Multivariate multitarget high order side-channel attacks

Nicolas Bruneau

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

TELECOM ParisTech

2017-ENST-0025

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité « Electronique et Communications »

présentée et soutenue publiquement par

Nicolas BRUNEAU

le 18/05/2017

Attaques par Canaux Auxiliaires Multivariées, Multi-cibles et d'ordre élévé

Directeur de thèse : **Sylvain Guilley**

Jury

Mme Svetla NIKOVA, Professeur, Katholieke Universiteit Leuven	Rapporteur
Mme Elisabeth OSWALD, Professeur, University of Bristol	Rapporteur
M. Michel AGOYAN, Ingénieur de Recherche, STMicroelectronics	Examinateur
M. Emmanuel PROUFF, Ingénieur de Recherche, Safran	Examinateur
M. Olivier RIOUL, Professeur, Télécom ParisTech	Examinateur
M. François-Xavier STANDAERT, Professeur, Université Catholique de Louvain	Examinateur
M. Yannick TEGLIA, Ingénieur de Recherche, Gemalto	Examinateur

TELECOM ParisTech
école de l'Institut Mines-Télécom - membre de ParisTech
46 rue Barrault 75013 Paris - (+33) 1 45 81 77 77 - www.telecom-paristech.fr

THÈSE

Remerciements

C'est par ces lignes de remerciements que je termine la rédaction de mon manuscrit de thèse. Une thèse loin d'être un exercice solitaire doit sa réussite à la contribution et à la participation de nombreuses personnes, celle ci ne fait pas exception. Je tiens donc à remercier ici toute les personnes qui ont permis, aidé et finalement fait réussir cette thèse.

Pour commencer je souhaiterai remercier, M. Michel Agoyan, Mme Svetla Nikova, Mme Elisabeth Oswald, M. Emmanuel Prouff, M. Olivier Rioul, M. François-Xavier Standaert et M. Yannick Teglia de m'avoir fait l'honneur d'accepter d'être les membres de mon jury de thèse. Je tiens également à remercier particulièrement Mmes Svetla Nikova et Elisabeth Oswald d'avoir accepté de relire mon manuscrit de thèse. Par leur relecture attentive et leurs commentaires elles ont grandement contribué à l'améliorer.

Je souhaite spécialement remercier Olivier Rioul avec qui j'ai eu la chance de travailler durant ma thèse. Ces idées, ces remarques l'on fait énormément progresser. Qu'il soit ici remercié pour son implication dans mes travaux.

Je souhaite aussi particulièrement et sincèrement remercier Michel Agoyan il fut pendant trois ans, pour moi, plus qu'un collègue. En effet, Michel a été un excellent professeur au-près du quel j'ai beaucoup appris, bien sûr d'un point de vue scientifique et technique, mais pas seulement.

Cette thèse a été rendue possible par le travail et l'implication constante de Yannick Teglia. Avant même cette thèse il m'avait donner l'opportunité d'être stagiaire au sein l'équipe AST de STMicroelectronics, il m'a ensuite donné la chance de réaliser cette thèse dont il a été l'encadrant coté industriel. J'ai beaucoup appris grâce à lui, dans de nombreux domaines techniques, mais aussi sur la vie dans l'entreprise. Yannick a su créer au sein de STMicroelectronics un environnement convivial et vraiment stimulant pour l'ensemble des doctorants amenés à y travailler. Grâce à lui ces années passées à STMicroelectronics ont été enrichissantes (mention spéciale à nos sessions au laboratoire de Gardanne).

Je tiens évidemment à particulièrement remercier Sylvain Guilley d'avoir dirigé cette thèse. Il en a été tout le long un remarquable encadrant. Il a bien sûr été d'un point vue scientifique une source inépuisable d'idées et de conseils; me guidant toujours dans une bonne direction sans jamais m'en imposer aucune. Il a toujours su me communiquer son enthousiasme en nos travaux et dans la science en générale. Cet enthousiasme partagé a été durant ma thèse une

source de motivation. Je tiens pour finir à le remercier pour sa confiance qu'il n'a eu de cesse de me donner.

Cette thèse a été réalisée en collaboration entre STMicroelectronics et Télécom ParisTech. Je souhaiterai donc adresser de chaleureux remerciements à l'ensemble de mes collègues qui m'ont accompagné pendant ces années de thèse. Je souhaite particulièrement adresser un amical remerciement à l'ensemble des personnels administratifs de Télécom ParisTech et de STMicroelectronics qui ont facilité l'ensemble des démarches que j'ai eu à effectuer.

Pendant ces années j'ai été amené à partager le quotidien de l'équipe AST de STMicroelectronics et du groupe SEN de Télécom ParisTech. Je souhaiterai donc remercier l'ensemble de leurs membres. En particulier je remercie Jean-Luc Danger et Bernard Kasser de m'avoir permis de faire partie de ces équipes.

Je souhaiterai spécialement remercier mes collègues Mathieu Carbone et Patrick Haddad qui m'ont précédé en tant que doctorant à STMicroelectronics. J'ai beaucoup appris de leurs conseils et de leur expérience qu'ils ont accepté de me transmettre.

Je voudrais également remercier Annelie Heuser avec qui j'ai eu la chance de travailler durant cette thèse. J'ai grâce à elle appris beaucoup sur la recherche, sur ce qu'est être doctorant et chercheur.

Je dois également un immense remerciement à Zakaria Najm. Il fut mon collègue durant cette thèse mais dès avant durant mon stage. J'ai beaucoup appris grâce à lui, il a toujours partagé avec moi ces connaissances dans tellement de domaines qu'il seraient trop longs à citer.

Je souhaite également remercier mes collègues, mes amis: Maxime Madau et Lydie Terras. Avec Maxime et Lydie j'ai partagé de formidables années. Je souhaite remercier Maxime pour toute l'aide qu'il m'a donné et pour m'avoir fait profiter de son expertise. Je souhaite surtout le remercier pour toutes les discussions passionnantes, parfois futiles mais jamais inutiles que nous avons eu. Je remercie Lydie, bien sûr pour toute l'aide qu'elle m'a donné dans beaucoup de domaines, mais avant tout je la remercie pour avoir toujours su me communiquer sa bonne humeur et son énergie.

J'ai, durant ma thèse toujours pu compter sur le soutien de mes amis Chloé Cahuet et Yuhei Oshima. Ils ont toujours été là pour me soutenir, m'encourager et m'accompagner. Je les remercie sincèrement pour leur amitié et simplement pour avoir toujours été là quand j'en ai eu besoin.

Enfin je tiens à exprimer toute ma gratitude à ma famille d'avoir été et d'être toujours ce soutien indéfectible et inconditionnel qui a été le pilier de cette thèse.

Abstract

Side Channel Attacks are a classical threat against cryptographic algorithms in embedded systems. They aim at exploiting the physical leakages unintentionally emitted by the devices during the execution of their embedded programs to recover sensitive data. They exploit the dependencies between the leakages and the values manipulated by the algorithms. As such attacks represent a real threat against embedded systems different countermeasures have been developed. They ensure the security of the devices against Side Channel Analysis. Data masking are one of the most classical countermeasures. In a masking scheme any sensitive variable is randomly split into several shares. The security of these algorithms comes from the fact that the first high order moment which depends on the secret data is increased. This order is one of the fundamental parameters of protected implementations. Nevertheless these countermeasures are also the target of particular kinds of Side Channel Analysis. In these attacks, called High Order Attacks an attacker must combine different variables (the shares) in order to recover the sensitive variable.

In this manuscript we investigate the relevant parameters which allow to build security evaluation of the cryptographic algorithms protected implementations. Specifically we focus on masking schemes, but we also investigate the case of shuffling. We investigate their security in presence of multiple leakages. Indeed there often are in the leakage measurements several variables which can be exploited to mount Side Channel Attacks. Each variable may leak through many leakage samples. Any one of these leakages represents a way to improve the results of the attacks.

In the first part of this manuscript we show that the multiple leakages of a unique variable can be exploited together to build efficient attacks. In particular we show the optimal way to exploit these leakages. This optimal treatment coincides with a dimensionality reduction. We show that, in some cases, this dimensionality reduction comes with no loss on the overall exploitable information. We additionally show that this dimensionality reduction is asymptotically equivalent to a well known dimensionality reduction method. Based on this result we investigate further how such dimensionality reduction methods can be applied in the case of protected implementations. We show that the impact of such methods increases with the security "level" of the implementation. This observation gives us a first example of a case where the standard parameter to evaluate the security of the implementation may not be sufficient. Additionally we present a new optimal

dimensionality reduction method which is available without a priori knowledge on the leakage function.

