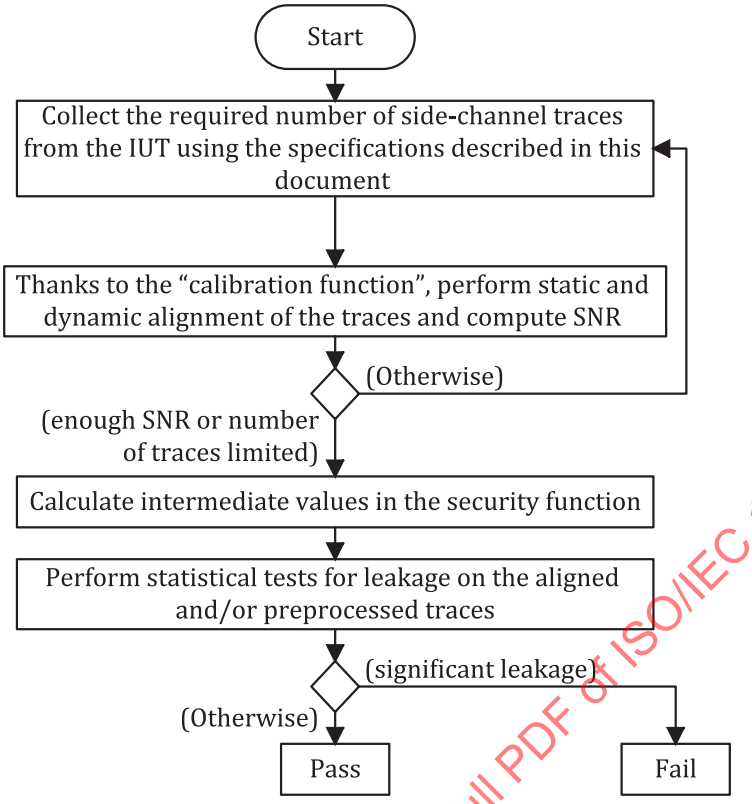

Figure 6 — DPA (DEMA) leakage analysis

Figure 6 shows the DPA/DEMA leakage analysis flow. The test lab shall [07.09] collect enough traces. The method for calculating the number of required traces is sketched in Figure 7 and detailed in Annex A. The result represented indicates whether or not significant leakage has been observed. In case of doubt, clear box or white box are required to clarify the ambiguity (see Reference [45]). Indeed, if the test is a Student's *t*-test, not all leakages are sensitive: typically, possible test violations are incurred by non-CSP variables, such as the plaintext or the ciphertext of a block cipher (see Annex H). Therefore, based on the analysis of the IUT documentation, it can be decided whether test violations depend on the CSP or not. The same expressions apply to Figure 6.

As a general rule, it is supposed that the cryptographic operations occur always in the same moment in each measurement (consumptions or emanations). Nonetheless, the developers have the possibility to include internal clocks modifying the operation frequency or introduce randomly non-operative wait status in the algorithms execution, thus the time is no longer constant and the cryptographic operations are not performed in the same instant. This produces the well-known misalignments in the set of traces, making the analysis difficult and much more costly in terms of the number of traces needed to be processed. These modifications of the original behaviour are countermeasures implemented by the developers to counteract the possibility of acquiring information through side channels, breaking the assumptions that characterize the known attacks.

In cryptographic implementations without specific countermeasures, misalignments come from errors in the measurement configuration, different clock domains, bus contention, OS interrupts, etc., when starting the power consumption (or emanations) acquisition. In this case, the traces can be aligned if the vagueness can be determined when launching the measurement, properly displacing the traces. This process is called static alignment. This vagueness can also be mitigated or, at least, facilitates the alignment, if a trigger is provided (or exists) signalling when the operation starts.

When the implementation actively introduces random timing delays or clock frequency variations, the static displacement cannot attain the full alignment of the traces. In this case, the so called dynamic alignment is applied by matching parts of traces with different displacements and performing a nonlinear sampling of

the traces. After this process, the different parts along the traces are located in the same positions (e.g. the location of the different rounds matches in all the traces).

The vendor should collaborate with the testing laboratory by implementing in the IUT a function that helps the testing laboratory to synchronize the waveforms (by providing a trigger signalling the beginning of the cryptographic operation) and to check the quality of its side-channel measurements. A function should be provided by the vendor for each countermeasure. This can also help to test external noise reduction methods (filtering in frequency, mean calculation, etc.). This function can be simply the processing/storage of a known public (non-sensitive) variable in the IUT (e.g. the public key *e* in RSA). The testing laboratory should retrieve this known value. If there is sufficient signal to noise ratio (SNR) of the side-channel measurements portion that corresponds to the processing of this known value, the testing laboratory can perform the tests. If not, the testing laboratory should find a way to improve the quality of its measurements before performing the tests.

Pre-processing conditions in differential analysis given in Annex A shall [07.10] be met.

The testing laboratory shall [07.11] then calculate intermediate values in the security function. It is feasible since the testing laboratory collects side-channel measurements using a pre-specified set of test vectors. These test-vectors are carefully chosen using methods like the chosen-input method (see Annex F) by the testing laboratory to expose and isolate potential leakages.

The last step consists in performing statistical tests for leakage on the aligned and/or pre-processed traces. The testing laboratory should apply a simple statistical test (e.g. Welch's test) to multiple, pre-specified data sets in order to detect sensitive information leakage in the side-channel.

Clauses 8 and 9 respectively describe the guidelines for assessing the side-channel attack resistance of symmetric and asymmetric cryptosystems.

8 Side-channel analysis of symmetric-key cryptosystems

8.1 General

This clause focuses on side-channel analysis of symmetric-key cryptosystems. The framework depicted in Figure 3 is used. Resistance against timing attacks, simple side-channel analysis, and differential side-channel analysis shall [08.01] be assessed.