In the second part of this manuscript we investigate how to exploit the leakages of multiple variables in order to improve the results of Side Channel Analysis. We start by improving the attacks against a particular kind of masking schemes, namely the ones with a precomputed table recomputation step. We give the optimal attack against such schemes which provides better results than the state-of-the-art attacks. This new attack takes into account the multiple leakages of the table recomputation. Some protections have been developed to protect the table recomputation steps. As a consequence we investigate the security provided by these protections. In this context we present results which show that the main parameter to evaluate the security of the masking schemes, namely the order, is not sufficient to estimate the global security of the implementation. Indeed we exhibit a new attack which gives better results than the classical minimal order attack of the state-of-the-art. We extend this result in different scenarios in terms of leakage functions and type of implementations. We theoretically investigate the best possible attacks in presence of masking and shuffling. This generalizes the previous case study. We show that in context of shuffling the optimal attack is not computable. As a consequence we present a truncated version of this attack with a better effectiveness. This new attack has efficiency close to the optimal attacks but with a complexity which makes it computable. Additionally it allows a better understanding of the behaviors of attacks with multiple leakages at multiple orders.

Résumé de la thèse en français

Les analyses par canaux auxiliaires représentent une vulnérabilité classique des systèmes embarqués. Elles exploitent les fuites physiques consécutives aux interactions du composant et de son environnement lors de l'exécution de ses programmes embarqués. Un attaquant va exploiter ces fuites qui peuvent par exemple être la consommation du composant ou bien ses émissions électromagnétiques pour retrouver des données secrètes. En effet, il existe des liens entre ces fuites et les variables manipulées par les algorithmes. Ces attaques représentent une réelle menace contre les systèmes embarqués ; c'est pourquoi différentes contre-mesures ont été développées. Elles visent à garantir la sécurité des systèmes contre les analyses par canaux auxiliaires. Cette thèse va s'intéresser à la sécurité fournie par ces contre-mesures. Parmi l'ensemble des contre-mesures les schémas de masquage sont particulièrement utilisés. Dans de tels schémas toutes les variables sensibles sont aléatoirement découpées en plusieurs « parties ». La sécurité de ces algorithmes provient du fait que l'ordre du premier moment dépendant de la donnée secrète est augmenté. Cet ordre est l'un des paramètres fondamentaux des implémentations protégées. Néanmoins ces contre-mesures sont elles mêmes la cible de types particuliers d'analyses par canaux auxiliaires. Ces attaques appelées attaques d'ordres élevés doivent combiner différentes variables pour retrouver la variable sensible. Dans ce manuscrit nous étudions les paramètres importants qui permettent la construction d'évaluations sécuritaires des implémentations protégées d'algorithmes cryptographiques. Nous nous intéressons en particulier aux schémas de masquage mais aussi aux protections basées sur du « shuffling ». Nous étudions leur sécurité dans le contexte où de multiples fuites sont présentes. Il arrive régulièrement que plusieurs fuites de plusieurs variables puissent être exploitées pour monter des analyses par canaux auxiliaires. En effet, chacune de ces variables peut fuiter à de multiples reprises. Toutes ces fuites représentent des chemins possibles d'améliorations des attaques.

Dans la première partie de ce manuscrit nous montrons que les multiples fuites d'une unique variable peuvent être exploitées pour bâtir des attaques efficaces. En particulier nous présentons la méthode optimale pour exploiter l'ensemble de ces fuites. Cette méthode optimale correspond à une réduction de dimensionnalité. Sous certaines contraintes, nous montrons de plus que cette réduction de dimensionnalité n'entraîne aucune perte sur l'information exploitable. Nous montrons également que cette méthode est asymptotiquement équivalente à une méthode connue de réduction de dimensionnalité. En nous appuyant sur ces résultats nous étudions également

comment de telles méthodes de réduction de dimensionnalité peuvent être appliquées dans le contexte d'implémentations protégées. Nous montrons dans ce manuscrit que de telles méthodes voient leur efficacité augmentée avec le niveau de sécurité de l'implémentation. Cette observation nous donne un premier exemple dans lequel le paramètre standard pour évaluer la sécurité des protections peut ne pas être suffisant. De plus nous proposons une nouvelle méthode de réduction de dimensionnalité qui est calculable même sans connaissance a priori de la fonction de fuite.

Dans la seconde partie de ce manuscrit nous investiguons comment exploiter les fuites de multiples variables pour améliorer les résultats d'analyses par canaux auxiliaires. Dans un premier chapitre nous améliorons l'état de l'art des attaques contre un type particulier de schémas de masquage, à savoir les schémas de masquage avec recalcul de table. En effet dans ce contexte nous présentons l'attaque optimale dont les résultats sont meilleurs que ceux des attaques de l'état de l'art. Comme les schémas de masquage avec recalcul de table peuvent être protégés contre ce type d'attaque nous étudions dans un second temps la sécurité de ces protections. Dans ce scénario nous présentons des résultats qui montrent que le principal paramètre pour évaluer la sécurité des schémas de masquage, c'est-à-dire l'ordre n'est pas suffisant. En effet nous présentons une nouvelle attaque qui donne de meilleurs résultats que l'attaque d'ordre minimal. Nous étendons ces résultats dans différents scénarios : avec différentes fonctions de fuites et différentes implémentations. Pour finir nous étudions de façon théorique la meilleur attaque possible en présence de masquage et de « shuffling » ce qui généralise le précédent cas d'étude. Dans ce cas nous montrons que l'attaque optimale n'est pas calculable. Pour y remédier, nous présentons une version tronquée de l'attaque optimale avec une meilleure efficacité calculatoire. Cette nouvelle attaque conserve des résultats proches de l'attaque optimale mais avec une bien meilleure complexité, ce qui la rend calculable. De plus sa formule permet une meilleure compréhension des différents comportements des attaques en présence de multiples fuites à différents ordres.

Ce manuscrit comporte six chapitres.

Chapitre 1 Introduction.

La cryptographie est devenue au cours de ces dernières années un élément fondamental des communications notamment numériques. En effet la cryptographie va assurer la sécurité des

communications en fournissant un certain nombres d'outils appelés primitives cryptographiques permettant d'assurer différentes propriétés:

- La confidentialité des données, cette propriété assure que l'information transmise n'est compréhensible que par les parties autorisées.
- L'intégrité des données, cette propriété garantie que les données n'ont aps été altérées par une tierce partie.
- L'authentification, cette propriété assure que l'échange se fait bien entre les parties voulues.
- La non répudiation, cette propriété assure que les parties ne pourrons nier leurs actions.

Ces propriété vont être assurées par différents types d'algorithmes cryptographiques qui peuvent se décomposer en au moins deux grandes familles.

Les algorithmes cryptographiques symétriques. Ces algorithmes sont basés sur le fait que les différentes parties impliquées dans la communication possèdent un secret commun appelé clef secrète. Ce type d'algorithmes va pouvoir assurer notamment la confidentialité des données. Dans ce manuscrit nous nous intéressons tout particulièrement à un type particulier d'algorithmes cryptographiques symétriques: les chiffrements par bloc.

Les algorithmes cryptographiques asymétriques. Dans le cas des algorithmes cryptographiques asymétriques une seule partie possède la clef secrète alors que les autres parties ont accès à une clef publique. C'est de cette asymétrie dans la connaissance du secret que ce type d'algorithmes tire son nom. En plus de facilité l'échange de clef ces algorithmes permettent la confidentialité des données mais également l'intégrité et l'authentification.

Ces algorithmes sont supposés sûrs dans un modèle d'attaquant en boîte noire. Néanmoins dans la pratique ces algorithmes sont exécutés sur des composants physiques qui peuvent être par exemple des cartes à puce des micro-contrôleurs. Ce type d'implémentation est vulnérable aux attaques dites physiques. Parmi l'ensemble des attaques physiques nous nous intéressons dans ce manuscrit aux attaques par canaux auxiliaires. Ces attaques exploitent les fuite physiques induites par les interactions entre le composant et son environnement.

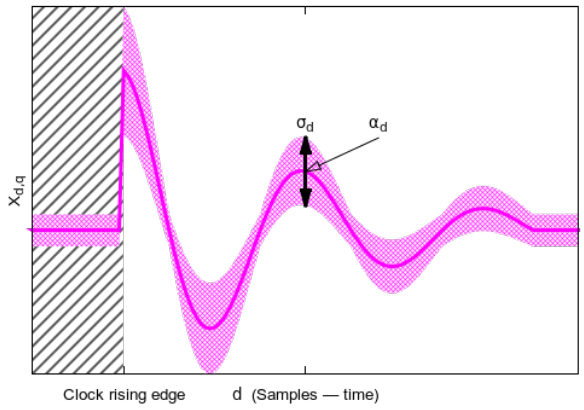


Figure 1: Exemple de trace de consommation

Les attaques par canaux auxiliaires. Les attaques par canaux auxiliaires sont un type particulier d'attaques physiques, ces attaques n'interagissent pas directement avec le composant de test, par conséquent ces attaques sont qualifiées d'attaques physiques non-invasives et passives. Ces attaques exploitent en général des enregistrements numériques, appelé traces ou encore mesures, de différentes grandeurs physiques qui peuvent être par exemple les émissions électromagnétiques ou bien la consommation du composant. Comme présenté en Figure 1. Chacune de ces mesures peut être porteuse d'information en différents points qu'ils convient d'exploiter afin de construires les attaques le plus efficacement possible.

Afin d'extraire des fuites physiques d'information les valeurs secrètes ce type d'attaque exploitent souvent les comparaisons entre les mesures et un modèle théorique en utilisant un "distingueur".

Les caractéristiques de ces "distingueurs" peuvent varier en fonctions des différents prérequis nécessaires à leurs calculs ou encore des fuites à exploiter.

Contre-mesures contre les attaques par canaux auxiliaires. Afin de protéger les implémentations cryptographiques contre les attaques par canaux auxiliaires différentes contre-mesures ont été développées. Les contre-mesures les plus utilisées sont les schémas de masquage. En effet, leur sécurité peut être formellement prouvée. Dans ces schémas chacune des variables sensibles Z est "découpée" en Ω parties dans le cas d'un schéma d'ordre $\Omega - 1$ en utilisant Ω nombres aléatoires appelés les masques. Les implémentations cryptographiques peuvent être également protégées en utilisant une contre-mesure appelée "shuffling". Cette contre-mesure consiste à

exécuter dans un ordre aléatoire les différentes opérations indépendantes d'un algorithme. Ces contre-mesures sont elles mêmes la cible d'un type particulier d'attaques par canaux auxiliaires appelé attaques d'ordre élevé. Ces attaques reposent sur la combinaison de différents points de fuites dépendant de différentes variables.

Il convient donc dans l'optique de construire des attaques efficaces voir optimales de prendre en compte différents points de fuite. Dans ce manuscrit nous montrons comment exploiter ces multiples fuites que ce soit dans le cas d'attaque contre des implémentations non protégées ou dans le cas d'implémentations protégées.

Chapitre 2 Réduction de dimensionnalité optimal.

Réduire la dimensionnalité des mesures est un problème important dans les analyses par canaux auxiliaires.

En effet les fuites exploitables sont souvent multi-dimensionnelles il y a donc en général un intérêt à réduire leur dimension afin de réduire la masse de données à traiter lors des attaques. De manière générale cela permet des capturer les fuites multi-dimensionnelles en un seul échantillon compressé, et donc de réduire la complexité calculatoire. Le revers de la médaille est que de telles méthodes peuvent réduire l'efficacité des attaques en termes de probabilité de succès. En effet la question se pose de savoir quel est l'impact de la réduction de dimensionnalité en termes d'information exploitable dans les attaques et par conséquent en terme de probabilité de succès.

Dans ce chapitre nous analysons mathématiquement la réduction de dimensionnalité. Nous montrons que l'attaque optimale reste optimale après une première passe de prétraitement qui prend la forme d'une projection linéaire des échantillons. C'est à dire que la réduction de dimensionnalité peut se faire sans perte d'information. Nous étudions l'état de l'art des méthodes de réduction de dimensionnalité et trouvons que asymptotiquement, la stratégie optimale coïncide avec l'analyse discriminante linéaire. Nous prouvons également qu'en général l'analyse en composantes principales ne coïncide pas avec la réduction de dimensionnalité optimale.

Chapitre 3 Réduction de dimensionnalité dans le cas du masquage.

Les attaques multivariées permettent « d'attaquer » les schémas de masquage d'ordre élevé en combinant plusieurs points de fuites. Mais dès lors la question se pose de savoir quelle est la

meilleur façon d'extraire l'information répartie dans tous les Ω-uplets de points. Dans ce chapitre nous proposons l'outil de pré-traitement qui répond à cette question. Nous montrons d'abord qu'il est équivalent de résoudre le problème de la maximisation du coefficient des attaques par corrélation notées CPA d'ordre élevé et le problème de maximisation de la covariance. Nous pouvons alors, dans un premier temps, appliquer cette équivalence au problème de réduction de dimensionnalité des traces par combinaisons linéaires, les pondérations de cette combinaison sont notées par le vecteur α. Cela nous permet ensuite de lier ce problème avec l'Analyse en Composante Principale notée PCA. Dans un second temps nous présentons la solution optimale au problème de maximisation de la covariance comme présentée en Figure 2.

Nous montrons ensuite théoriquement que ces deux méthodes donnent des résultats équivalents lorsque les traces attaquées sont « modulées ». Nous comparons théoriquement et empiriquement ces méthodes.

Pour finir nous appliquons ces résultats sur les traces du DPA Contest V4 afin d'évaluer combien les techniques proposées améliorent les attaques du second ordre.

Comme nous pouvons le voir sur la Figure 3 les attaques construites sur une étape de réduction de dimensionnalité ont de meilleurs résultats. De plus nous remarquons que les deux méthodes de réduction de dimensionnalité c'est à dire l'analyse en composantes principales et la méthode de covariance donnent des résultats identiques. Ceci prouve que les traces du DPA Contest V4 sont proches de traces modulées.

Chapitre 4 Attaque Optimale contre le recalcul de table.

Les parties non linéaires des schémas de masquage sont souvent construites en utilisant des méthodes de recalcul de table. De tels algorithmes sont caractérisés par leurs nombreuses fuites pouvant être exploitées afin de réaliser des attaques par canaux auxiliaires. Par conséquent différentes attaques ont été proposées dans l'état de l'art afin d'exploiter ces fuites. Dans ce chapitre nous étudions de façon théorique la méthode optimale, au sens de la méthode maximisant la probabilité de succès, exploitant ces multiples fuites.

Nous montrons en particulier que l'attaque optimale, notée OPT surpasse en terme de probabilité de succès les différentes attaques de l'état de l'art. En effet nous pouvons voir en Figure 4 que les attaques multi-variées exploitant le recalcul de table sont plus efficaces (ont une meilleure probabilité de succès) que les attaques exploitant uniquement les fuites hors de l'étape de recalcul, comme par exemple l'attaque en corrélation du second ordre notée 2O-CPA.