8.2 Timing attacks

For (software) symmetric-key cryptosystems, the only known threat related to timing attacks concerns cache-timing attacks.[50] They rely on the micro-architectural properties of the CPU (e.g. cache architecture, branch prediction unit). Cache attacks exploit the cache behaviour (i.e. cache hit/miss statistics) of cryptosystems. Cache architecture leaks information about memory access patterns. The execution time is a source of leakage (cache misses take more time to execute than a cache hit). Cryptosystems have dependant memory access patterns. Once the access patterns are extracted, the testing laboratory can recover the secret key.

If the IUT is a software/firmware implementation of a symmetric-key cryptosystems, and if the IUT contains a cache-memory, the testing laboratory can test the IUT against timing attacks following the framework described in Reference [50]. In the contrary case, the test result is pass.

8.3 SPA/SEMA

8.3.1 Attacks on key derivation process

For symmetric-key cryptosystems, the only known threat related to SPA or SEMA attacks concerns key derivation process (key schedule). If the testing laboratory can determine the Hamming weights of intermediate values that occur in a symmetric-key cryptosystem, it allows the key to be revealed. For example, for AES,[53] the testing laboratory can use the dependencies between the bytes of the round keys

within the AES key schedule to reduce the number of possible key values. The key is then determined by using a known plaintext-ciphertext pair.

If the IUT contains a key derivation process, the testing laboratory can try to apply some known methods to extract the key (e.g Reference [53] for AES), or any other relevant method.

8.3.2 Side-channel collision attacks

Side-channel collision attacks also make use of the fact that side-channel measurements of the IUT depend on the processed data. Cryptographic security functions include some steps to produce intermediate values from input value and cryptographic key. If the intermediate values become the same in value against different input values, the resultant power consumption or electromagnetic emanation would be quite similar. This kind of "collision" can be exploited in order to reduce the key space (see References [38] to [42]).

The testing laboratory can check if the IUT is susceptible to such side-channel collision attacks, by following a known framework (e.g Reference [39] for AES) or any other relevant method.

8.4 DPA/DEMA

The statistical test shall [08.02] be made according to the following. The side-channel traces are divided into two subsets such that the sensitive information being processed is significantly different between the two subsets.[20] This partitioning is feasible since the cryptographic algorithms are performed with known parameters and data, and all intermediate states are known.

If the side-channel traces in the two subsets are statistically different with high confidence, then information leakage is present and the device fails (the leakage should be within the sensitive boundary, see Annex H). Otherwise, information leakage is either not present or is suppressed.

The statistical tool which is used is the Welch *t*-test. A high positive or negative value of the *t*-test statistic value T (defined below) at a point in time indicates a high degree of confidence that the null hypothesis (i.e. the two subsets means are equal) is incorrect. A confidence level is arbitrary between 0 and 1, but normally chosen close to 1 indicating the high confidence based on the desired criteria, such as 0,99 (or 99 %). The confidence level determines the threshold value C for the positive and negative threshold ($+C/-C$) for T with the t distribution. For example, the confidence level 99,99 % corresponds to C = 3,9 and 99,999 % corresponds to C = 4,5.

In order to control the false positives, multiplicity corrections are required. In this document, Bonferroni correction is preferred. That is, per-test significance level obtained by dividing the desired overall significance level by the number of tests m, i.e. $\alpha_{\text{per-test}} = \frac{\alpha}{m}$.

For each algorithm, multiplicity corrections shall [08.03] be performed, each targeting a different type of leakage. Figure 7 describes the general statistical test procedure.

Before processing the statistical test, the testing laboratory shall [08.04] specify the parameters such as standardised effect size, false positive rate α, false negative rate β, etc. Example values for α and β can be α = 0,05, α = 0,000 01, β = 0,05, etc. The tester shall [08.05] also carry out the multiplicity corrections to get the per-test significance level.[62] Then the tester shall [08.06] calculate the number of traces $N = N_A + N_B$ required for the test.

The following notations are denoted:

N_A, N_B are the size of the subsets *A* and *B*;

N is the sample size, which is N_A+N_B;

μ_A is the average of all the traces in group *A*;

μ_B is the average of all the traces in group *B*;

σ_A is the sample standard deviation of all the traces in group *A*;

σ_B is the sample standard deviation of all the traces in group *B*;

T is $\frac{(\mu_A - \mu_B)}{\sqrt{\frac{\sigma_A^2}{N_A} + \frac{\sigma_B^2}{N_B}}}$;

α is significance level (or false positive) ;

β is false negative;

d standardized effect size.

For the simple case of a *t*-test with equal sample sizes, the formula for required sample size can be expressed in terms of the standardised effect size, as shown in Formula (1):

$$N = 4 \cdot \frac{\left(Z_{\alpha/2} + Z_{\beta}\right)^2}{d^2} \quad (1)$$

In addition to the *t*-test, the NICV (Normalized Inter-Class Variance, aka coefficient of determination)[60][61] can be used. The advantages are:

a) it can be multibit;

b) it is comparable between implementations, as it is bounded between 0 and 1;

c) it relates to the Pearson correlation coefficient ρ like $0 \leq \rho^2 \leq \text{NICV} \leq 1$;

d) it generalizes naturally to high-order leakage.

Guidance on the choice for parameters α (alpha) and β (beta) is as follows:

The value of beta should be set low, as the main purpose of side-channel analysis is to detect a (latent) vulnerability; a leakage should not be missed by relaxing the constraint on beta.

On the other hand, the choice of alpha is more flexible. By being conservative (low alpha), the tester requires less analyses of false positives (though at the expense of more traces). At the opposite, with a larger alpha, some human check should be performed (to verify whether a potential leakage is real or an artefact), but less traces are required.