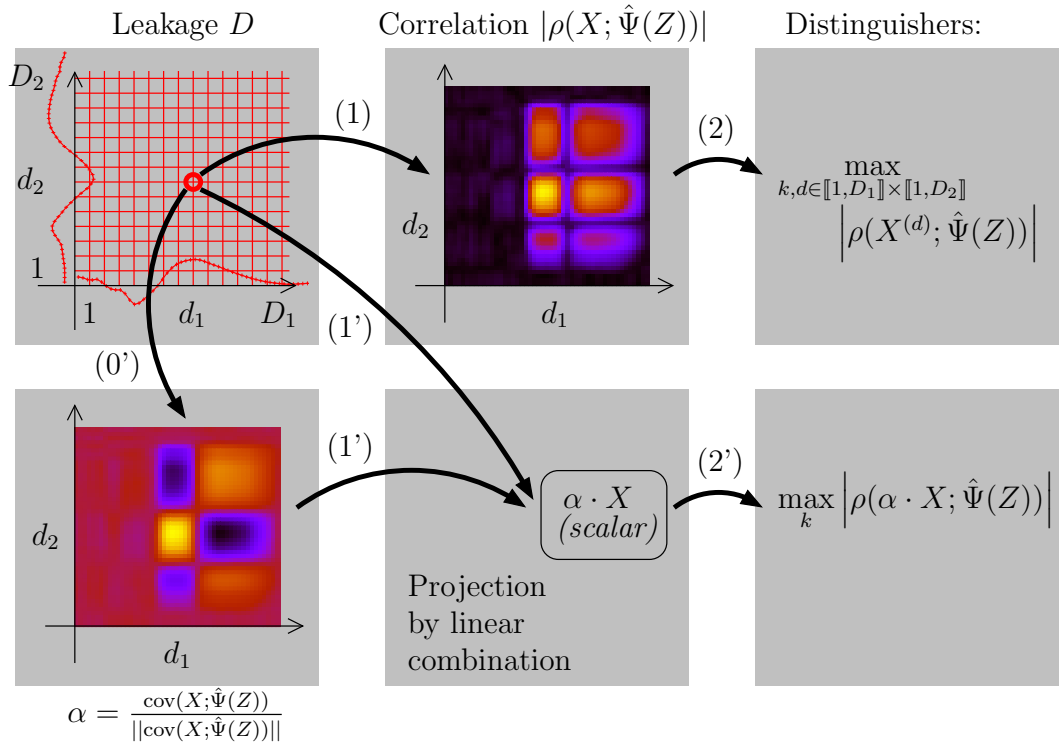


Figure 2: Presentation de la « méthode de covariance » . L' attaque 2O-CPA calcule la corrélation pour chaque paire (d_1, d_2) de la fuite (étape (1)), ensuite le maximum sur chacune des clefs et des instants temporels est recherché (étape (2)). Notre méthode obtient un « vecteur de covariance » sur un composant d'apprentissage (étape (0')), et ensuite projette la fuite X sur α (étape (1')), avant de chercher la meilleure clef lors de la maximisation du distingueur. Remarquons que le modèle $\widehat{\Psi}(Z)$ dépend implicitement de l'hypothèse de clef k.

Cela montre qu'il y a toujours un intérêt à augmenter le nombre de points de fuites exploités pour les attaques, dans la mesure où ceux ci sont exploités de façon optimale.

De plus l'utilisation du distingueur optimal contre des schémas de masquage avec recalcul de table permet également d'améliorer le résultat des attaques exploitant les fuites du recalcul de table en plus des fuites classiques de l'algorithme cryptographique. En effet nous montrons que les attaques de l'état de l'art exploitant les fuites du recalcul en deux étapes, une première pour retrouver le masque, la seconde pour retrouver la valeur secrète ne sont pas aussi efficaces, comme illustré en Figure 4. Ces attaques sont notées dans ce manuscrit $2\times\mathrm{CPA}^{mt}$. On en déduit donc que la meilleur approche est d'exploiter l'ensemble des fuites en une en appliquant l'attaque optimale.

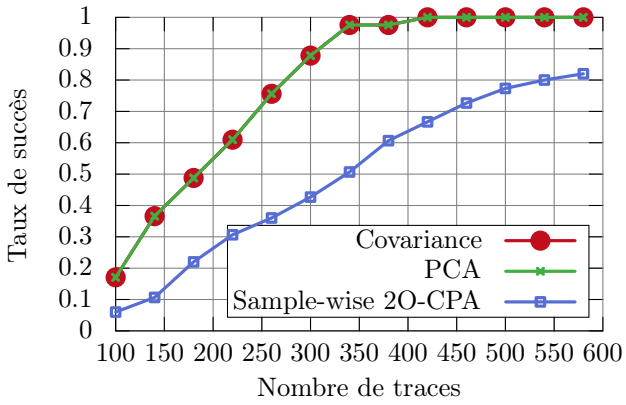
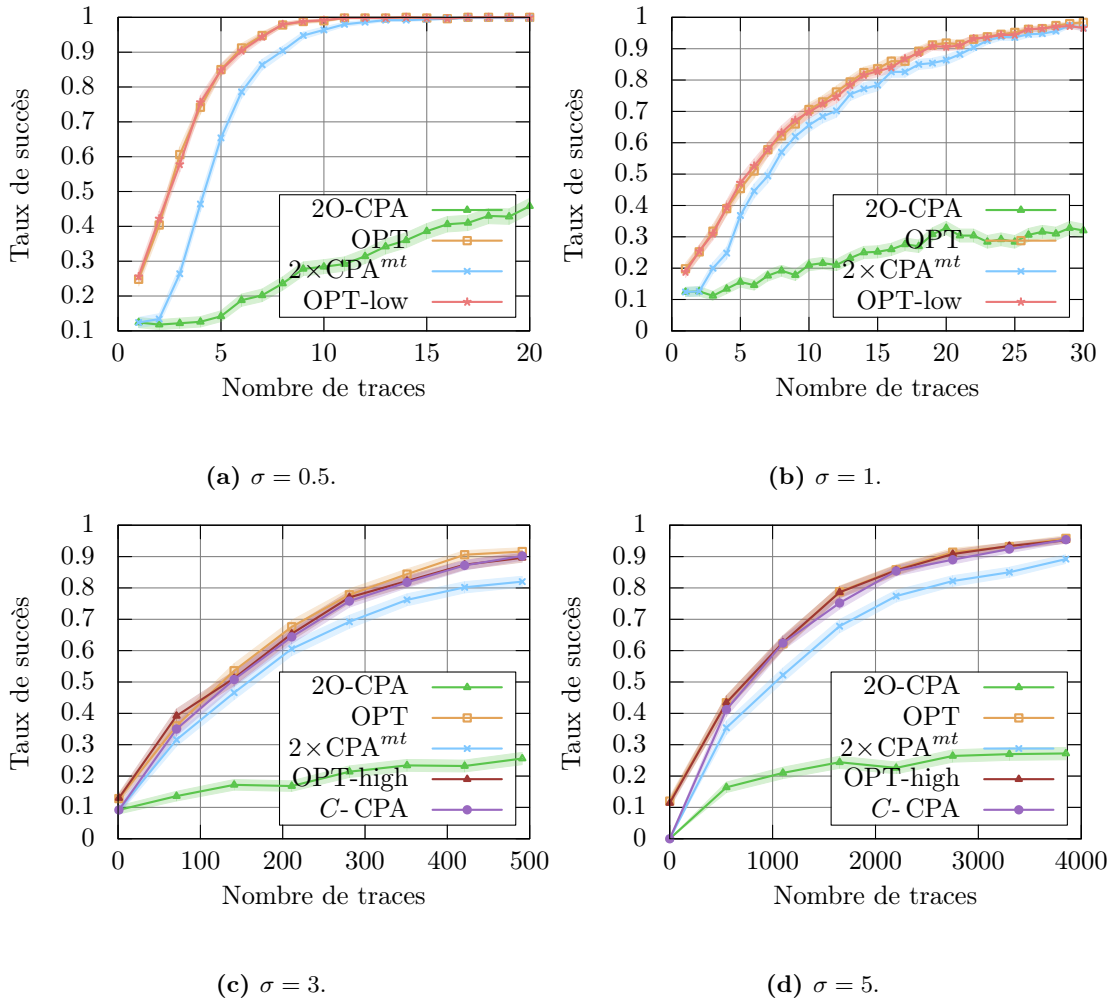


Figure 3: Comparaison entre une attaque CPA du second ordre classique et une attaque second ordre CPA avec prétraitement utilisant (S-boxes, S-Boxes)

Chapitre 5 Attaque multivariée d'ordre élevé contre le recalcul de table calculé dans un ordre aléatoire.

Les schémas de masquage basés sur le recalcul de table sont des contre-mesures classiques contre les analyses par canaux auxiliaires d'ordre élevé. Néanmoins ils sont connus pour être attaquables à l'ordre Ω dans le cas où le masquage utilise Ω parties. Dans ce chapitre nous montrons mathématiquement qu'une attaque d'ordre strictement plus grand que Ω peut être plus efficace qu'une attaque d'ordre Ω. Pour se faire nous étendons l'idée de Tunstall, Whitnall et Oswald de FSE 2013: en effet nous montrons des attaques utilisant les multiples fuites liées à un masque durant le recalcul de table. En particulier dans le cas d'une implémentation d'un recalcul de table effectué dans un ordre aléatoire, nous montrons qu'il existe une fenêtre d'opportunité en termes de variance de bruit, dans laquelle une nouvelle attaque multivariée du troisième ordre est plus efficace que la classique attaque bi-variée du second ordre.

L'attaque présentée dans ce cas d'étude utilise, à son avantage, les multiples fuites des différentes variables lors du recalcul de table. En effet nous montrons dans ce cas là qu'il est possible d'extraire du recalcul une variable dépendante du masque. Cette nouvelle variable, issue donc de la combinaison des différentes variables du recalcul, est elle même combinée à la variable secrète. C'est cette combinaison finale qui permet de retrouver la valeur de la variable secrète. Du fait des deux combinaisons successives cette nouvelle attaque est une attaque du troisième ordre. Cette attaque est notée MVA_{TR}. Nous pouvons vérifier de plus que cette attaque sera plus efficace que les attaques de l'état de l'art lorsque la variance du bruit est comprise dans un



(a) $\sigma = 0.5$.

(b) $\sigma = 1$.

(c) $\sigma = 3$.

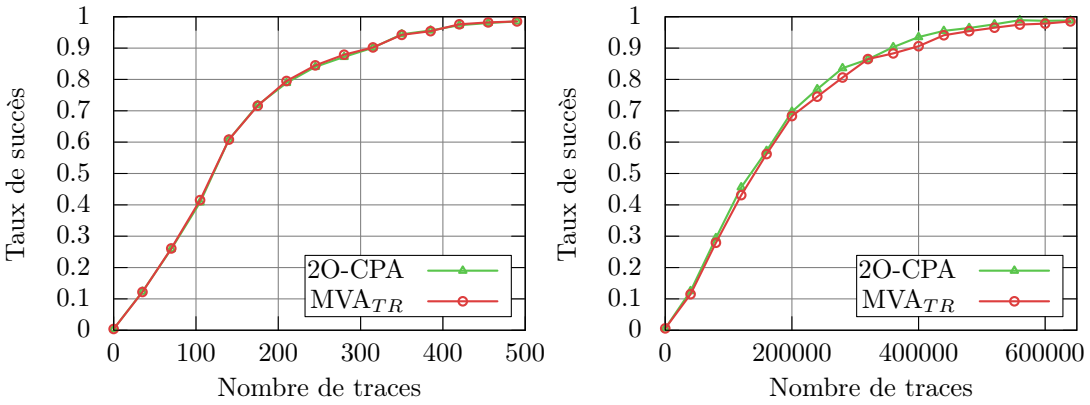
(d) $\sigma = 5$.

Figure 4: Taux de succès contre les tables masquées.

intervalle particulier que nous appelons intervalle de variance utile.

Afin de comparer les résultats de cette nouvelle attaque à ceux de l'état de l'art et de vérifier de façon empirique nos résultats théoriques, différentes simulations ont été effectuées. Pour chacune nous avons supposé que les variables fuitaient leur poids de Hamming, nous avons de plus supposé un bruit blanc gaussien. L'attaque de référence utilisée est l'attaque du second ordre par corrélation notée 2O-CPA. Nous pouvons dans un premier temps observer en Figure 5 qu'aux bornes de l'interval de variance utile la 2O-CPA et notre nouvelle attaque MVA$_{TR}$ coïncident, ce qui valide nos résultats théoriques. Dans un deuxième temps nous pouvons observer en Figure 6 qu'entre ces bornes l'attaque MVA$_{TR}$ montre de meilleurs résultats. En

effet, son taux de succès est plus élevé que la 2O-CPA.



(a) $\sigma = 0$. **(b)** $\sigma = 8$.

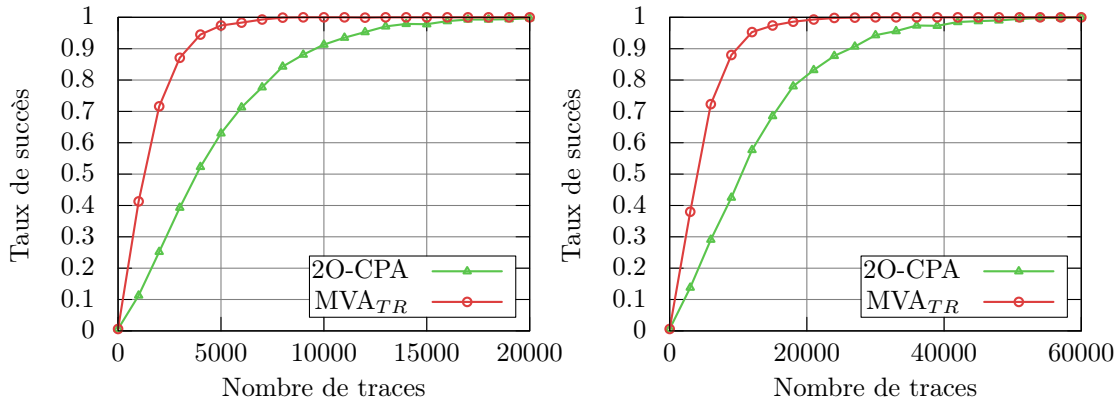
Figure 5: Comparaisons entre 2O-CPA et MVA$_{TR}$

De plus dans le cas du recalcul de table protégé à des ordres élevés présenté par Coron à EUROCRYPT 2014 nous montrons que cette fenêtre s'élargie linéairement avec l'ordre Ω. Dans ce cas nous redonnons une définition d'une attaque exploitant les fuites lors du recalcul de table. Cette nouvelle attaque notée MVA$^{\Omega}_{CS}$ va dépendre de l'ordre utilisé par le masquage de Coron.

Comme précédemment nous avons évalué ces résultats théoriques en utilisant des simulations. Nous supposons toujours que les variables fuitent en poids de Hamming avec un bruit blanc gaussien. On peut remarquer sur la Figure 7 que non seulement l'attaque MVA$^{\Omega}_{CS}$ surpasse l'attaque de référence qu'est l'attaque par corrélation d'ordre élevé notée ici ΩO-CPA, mais que cet avantage augmente avec l'ordre du schéma de masquage.

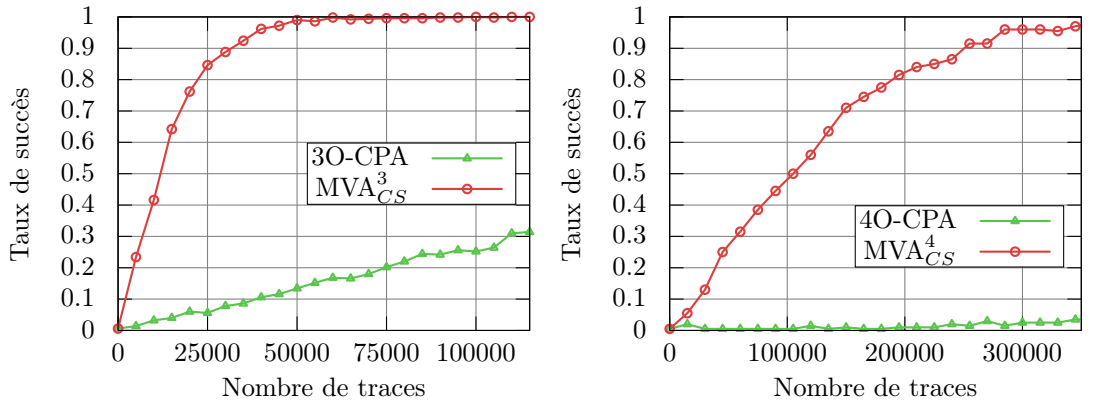
Dans ce chapitre nous étudions également le cas de différents modèles de fuite de degré un et montrons formellement que le modèle en poids de Hamming est le cas le moins favorable à l'attaquant. Nous montrons en effet que lorsque le modèle de fuite est une pondération des bits de la valeur sensible, le meilleur cas pour notre nouvelle attaque multivariée est lorsque tout les bits sont à zéro sauf un, inversement le pire cas est lorsque la pondération est la même pour tout les bits. Ainsi, l'ensemble des analyses effectuées précédemment en poids de Hamming sont une borne inférieure des résultats que nous pouvons obtenir avec l'attaque MVA$_{TR}$.

Finalement nous validons ces résultats sur une carte puce. La cible est une implémentation en assembleur d'un AES-128 avec recacul de table. Cette implémentation à été chargée sur une ATMEL ATMega163 8-bit. Cette carte à puce est connue pour fuiter. Les attaques sont



(a) $\sigma = 3$. **(b)** $\sigma = 4$.

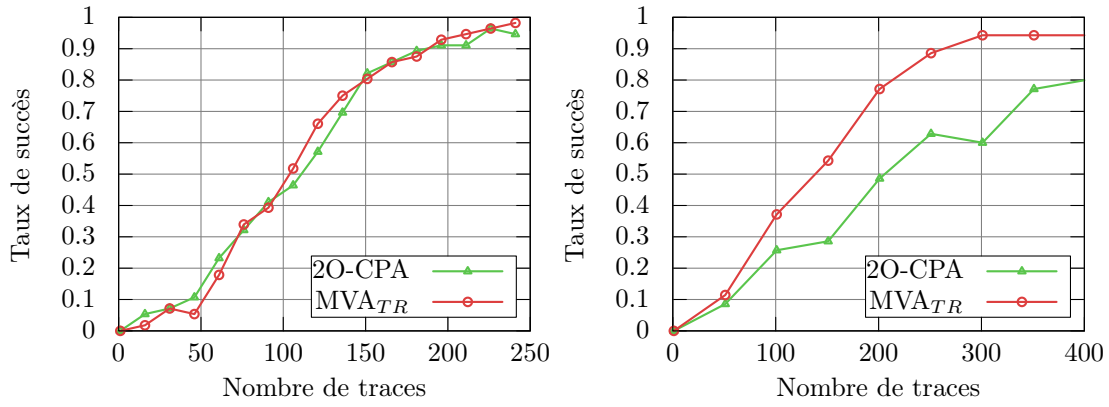
Figure 6: Comparaisons entre 2O-CPA et MVA$_{TR}$



(a) $\Omega = 3, \sigma = 3$. **(b)** $\Omega = 4, \sigma = 3$.