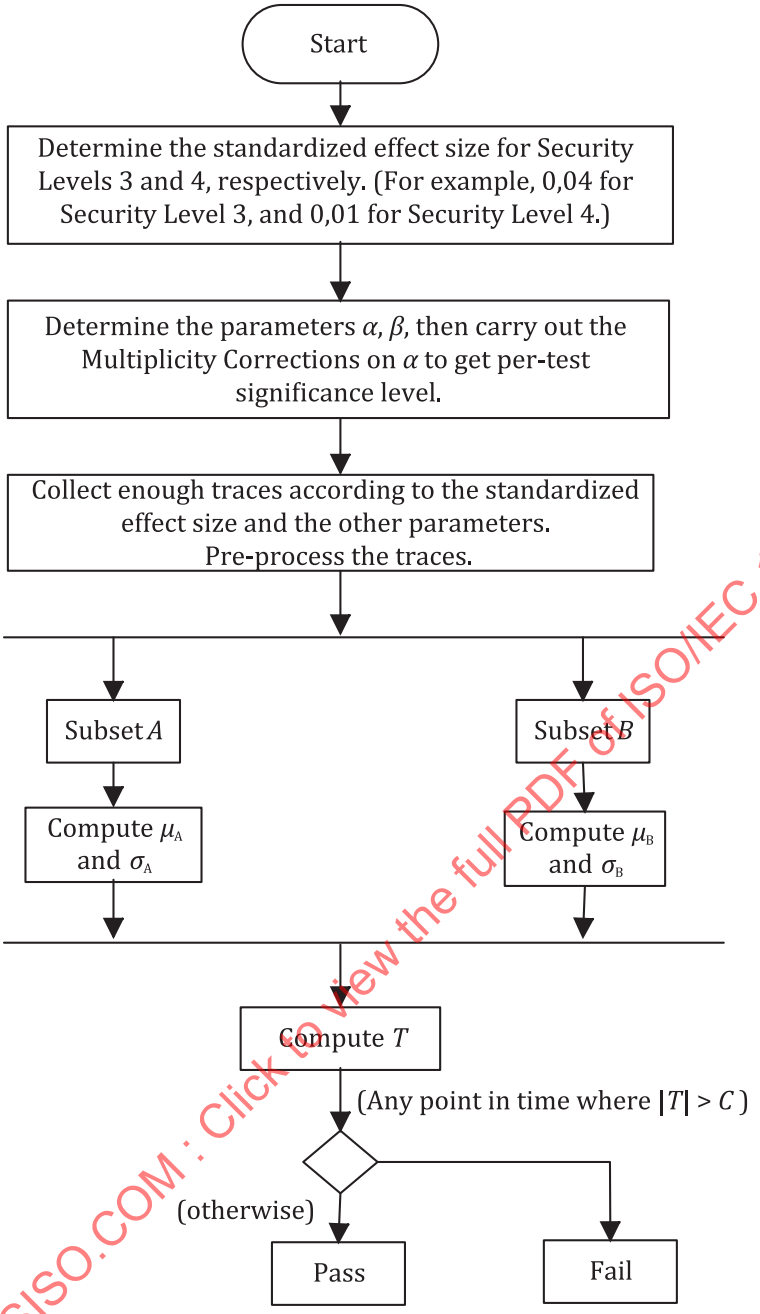

Figure 7 — General statistical test procedure

9 ASCA on asymmetric cryptography

9.1 General

This clause focuses on side-channel analysis of asymmetric-key cryptosystems.

Asymmetric algorithms can fulfil three different tasks: signature, encryption and key agreement. As pointed out in Table C.1, the most used asymmetric algorithms are RSA and elliptic curve cryptosystems (ECC).

For signature, encryption or key agreement, the main operation is the computation of:

a) a modular exponentiation in the case of RSA; that is the computation of m^d mod n for some integer m, private key d and an integer n, and

b) an elliptic curve scalar multiplication in the case of ECC; that is the computation of $d \cdot p$ for a point on the given elliptic curve and the private key d.

There are many possibilities to compute a modular exponentiation and an elliptic curve scalar multiplication. The actual exponentiation algorithms depend on the integers' representation and on the arithmetic modular operations.

NOTE Actual exponentiation algorithm implementations also depend on the performance, complexity, compactness and targeted security level, and therefore result from a necessary trade-off.

The framework depicted in Figure 3 is used. Resistance against timing attacks, simple side-channel analysis, and differential side-channel analysis shall [09.01] be assessed.

Regarding ECC, a complete survey of side-channel attacks is available in Reference [59].