Figure 7: Comparaison entre ΩO-CPA et MVA$^{\Omega}_{CS}$

effectuées sur des acquisitions basées sur les ondes électro-magnétiques émises lors de chiffrements. Nous pouvons remarquer en Figure 8a que sur ces traces les deux attaques MVA$_{TR}$ et 2O-CPA ont la même efficacité. En effet, puisque ces acquisitions sont peu bruitées nous sommes proches de la borne inférieure de l'interval de variance. Néanmoins quand on ajoute du bruit comme en Figure 8b l'attaque MVA$_{TR}$ devient plus efficace ce qui confirme nos résultats.



(a) Comparaison sur traces brutes

(b) Comparaison avec ajout de bruit

Figure 8: Comparaisons des SR des attaques MVA$_{TR}$ et 2O-CPA

Chapitre 6 Ordre mixte.

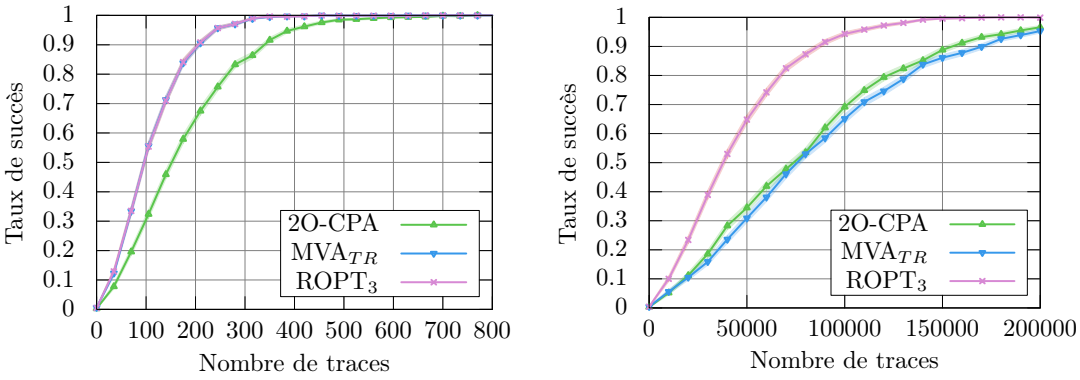
Dans ce chapitre nous approximons le distingueur d'une attaque "Template" basée sur le maximum de vraisemblance en attaques de degrés inférieurs. En exploitant cette décomposition nous montrons qu'il est possible de construire des attaques hautement multivariées qui restent efficaces quand la vraisemblance ne peut pas être calculée du faite de sa complexité. Un algorithme basé sur un recalcul de table aléatoire est utilisé comme illustration pour construire une nouvelle attaque qui surpasse les attaques de l'état de l'art. Cette nouvelle attaque combine deux degrés d'attaque et est capable d'exploiter des fuites multi-dimensionnelles ce qui explique son efficacité.

Le maximum de vraisemblance est le distingueur qui maximise la probabilité de succès. En ce sens nous le considérons donc comme le distingueur optimal. Cette vient néanmoins avec une complexité calculatoire importante. En effet, dans le cas d'une attaque contre une implémentation masquée et calculée dans un ordre aléatoire (« shufflée »), cette complexité dépends de différents paramètres dont le factoriel de la taille des permutations utilisées pour construire l'ordre aléatoire. Cette valeur peut être extrêmement importante quand la taille des permutations augmente. Dans ce cas du fait de sa complexité le distingueur optimal ne peut être calculé. Dès lors il convient de trouver une alternative au maximum de vraisemblance dans ces cas.

L'idée de notre approche est d'utiliser les premier termes de la décomposition du distingueur

optimal en sa série de Taylor. Nous définissons par ROPT_L cette nouvelle attaque exploitant les L premiers termes de la série de Taylor. Notre approche basée sur l'approximation de Taylor va être efficace d'un point de vue calculatoire (réduction de la complexité) mais reste également efficace du point de vue du nombre de traces nécessaires pour construire les attaques.

Nous prenons comme exemple pour illustrer cette nouvelle attaque le cas de d'une implémentation d'un schéma masquage avec recalcul de table dans lequel le recalcul de table est effectué dans un ordre aléatoire. Le gain en termes d'efficacité calculatoire comparé au distingueur optimal est évident puisque cette attaque est calculable ce qui n'était pas le cas du distingueur optimal. Afin d'évaluer son efficacité en terme de nombre de traces nécessaires pour effectuer les attaques nous avons utilisé des simulations. Nous supposons ici un modèle de fuite en poids de Hamming. Nous appliquons une attaque en utilisant les trois premiers termes de la série de Taylor. En effet, nous savons grâce au chapitre précédent qu'une attaque d'ordre trois est efficace. Pour comparer notre attaque aux attaques antérieures nous choisissons comme attaques de référence l'attaque 2O-CPA et l'attaque MVA_{TR}. Nous remarquons que pour les faibles bruits comme par exemple pour un bruit d'écart type $\sigma = 1$ (Figure 9a) que l'attaque ROPT_3 est proche de l'attaque MVA_{TR}. Cela veut dire que le terme dominant dans la série de Taylor est le terme de degré trois. Inversement lorsque le bruit est élevé $\sigma = 13$ (Figure 9d) l'attaque ROPT_3 est proche de l'attaque 2O-CPA. Cela veut dire que le terme dominant est le terme de degré deux. Le meilleur cas pour l'attaque ROPT_3 comparé aux attaques de référence est lorsque $\sigma = 8$. Dans ce cas les attaques 2O-CPA et MVA_{TR} ont les mêmes résultats et l'attaque ROPT_3 à des résultats deux fois meilleurs.



(a) $\sigma = 1$ **(b)** $\sigma = 8$

(c) $\sigma = 9$ **(d)** $\sigma = 13$

Figure 9: Attaque sur le recalcul de table

Contents

List of Figures

LIST OF FIGURES

List of Tables

CHAPTER 1

Introduction

Contents

1.1 Introduction to the cryptography

Cryptography aims at ensuring the security of the communications between several parties. Historically mainly used for military or diplomatic purposes, it has seen its uses exponentially grow up with the development of the communication technologies.

To ensure the security, *cryptographic tools* (*primitives*) provide one or more of the following properties:

Confidentiality. This property guarantees that the information transmitted during the communication will be only intelligible by authorized parties. The confidentiality of data will be

obtained by *encryption* schemes. Such schemes transform an initial message called the *plain-text* to an unreadable, for any unauthorized party, message called the *cipher-text*. This transformation process is called the encryption. The algorithms which perform the revers process are called *decryption* algorithms. This process is called decryption.

Data integrity. This property guarantees that the data have not been modified by a malicious party. To ensure the Data integrity a *hash* function may be used. A *hash* function can be roughly defined as a one way mapping from a string of arbitrary length to a binary string of fixed length. The security of such schemes depends on the difficulty to find two binary strings with the same output. Then the data integrity is constructed as follows: first the hash value of some data is initially computed. To verify that the data have not been altered the hash value of these data is computed and compared to the original one. Noticed that the integrity of the hash value has to be ensured in some manners.

Authentication. This property can be applied both on data and parties. The entity authentication ensures that the communication is made between the expected parties. The data authentication could ensure the origin of the message, the emission date of the message. The Authentication is often provided by *digital signature* schemes. A digital signature depends on, a secret only known by the signer and the message. Exploiting this secret the signer is the only one able to generate the digital signature. Then anyone else should be able to verify if this number has been generated by the signer, in general a digital signature is provided with a public value to allow the verification.

Non repudiation. This property ensures that parties will not deny actions or commitments. This property can also be ensured by digital signature schemes.

Regarding these properties different kinds of primitive have been developed.

1.1.1 Symmetric Cryptography

The symmetric cryptography also called secret key cryptography is based on the fact that all the parties involved in the communication share the same secret (unknown by anyone else) called the secret key. In a symmetric key primitive only one key is involved and as a consequence the encryption and decryption algorithms take as input the same key. Symmetric encryption schemes ensure the confidentiality of the data. Block cipher algorithms are often used in symmetric cryptography. Formally the block ciphers can be defined as functions which take as input

two bit-strings: the key and the plain-text and output a bit-string: the cipher-text. The key length and the plain-text length are two parameters associated with the block cipher algorithm. Notice that for any key and for any block cipher, the block cipher restricted to this key is a bijection. In a classical usage the secret key is randomly generated and shared between the two parties and the security will depend on the secrecy of the key. Several block ciphers have been proposed, non exhaustively we can cite: Rijndael (118), Serpent (14), Twofish (149), RC6 (143), MARS (28) which are the Advanced Encryption Standard finalists, but also the Data Encryption Standard (119) (DES), the Triple Data Encryption Standard (119) (TDES), the International Data Encryption Algorithm (90) (IDEA)... Let us present in more details two of them which are standardized: the Data Encryption Standard (119) (DES) expended from the Lucifer (153) cipher and the Advanced Encryption Standard (AES (118) originally called Rijndael)

Data Encryption Standard (119). The DES algorithm is one of the most widely used block cipher algorithm. It was adopted by the National Bureau of Standards (NBS) now known as the National Institute of Standards and Technology (NIST) in 1976 as a standard. The DES takes as input a key of 56 bits and 64 bits of plain-text. The algorithm consists in 16 rounds of a "Feistel network". In order to increase the size of the key an evolution of the DES has been proposed: the Triple DES where three DES are executed sequentially with a three times longer key.

Advance Encryption Standard (118). In order to replace the DES, the NIST lunched in 1998 a new competition to select the new standard block cipher. Fifteen candidates have been proposed and at the end of the selection process the Rijndael algorithm has been chosen to be the AES. This cipher takes as input a key of length 128, 256, 512 depending on the level of security required. The plain-text has a length of 128 bits. The algorithm consists of 10, 12 or 16 rounds depending on the length of the key. Each round is composed of different operations ensuring the *diffusion* and *confusion*.

In modern block ciphers the security against statistical attacks is provided by two main properties the *confusion* and the *diffusion* (151).

Diffusion. The diffusion property hides the relationship between the plain-text and the cipher-text. In general this property is implemented using *permutations*. The diffusion will impress on each output bit information provided by each input bit. Generally the *diffusion* property

comes from manipulations of the order of bits and is provided in the AES by the ShiftRows and MixColumns.

Confusion. The confusion property hides the relationship between the cipher-text and the secret key. The confusion property ensures that each bit of the cipher-text is related to the secret key by a complex function. In general this property is implemented using *substitutions*. The *confusion* comes from non linear operations provided by the SubBytes in the AES. The operations providing the confusion are often the target of the side-channel analysis. Indeed the substitutions often manipulated several bits of the key in one operation.

As the length of data may be higher than the plain-text length different mechanisms can be used to encrypt data using a block cipher. In these cases the way the block cipher are used on the different parts is called the cipher mode (see (103) for details).

The Electronic CodeBook (ECB) mode. In this mode the whole text to cipher is split into words which have as size the input size of the block cipher. Then each word is ciphered with the same key.

The Cipher-Block Chaining (CBC) mode. Similarly to the ECB mode the text is split into words. The first word is xored with an Initialization Vector (IV) and the result of this xor is ciphered. The other words are proceed in an iterative process where the i-th word is xored with the result of the $(i-1)$-th step.

These two modes are often used in practice nevertheless other modes exist such as the CBC with counter (CBCC), the Cipher FeedBAck (CFB), the Output FeedBack (OFB), the counter mode (CTR), the counter based CTR (CTRC)...

As no random IV is used in the ECB mode this mode is often the one used to perform side-channel analysis.

1.1.2 Asymmetric Cryptography

At the opposite of the symmetric cryptography in asymmetric cryptography, also called public cryptography, a public key, derived from a secret key, only known by one party, is shared to other parties. It is this asymmetry in the information held by the different parties which gives the name of asymmetric cryptography. Asymmetric cryptography provides two kinds of primitive ensuring the *confidentiality* (asymmetric encryption) and the *data integrity* (digital signature). While symmetric schemes are based on a sequence of substitutions and permutations the asymmetric

schemes are based on one-way functions meaning that they are easy to compute and difficult to invert with a trapdoor meaning that they are easy to invert given an extra information. The TrapDoor One Way Function (TOWF) are relied to a *hard problem*. In one hand the security of the asymmetric is based on the difficulty to solve this problem without any extra-information. On the other hand the problem can be easily solved knowing extra-information.

1.1.2.1 Diffie and Hellman key exchange

One of the main drawback of symmetric primitives is that an a priori shared secret is needed. In order to allow secure communications without this shared secret Diffie and Hellman have presented in (45) an asymmetric cryptography primitive which allows a key exchange without *trusted authority*. The hard problem is in this case the Diffie-Hellman Problem (DHP) Definition 1 which is itself closed to the Discrete Logarithm Problem (DLP) Def. 2. Nevertheless the DHP is at least as easy as the DLP.

Definition 1 (Diffie-Hellman Problem)**.** *Given a prime p a generator α of $\mathbb{Z}_p^*$, and elements $\alpha^a \mod p$ and $\alpha^b \mod p$ find $\alpha^{ab} \mod p$.*

Definition 2 (Discrete Logarithm Problem)**.** *Given a prime p a generator α of $\mathbb{Z}_p^*$, and an element $\beta \in \mathbb{Z}_p^*$ find the integer x $0 \leq x \leq p-2$ such that $\alpha^x \equiv \beta \mod p$.*

1.1.2.2 Rivest Shamir Adleman (RSA) cryptosystem (144)

In 1978 Rivest Shamir and Adleman proposed a first example of asymmetric encryption algorithm and signature scheme in (144) it is the well known RSA cryptosystem. This cryptosystem is based on the difficulty of factoring integers which are the products of large primes. The underlying hard problem is the RSA problem (RSAP) Def. 3 which is to find the inverse of the TOWF of the protocol. This problem is closely linked to the factoring problem Def 4. Indeed finding a solution of the factoring problem gives a solution to the RSAP, as a consequence the RSAP is at least as easy as the factoring problem.

Definition 3 (RSA Problem)**.** *Given a positive integer n that is a product of two distinct odd primes p and q, a positive integer e such that $\gcd e\,(p-1)\,(q-1) = 1$, and an integer c find a m such that $m^e \equiv c \mod n$.*

Definition 4 (Factoring Problem)**.** *Given a positive integer n find its prime factorization; that is written $n = p_1^{e_1} p_2^{e_2} ... p_k^{e_k}$ where the p_i are pairwise distinct primes and $e_i \geq 1$.*

1.1.2.3 El Gamal cryptosystem (50)

In 1988 Taher ElGamal has proposed an asymmetric cryptosystem and a signature scheme based on the DLP. While the first publication of Diffie Hellman provides a first example of asymmetric cryptography algorithm for the key exchanges ElGamal algorithms are the first ones to used the DLP to build encryption scheme and digital signature.

The previous schemes need to compute a modular exponentiation that is for the given positive integer m the modulus n and an exponent e computing $m^e \mod n$.

1.1.2.4 Elliptic Curves cryptosystem (82, 110)

In 1985 Neal Koblitz (82) and Victor Miller (110) have independently presented a new cryptosystem based on Elliptic Curves (EC). The EC Cryptography is based on the difficulty to solve the Elliptic Curve Discrete Logarithm Problem (ECDLP). Currently the best known algorithm to solve this problem has an exponential running time. This allows smaller key length compared to RSA. Indeed the factoring problem can be solved in sub-exponential times. Different cryptographic algorithms have been proposed exploiting ECDLP. Smaller keys are a real advantage especially in the embedded systems where the memory can be limited. ECC provides signature, encryption and key exchange primitives.

Definition 5 (Discrete Logarithm Problem)**.** *Given an elliptic curve E over a finite field K, an integer k and points $P \in E(K)$ and $Q = [k]P \in E(K)$ find k where $[k]P = \underbrace{P + ... + P}_{k \text{ times}}$.*

1.2 Physical Attacks

Classical cryptographic algorithms are guessed to be secured in a *black box model* where only the algorithm and some couples plain-text cipher-text are known by the attacker. In such models classical cryptanalysis exploit intrinsic properties of the cryptographic algorithms in order to recover sensitive information. Nevertheless cryptographic algorithms never come alone and are implemented on physical devices e.g smart-cards, micro-controllers. In the rest of the manuscript we call the device on which the cryptographic algorithms are running the Device Under Test (DUT).