9.2 Detailed side-channel resistance test framework

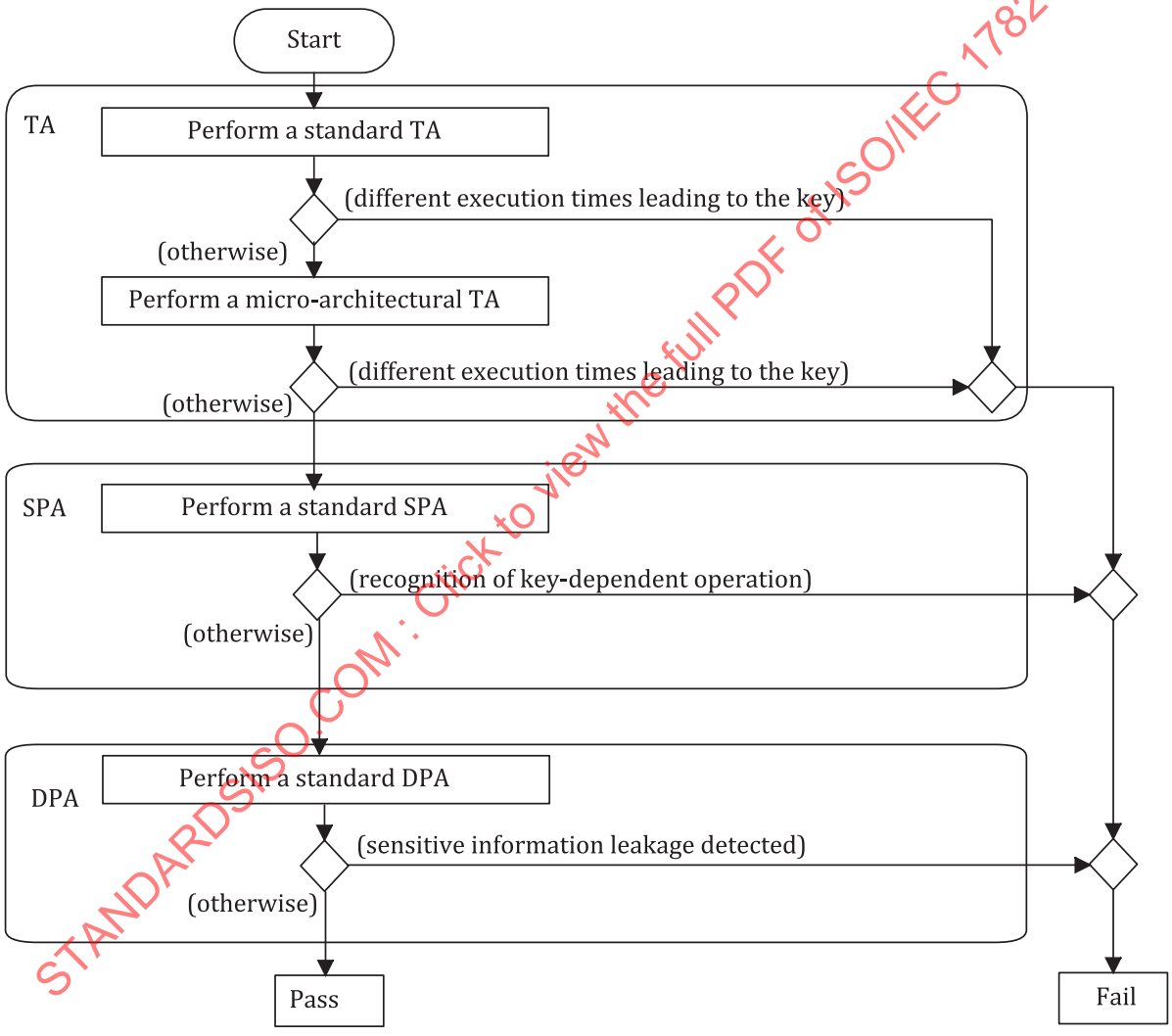

Figure 8 — Side-channel resistance test framework for asymmetric-key cryptosystems

Compared to symmetric-key cryptosystems, there are more ways to attack asymmetric-key cryptosystems due to the varieties of structure and underlying arithmetic. Figure 8 describes the side-channel resistance test framework for asymmetric-key cryptosystems. The security tests are aimed at checking the security against conventional attacks such as the TA, SPA, DPA. More sophisticated attacks like Markov SPA, address-bit DPA and doubling attacks are given in Annex D. The relevant approval authority can choose whether to perform security tests against these attacks. Each of these is a particular attack on asymmetric ciphers.

9.3 Timing attacks

9.3.1 General

With respect to Figure 8, the testing laboratory should assess the security of asymmetric-key cryptosystems against standard and micro-architectural timing analysis.

9.3.2 Standard timing analysis

Since asymmetric-key cryptosystems compute key-dependant operations and can use basic mathematical operations with variable time computations (e.g. modular multiplication), they can be vulnerable to timing attacks. The following details the method described in Reference [2] to attack RSA with a timing attack.

In Reference [2] attacks are done on an IUT which implements:

— a modular multiplication (Montgomery method) that shows computation time variations;

— the standard square-and-multiply exponentiation routine that allows these variations to be exploited.

The result of each multiplication lies in $[0,2N-1]$, where N is the modulus.

Concerning the exponentiation routine, the IUT computes a multiply step when the current secret key bit d_I is equal to 1. If the result of the multiply step is greater than N, a subtraction by N is computed.

In Reference [2], the testing laboratory shall [09.02] have knowledge of secret key bits d_k-1 to d_k-i+1 when attacking d_k. Knowing the message, the intermediate value after the square step (called s) at iteration $k - i$ is computed. It can be stated whether the subtraction in the multiply step is required.

The attack is based on an oracle. The oracle is a clone of the IUT, in which the tester can change the CSP. The testing laboratory shall [09.03]:

a) sign with same (d, N) for many random messages;

b) make the assumption that $d\_k-i = 1$;

c) construct two sets of messages depending on the fact that the subtraction happens (set A) or not (set B) during the multiplication.

In case:

— $d\_k-i = 0$, global times for sets A and B are not statistically distinguishable (the split is based on a multiplication which does not occur);

— $d\_k-i = 1$, global times for sets A and B show a statistical difference related to the optional subtraction (the multiplication does occur).

Time measurements validate or invalidate the oracle. The testing laboratory shall [09.04] compute the mean of the global duration for each subset. $< A >$ (resp. $< B >$) is the mean global duration for messages A (B). The oracle criterion is the following:

— If $< A > - < B > > 0$, then the oracle was right ($d\_k-i = 1$);

— If $< A > - < B > = 0$, then the oracle was wrong ($d[k-i] = 0$).

If the testing laboratory finally retrieves the entire key with this method (coming from Reference [2]) or any other relevant one, the test result is fail. Otherwise, the test result is pass.

NOTE RSA PKCS#1 v2.1, DSA and ECDSA are not vulnerable to the standard timing analysis since:

— the used padding for RSA PKCS#1 v2.1 is probabilistic and the attacker cannot predict on the intermediate values;

— DSA and ECDSA use an ephemeral exponent and scalar respectively, so the attacker cannot target a specific bit.

9.3.3 Micro-architectural timing analysis

(Software) asymmetric-key cryptosystems are also vulnerable to micro-architectural attacks. This side-channel is enabled by the branch prediction capability common to all modern CPUs. The penalty paid (extra clock cycles) for a mispredicted branch can be used for cryptanalysis of cryptographic primitives that employ a data-dependent program flow. (Software) asymmetric-key cryptosystems are then susceptible to so-called "Branch Prediction Analysis"[52]. The testing laboratory should test the IUT against Timing (Micro-Architectural) Attacks following the framework described in Reference [52], or any other relevant method.

9.4 SPA/SEMA

With respect to Figure 8, the testing laboratory shall [09.05] assess the security of asymmetric-key cryptosystems against standard SPA.

Since asymmetric-key cryptosystems executes key-dependant sequence of operations, it is naturally vulnerable to "standard" SPA/SEMA.

For all the side-channel measurements, if the cross-correlation leads to a regular operation sequence, for example with an RSA (here, "M" represents a multiply operation, and "S" a square one):

— "MMMMMM…"

— "MSMSMS…"

then the test result is pass.

NOTE 1 Instead of random keys and inputs, the testing laboratory can choose particular keys and inputs:

— Keys: random, `0x80…01`, `0xff…ff`, `0xaa…aa` (in binary: 1010 1010 …).

— Inputs: random, low hamming weight, high hamming weight.

In an asymmetric cipher, the bits of the exponent/scalar highly influence the type of operation or data manipulated. The particular keys can help to distinguish the difference depending on the current bit, for example, to distinguish the differences of square and multiply (double from add for ECC), particular moving data for the square and multiply always (double and add always for ECC), bad use of jump instructions depending on the current bit, etc.

NOTE 2 An IUT using some SPA/SEMA countermeasures such as "atomic double-and-add algorithm" can leak the Hamming weight of the secret key. In some conditions it is sufficient to leak the entire key.[57] The testing laboratory can then optionally follow the methodology proposed in Reference [57].

9.5 DPA/DEMA

With respect to Figure 8, the testing laboratory shall [09.06] assess the security of asymmetric-key cryptosystems against standard DPA.

In the general case, the DPA/DEMA resistance test for asymmetric-key cryptosystems is similar to the symmetric-key ones except that there are much more leakage models.

DSA and ECDSA can be vulnerable to a specific DPA/DEMA when the private key is multiplied with a known number.

When the private key d is used for a multiplication by r, in this case, the attacker can perform a DPA/DEMA during the computation of $d \cdot r$.

A multiplication is generally performed word by word. The leakage model can be the Hamming Weight of the first word of d (d [0]) multiplied by the first word of r (r[0]), i.e. HW(d[0]·r[0]).

As for symmetric encryption, perform a t-test with the good hypothesis and a random wrong hypothesis.

The tests can be performed with different architecture sizes: 8, 16, 32 and 64.

It is not necessary to perform this test with all words of *d*. A deduction can be made that the other words are recovered the same way in case of a success and the other words are protected as well in case of a failure.

Annex A
(normative)

Non-invasive attack mitigation pass/fail test metrics

A.1 Introduction

This annex specifies the test metrics and pass/fail criteria for each associated combination of attack method and security function shown in Table C.1.

In this annex, test metrics are provided in data collection time, analysis time, and amount of data. The selection of the limits of the data collection time, analysis time, and amount of data should take into account the class of devices under test. Data sets used when testing a simple single-chip device are likely to be smaller than those required for meaningful analysis on a complex device, such as the complex System-On-Chip. The limits in this annex are more appropriate for simple single-chip devices.

A performance reference for computational analysis is given as the specification of the reference computer by the approval authority to provide fair criteria in analysis time.

Test metrics and pass/fail criteria from an approval authority can supersede this annex in its entirety.

A.2 Security level 3

A.2.1 Time limit

The maximum acquisition time shall be no more than 6 h for each elementary test for security level 3. When this limit is reached, the measurement is terminated even if the number of measurements has not met the specified maximum. The total sequence of acquisition time shall be no more than 72 h.

A.2.2 SPA and SEMA

To complete the SPA or SEMA test at security level 3, the provided test suite should collect:

— 11 waveforms using the input data patterns, each comprising a CSP and plaintext provided by the test suite (1 pre-determined input data pattern is used for the same-data-pair input; 1 pre-determined pair and 4 random-data pairs are used for the different-data-pair inputs).

For the analysis of each CSP bit, the resolution of each waveform is 100 points or greater.

The similarity of the resultant traces for a core test is inspected both visually and with a statistical test. To satisfy the core test, both test results shall be passed. The test will otherwise fail.

A.2.3 DPA and DEMA

To complete the DPA or DEMA test at security level 3, the provided test suite should collect:

— N waveforms of one type of side-channel leakage from different input data patterns for each CSP in a set of provided CSPs. Where N is calculated according to Formula (1), with d = 0,04.

If the calculated leakage is determined significant against the significance level pre-defined, the test will fail. Otherwise, the test will pass.

A.2.4 Timing analysis

To complete the timing analysis test at security level 3, the provided test suite shall collect:

— 1 000 timing measurements for random CSPs and a pre-determined plaintext for the first measurement block, and

— 1 000 timing measurements for a pre-determined CSP and random plaintext for the second measurement block.

A.2.5 Pre-processing conditions in differential analysis

To complete the differential power or emanation analysis at security level 3, the following factors are applicable:

— A synchronisation signal is available signalling the beginning of the cryptographic operation.

— A noise reduction shall [A.01] be performed by the tester calculating the mean of different traces (10 cryptographic executions should be performed for the same inputs set, to get one single trace).

A.2.6 Pass / fail condition

a) If the traces obtained using the trigger are misaligned from the different executions, the test passes and the statistical test for the cryptographic algorithm shall [A.02] not be performed.

b) If the traces obtained using the trigger are aligned, then the mean is to be computed and the verdict is the one obtained from the statistical test applied for the cryptographic algorithm and the traces.

A.3 Security level 4

A.3.1 Time limit

The maximum acquisition time shall be no more than 24 h for each elementary test for security level 4. When this limit is reached, the measurement is terminated even if the number of measurements has not met the specified maximum. The total sequence of acquisition time shall be no more than 288 h.

A.3.2 SPA and SEMA

To complete the SPA or SEMA test at security level 4, the provided test suite should collect:

— 21 waveforms using the input data patterns each comprising a CSP and plaintext provided by the test suite (1 pre-determined input data pattern is used for the same-data-pair input; 5 pre-determined pair and 15 random-data pairs are used for the different-data-pair inputs).

For the analysis of each CSP bit, the resolution of each waveform is 1 000 points or greater.

The similarity of the resultant traces for a core test is inspected both visually and with a statistical test. To satisfy the core test, both test results shall pass. Otherwise, the test will fail.

A.3.3 DPA and DEMA

To complete the DPA or DEMA test at security level 4, the provided test suite should collect:

— N waveforms of one type of side channel leakage from different input data patterns for each CSP in a set of provided CSPs. Where N is calculated according to Formula (1), with d = 0,01.

If the calculated leakage is determined significant against the significance level pre-defined, the test will fail. Otherwise, the test shall pass.

A.3.4 Timing analysis

To complete the timing analysis test at security level 4, the provided test suite should collect:

— 10 000 timing measurements for random CSPs and a pre-determined plaintext for the first measurement block, and

— 10 000 timing measurements for a pre-determined CSP and random plaintext for the second measurement block.

A.3.5 Pre-processing conditions in differential analysis

To complete the differential power or emanation analysis at security level 4 in addition to the factors specified for the security level 3, the following are applicable:

— A noise reduction is to be performed by the tester applying a spectrum analysis and filtering the traces in frequency (band-pass filter according to the module operation frequency).

— A static and dynamic alignment is performed to bypass any immediate countermeasure or misalignments coming from errors in the configuration measurement when starting the power consumption (or emanations) acquisition.

A.3.6 Pass / fail condition

Apply the filter in frequency.

a) If the traces present the inclusion of random timing delays or clock frequency variations so that they cannot be fully aligned with a static alignment, the test passes and the statistical test for the cryptographic algorithm shall not be performed.

b) If the static alignment succeeds, the verdict is the one obtained from the statistical test applied for the cryptographic algorithm and the traces filtered and aligned.

Annex B
(informative)

Requirements for measurement apparatus

B.1 General

This annex provides requirements and general information about capture characteristics.[20] Some of these requirements are not necessary to meet in some scenarios, as long as the signal to noise ratio is sufficient (see 7.3.6). For example, it is possible to perform effective attacks at much lower sample rates when sample clocks are synchronized,[63] or where leakage is at much lower frequencies common with public-key algorithms.[64]

B.2 Speed

a) Bandwidth shall [B.01] be at least 50 % of the device clock rate for software implementations and at least 80 % of the clock rate for hardware implementations.

b) There shall [B.02] be a capability to capture samples at 5 × the bandwidth.

B.3 Resolution

a) There shall [B.03] be a minimum of 8-bits of sampling resolution.

B.4 Capacity

a) Enough storage shall [B.04] be available to capture the entire signal required for the test and analysis.

B.5 Probe

A probe is required to measure the IUT currents.

If the used side-channel is the power consumption of the IUT, a resistor shall [B.05] be placed between the VCC line supplying the IUT and the IUT. The testing laboratory shall [B.06] choose the highest value resistor that allows the IUT to function.

If the used side-channel is electromagnetic emanations of the IUT, a near-field magnetic probe shall [B.07] be used, provided the bandwidth of the probe is at least that of the IUT clock rate.

Annex C
(informative)

Associated security functions

The non-invasive attack methods specified in Clause 6 are associated with the specific security functions that use the CSPs which are attacked by the target. The security functions are listed in ISO/IEC 19790:2012, Annexes C, D and E.

The associations are shown in Table C.1. Other non-invasive attacks and other associations between the attack methods and security functions can exist, but defence against them is not currently addressed in this document.

This does not preclude the use of approval authority approved non-invasive attacks.

A list of non-invasive attacks and other associations from an approval authority can supersede this annex in its entirety.

Table C.1 — Associations between non-invasive attack methods and security functions covered by this document

Security functions		Non-invasive attack methods		
		SPA/SEMA	DPA/DEMA	TA
Symmetric-key	AES	A[16]	A[16]	A[50]
	Triple-DES	A[24]	A[15]	A[23]
	Stream Ciphers	A[63]	A[63]	A[63]
Asymmetric-key	Plain RSA (Key wrapping)	A[15]	A[15]	A[1]
	RSA PKCS#1 v1.5	A[15]	A[15]	A[1]
	RSA PKCS#1 v2.1	A[15]	NKR	NKR

Key

A Applicable

NKR No known reference

NOTE 1 Applicable means that the security functions are susceptible to these types of attacks in public literature.

NOTE 2 No known reference means that the security functions are not known to be susceptible to these types of attacks in public literature, which does not mean that the attacks are not applicable.

NOTE 3 An HMAC implementation can be compromised by applying DPA/DEMA, however block-cipher based MAC is covered through AES and/or triple DES.

NOTE 4 RSA PKCS#1 v1.5 can be compromised by applying DPA/DEMA since the used padding is deterministic.

NOTE 5 Timing attacks on RSA PKCS#1 v2.1 are not practicable since the used padding is probabilistic. RSA PKCS#1 v2.1 cannot be compromised by applying DPA/DEMA since the used padding is probabilistic (different random numbers are used for each new signature of the message).

NOTE 6 There are two operations in DSA (resp. ECDSA) that involve the private key or an ephemeral (secret) key:

— The modular exponentiation (scalar multiplication) of a secret value with a known parameter. This operation is vulnerable to simple side-channel analysis and to horizontal differential ones.

— The modular multiplication of a known value and the private key. If the multiplication is implemented in such a way that the multiplier is the private key and the multiplication is carried out with a variant of the binary algorithm, then this implementation is, in principle, vulnerable to side-channel analysis.

NOTE 7 SHA can be used for password hashing, e.g. in a password-based key derivation function. In this case, a non-protected SHA against SPA/SEMA (including side-channel collision attacks) or DPA/DEMA can lead to the password.

NOTE 8 The definitions SPA/SEMA, DPA/DEMA and TA in this annex are more general than those in Clause 6. In particular, the DPA/DEMA in this annex includes the advanced EMAs and PAs in Figure 1.

Table C.1 *(continued)*

Security functions		Non-invasive attack methods		
		SPA/SEMA	DPA/DEMA	TA
	DSA	A[59]	A[59]	A[59]
	ECDSA	A[59]	A[59]	A[59]
Hashing mechanisms	SHA	A[16]	A[65]	NKR
RNG and RBG	Deterministic	A[16]	NKR	NKR
	Non-deterministic	A[16]	NKR	NKR
Data authentication mechanisms	HMAC	A[16]	A[65]	NKR
Key generation	Symmetric-key Ciphers	A[15]	NKR	NKR
	RSA	A[59]	NKR	A[1]
	ECDSA	A[59]	NKR	A[1]
Key derivation from other keys		NKR	NKR	A[66]
Key derivation from passwords		NKR	NKR	A[66]
Key establishment	DLC	A[59]	NKR	NKR
	IFC	A[59]	NKR	NKR
Operator authentication mechanisms	PIN/Password	A[16]	A[65]	NKR
	Key	NKR	NKR	NKR
	Biometrics	NKR	NKR	NKR

Key

A Applicable

NKR No known reference

NOTE 1 Applicable means that the security functions are susceptible to these types of attacks in public literature.

NOTE 2 No known reference means that the security functions are not known to be susceptible to these types of attacks in public literature, which does not mean that the attacks are not applicable.

NOTE 3 An HMAC implementation can be compromised by applying DPA/DEMA, however block-cipher based MAC is covered through AES and/or triple DES.

NOTE 4 RSA PKCS#1 v1.5 can be compromised by applying DPA/DEMA since the used padding is deterministic.

NOTE 5 Timing attacks on RSA PKCS#1 v2.1 are not practicable since the used padding is probabilistic. RSA PKCS#1 v2.1 cannot be compromised by applying DPA/DEMA since the used padding is probabilistic (different random numbers are used for each new signature of the message).

NOTE 6 There are two operations in DSA (resp. ECDSA) that involve the private key or an ephemeral (secret) key:

— The modular exponentiation (scalar multiplication) of a secret value with a known parameter. This operation is vulnerable to simple side-channel analysis and to horizontal differential ones.

— The modular multiplication of a known value and the private key. If the multiplication is implemented in such a way that the multiplier is the private key and the multiplication is carried out with a variant of the binary algorithm, then this implementation is, in principle, vulnerable to side-channel analysis.

NOTE 7 SHA can be used for password hashing, e.g. in a password-based key derivation function. In this case, a non-protected SHA against SPA/SEMA (including side-channel collision attacks) or DPA/DEMA can lead to the password.

NOTE 8 The definitions SPA/SEMA, DPA/DEMA and TA in this annex are more general than those in Clause 6. In particular, the DPA/DEMA in this annex includes the advanced EMAs and PAs in Figure 1.

Annex D
(informative)

Emerging attacks

D.1 Overview

This annex lists non-invasive attacks and side-channels against which pass/fail test metrics are not defined currently.

D.2 Template attack

The template attack makes use of the fact that the power consumption or electromagnetic emanation of the IUT depends on the processed data. This behaviour is characterized by the so-called template. The processed data, including CSP if any, can be identified by matching with the template (see References [35-37]).

Once the template is built, this is the most powerful attack in theory. However, in order to build the template, sometimes open samples are needed which allows testers to input various values. Also, it is known that there is some variation in the resultant templates depending on the instance of IUT. Therefore, the applicability and pass/fail test metrics are not defined in this document.

D.3 Side-channel collision attack

The side-channel collision attack also makes use of the fact that the power consumption or electromagnetic emanation of the IUT depends on the processed data. Cryptographic security functions include some steps to produce intermediate values from input value and cryptographic key. If the intermediate values become the same in value against different input values, the resultant power consumption or electromagnetic emanation would be quite similar. This kind of "collision" can be exploited in order to reduce the key space (see References [38-42]).

D.4 Sophisticated attacks on asymmetric cryptography

D.4.1 Doubling attack

In some cases, a testing laboratory can send specific input vectors to the IUT that allows the key to be retrieved with SPA/SEMA using a small number of waveforms. That is the principle of "doubling attacks".

The doubling attack relies on the fact that similar intermediate values are manipulated when working with a point P and its double (denoted $[2]P$), as shown in Reference [54]. If the testing laboratory can identify point doubling operations with identical data in two side-channel measurements, it can deduce key bit values. In ideal conditions, this attack only needs two side-channel measurements. If the testing laboratory retrieves the key, the test result has failed.

NOTE ECDSA is not vulnerable to doubling attack since the base point is fixed.

D.4.2 Markov SPA/SEMA

Some SPA/SEMA countermeasures lead to sequence of operations that can bring enough information to retrieve the key. "Markov SPA/SEMA" can then allow to clear the dependency between the sequence and the key.

If the cross-correlation leads to non-regular operation sequences, and if the link between this latter and the key is not clear at first sight, the testing laboratory can apply specific attacks to finally retrieve the

key. The testing laboratory can for example face a flawed randomized modular exponentiation (respectively scalar multiplication), and then it can use known weaknesses to finally retrieve the key by considering the exponentiation (or scalar multiplication) algorithm as a Markov process.[51] The attack proposed in Reference [51] on ECC works in four steps:

— Precomputation phase: find the Markov model. The testing laboratory shall calculate the conditional probabilities for sequences of bits and sequences of executed operations.

— Data collection phase: the testing laboratory shall deduce the sequence of operations square and multiply (double and add) operations.

— Data analysis phase: the testing laboratory shall split the sequence into a number of sub-sequences.

— Key testing phase: the testing laboratory shall check all possible keys by the known ciphertext.

The testing laboratory can test the IUT against Markov SPA/SEMA (especially when the relationships between the bits of the key and the operations are difficult to find) following the framework described in Reference [51], or any other relevant method. If it retrieves the key, the test result is fail.

D.4.3 Address-Bit DPA/DEMA

The address-Bit DPA/DEMA[14] exploits the fact that internal addresses of registers or memory locations are another type of data processed by the CPU. Hence, side-channel measurements of two intervals of an algorithm where the same instruction accesses different addresses will be less correlated than if that instruction was accessing the same address. The attack is easier if a small number of registers or memory locations are accessed during the algorithm depending on bits of the secret key.

The attacker computes the average side-channel measurements for two known keys: `0xfff…f`, and `0x80…01`. If the difference of the two averages (c_0 and c_1) shows spikes, the key bits leak and then the test result is fail (see Figure D.1).

NOTE DSA and ECDSA are not vulnerable to the address-Bit DPA/DEMA since they use an ephemeral exponent and scalar respectively.

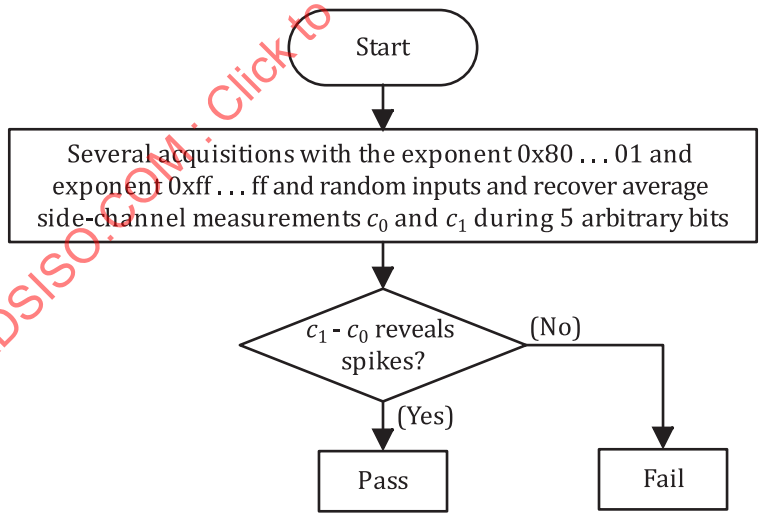

Figure D.1 — Address-bit DPA / DEMA

D.5 Refined SPA/SEMA

Concerning ECC, if projective coordinates are used in an IUT, they can be used to randomize the intermediate data. Standard SPA/SEMA as described in the previous clauses cannot be used directly. However, under the assumption that the secret key is not randomized, exploiting the properties of so-called special points can lead to an attack.[55] A special point *P0* (not equal to the infinity) is a point having the property that

one of the affine or projective coordinates is 0. Hence, randomization of projective coordinates does not affect this property, and the special property of this point can then be picked up from several side-channel measurements by averaging over the measurements.

In Reference [56], an extension to the attack in Reference [55] is described. This extended attack is based on the observation that, even if a point does not have a zero coordinate, some of the intermediate values that occur during point addition (or point doubling) can become zero.

The testing laboratory can test the IUT against Refined SPA/SEMA following the framework described in References [55] and/or [56] or any other relevant method. If it retrieves the CSP, the test result is fail.

ECDSA is the only cryptographic protocol based on elliptic curves in this document. Therefore, the refined SPA/SEMA is not practicable since the base point is fixed.

D.6 Use of new emerging side-channels

The number of possible side-channels can increase in the future (e.g. photonic emissions[49] acoustic emanations). At the time of writing this document, there is not enough knowledge on these new side-channels to define a pass/fail metric.

D.7 Timing variation due to power consumption

As a device consumes various amounts of power, the internal voltage of the device will vary slightly. This is the same signal picked up when using a resistive shunt for a DPA/SPA measurement.

The same variations in voltages cause variations in the timing of internal signals, which can be picked up using a time-to-digital converter (TDC) to output a signal similar to a power trace.[67] This power trace can be treated as it was a SPA/DPA measurement for the purpose of pass/fail testing, including applying the SNR tests in 7.3.6.

Annex E
(informative)

Quality criteria for measurement setups

E.1 Electronic noise

E.1.1 Noise of the power supply

To decrease the noise induced by the power supply, a highly stable power supply should be used. The IUT should never be powered directly by a PC (e.g. via the USB port).

E.1.2 Noise of the clock generator

As a power supply, a clock shall be stable. On the one hand, the implementers cannot be asked to provide a time-stable internal clock since it can be a way to desynchronize the executions and then be a countermeasure against side-channel. On the other hand, an amplitude-stable internal clock is mandatory since an unstable one can bring coupling effects. In general, a sinusoidal clock signal should be preferred to a rectangular one. The quality of the clock generator (e.g. crystal oscillator) should be assessed.

E.1.3 Conducted and radiated emissions

Except for the IUT, the other components that are present on the IUT board can bring conducted emissions, and then bring noise in side-channel measurements. Ideally, the measurement setup should be built with two PCBs: a measurement board and an interface board. The measurement board only contains the attacked device and a power measurement circuit. The interface board takes care of the communication with the PC. These two boards are ideally isolated by opto- or magnetic couplers in the communication lines. The sources of conducted emissions are therefore minimized.

The impact of radiated emissions on a measurement setup can be reduced by shielding. The PCB that features the attacked device can be put in a Faraday cage. The communication and the measurement lines that are connected to the attacked device should be shielded or decoupled accordingly.

E.1.4 Quantisation noise

This noise is the consequence of the analogue-to-digital conversion that is performed by the oscilloscope. A resolution of 8 bits is mandatory. In this case, the effect of quantisation noise is typically much smaller than the effect of the other kinds of noise.

E.2 Switching noise

Switching noise is referred to as the variations of the side-channel measurements which are caused by cells that are not relevant for the attack. For example, in a side-channel analysis of a hardware implementation of a 128-bit AES, the side-channel emitted by one part of the IUT (e.g. one SBox) is focused. However, if many parts of the IUT emit side-channels at the same time, the attacker shall [E.01] discriminate all the parts to retrieve the part of the IUT that has been chosen to attack. The side-channel emission of all other parts of the device is “noise” from the attacker’s point of view.

Since the power consumption of the IUT is a global side-channel and the electromagnetic (EM) emanation of the IUT is a local side-channel, EM should be preferred for side-channel resistance validation.

The amount of switching noise depends also on the frequency of the clock signal that is used to operate the cryptographic device. If, for example, a high clock frequency is used for the attacked device, it is possible that the power consumption signals of consecutive clock cycles interfere with each other.