These implementations are vulnerable to particular attacks the so called *Physical Attacks*. Such attacks exploit the interaction of the DUT with its environment.

Physical Attacks can be decomposed into two main categories (98).

Active Attacks. In an active attack the attacker could modify the execution of the algorithms executed on the DUT. The *Fault Attacks* are a classical example of active attacks. They consist in faulting the execution of the algorithms in order to recover the sensitive values. Since their introduction by Boneh et al. in (17) they received many publications introducing different ways to inject fault, different target values and different exploitation steps. To inject fault an attacker can use Optical fault injection (152) (using for example a Light Amplification by Simulated Emission of Radiation), Electromagnetic fault injections (136) or other means (17, 88, 148)... The target of Fault injection can be several kinds of values such as the Input Parameters (10), the Data processing path (126) or the instruction processing path (178)... Finally different methods have been proposed to exploit the faults in order to recover sensitive values: Differential Fault Attacks (DFA) (15), Safe error attacks (183)...

Passive Attacks. Contrary to the Active Attacks the Passive Attacks do not interact with the DUT. They may consist in observing some physical informations the so called *Side Channel Information* emitted by the device during the execution of the cryptographic algorithms. Attacks exploiting such information are called the Side Channel Attacks later denoted by SCA. SCA exploit various type of physical measurements such as the power consumption of the devices (83), theirs electromagnetic emanations (58, 135), the duration of the algorithm (84) but also more exotic ones such as the acoustic emanations, optical emanations or even the heat.

Following this first characterization of Physical Attacks another complementary decomposition is possible. An attack is defined as

- *Semi-invasive* when the attacker can modify the external package of the DUT but does not modify the internal structure.
- *Invasive* when it needs permanent modifications of the DUT.
- *Non invasive* when the attacker only observes the physical emanation generated by the DUT.

Depending on the context (mostly the way to inject fault) Fault Attack can be Semi-invasive active attacks or Invasive active attacks. The rest of this manuscript deals with SCA which in this nomenclature are Non-invasive passive or Semi-invasive passive attacks when the package is removed in order to help the attack.

1.3 Side Channel Attacks

A first example of SCA was provided by the program TEMPEST led by the US government to study the possible compromising emanations. One of the first scientific presentation about SCA is due to Kocher in (84) in which an attacker is able to recover sensitive data by exploiting the time of execution of asymmetric cryptographic algorithms such as Diffie-Hellman (45) and RSA (144).

The first paper dealing with the exploitation of power consumption is due to Kocher et al. and published in (83).

The general setup for building SCA works as follows. An attacker will run a cryptographic algorithm one time or more with different inputs. During these executions the attacker acquires (e.g. using an oscilloscope) the physical leakages he targets. Such acquisitions are often called *traces* or *measurements*. Exploiting these measurements he can directly recover the key in Simple Power Analysis (84) (SPA). In other approaches (83) the attacker extracts the value of the secret key by comparing a set of measurements and a guessed value called the prediction model using a statistical tool the *distinguisher*. Since the first publications about SCA many different *distinguishers* have been proposed. There is an incentive to choose the optimal distinguisher. This "optimality" depends on a value to maximize. In practice it is often the probability of success.

The distinguishers are ones of the core tools of SCA. They allow to compare the measurements and the prediction model. Thus the secret key is recovered based on this comparison. Through the literature many different distinguishers have been presented. A Side Channel Attack can be seen as the overall process which allows to recover the secret key based on physical measurements. This exploitation often uses measurements assuming a specific leakage and using a particular distinguisher.

1.3.1 Notations

1.3.1.1 General notations

In this thesis uppercase letters are used for random variables (e.g. U) and the corresponding lowercase letters for their realizations (e.g. u), calligraphic capital letters denote sets (e.g. $\mathcal{U}$).

Bold symbols are used to denote vectors that have length Q, the number of measurements. The empirical mean (resp the empirical standard deviation) of a vector $\mathbf{u}$ is denoted by $\overline{\mathbf{u}}$ (resp. ($\widehat{\sigma}_{\mathbf{u}}$)).

Namely, $\mathbf{X}$ denotes a set of Q random variables i.i.d. with the same law as X. X represents a leakage measurement. So, $\mathbf{X}$ is a $Q \times D$ matrix which represents a set of Q measurements (also named the queries) each measurement being of length D i.e. composed of D *leakage samples*. Some cryptographic algorithms involve random values. $\mathbf{R}$ denotes a set of random variables i.i.d. with the same law as R which represents this value. $\mathbf{t}$ denotes the set of input-texts of the measurements $\mathbf{X}$ with $t \in \mathcal{T}$ where $\mathcal{T}$ is the set of possible input-texts. Let $k^\star$ be the secret key. $k^\star \in \mathcal{K}$ where $\mathcal{K}$ is the set of the possible keys. In this manuscript we suppose that the computations are done on n-bit words which means that these words can be seen as elements of $\mathbb{F}_2^n$. As a consequence both $k^\star$ and t belong to $\mathbb{F}_2^n$.

Notations $\mathbf{X}_q$ and $\mathbf{X}^{(d)}$ are used to denote the d-th column and the q-th line of the matrix $\mathbf{X}$, respectively.

Definition 6 (Sensitive variable)**.** *A* sensitive variable *is an internal variable proceeded by the cryptographic algorithm which depends on a subset of the inputs not known by the attacker (e.g. the secret key but also the secret random value).*

Definition 7 (Selection Function)**.** *Let g be a mapping which maps the input data to a* sensitive variable*. This function is called selection function.*

$$\begin{array}{rccc} g: & \mathbb{F}_2^n \times \mathbb{F}_2^n \times \mathcal{R} & \longrightarrow & \left(\mathbb{F}_2^n\right)^D \\ & k^\star, T, R & \longmapsto & g\left(k^\star, T, R\right) \quad . \end{array} \tag{1.1}$$

Definition 8 (Leakage Function)**.** *The way that the sensitive values leak in the measurements depends on a leakage function that is a specific characteristic of the target device:*

$$\begin{array}{rccc} \Psi: & \left(\mathbb{F}_2^n\right)^D & \longrightarrow & \mathbb{R}^D \\ & V & \longmapsto & \Psi\left(V\right) \quad . \end{array} \tag{1.2}$$

Based on this definition a measured leakage is modeled by:

$$X = \Psi\left(g\left(k^\star, T, R\right)\right) + N, \tag{1.3}$$

where the random variable N denotes an independent additive noise. This noise is assumed to be independent between each measurement.

The function g depends on the algorithm while the function Ψ is a characteristic of the device. The function Ψ is in general not known by the attacker. As a consequence in order to perform SCA an attacker will often has to make a prediction on the unknown leakage function by selecting a specific *leakage model*. Let denote by $\widehat{\Psi}$ this leakage model $\widehat{\Psi}: \left(\mathbb{F}_2^n\right)^D \to \mathbb{R}^D$.

In order to simplify our notations we introduce the function $y = \widehat{\Psi} \circ g$, where $\circ$ denotes the composition law. Additionally we introduce the following notations for the *prediction model*:

$$Y^\star = y\left(k^\star, T, R\right) = \widehat{\Psi}\left(g\left(k^\star, T, R\right)\right) \quad , \tag{1.4}$$

$$Y_k = y\left(k, T, R\right) = \widehat{\Psi}\left(g\left(k, T, R\right)\right) \quad . \tag{1.5}$$

The prediction model denotes the association of a leakage model and a key guess k and gives an estimation of a possible values for the leakages. The key index will often be removed where there is no ambiguity.

Depending on the DUT the attacker can used some classical leakage models.

In software implementations the Hamming weight is often used. The Hamming weight (HW) is simply the sum of the bits and is given by:

$$\begin{array}{rccc} \mathsf{HW}: & \mathbb{F}_2^n & \longrightarrow & \mathbb{R} \\ & g\left(k^\star, T, R\right)^{(d)} & \longmapsto & \sum_{b \leq n} \left[g\left(k^\star, T, R\right)^{(d)}\right]_b \end{array} \quad , \tag{1.6}$$

where $[\cdot]_b : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ is the projection onto th b-th bit.

A more precise model is to assume weights on each value of the bit decomposition. Let us denote by WHW (for weighted Hamming Weight) this kind of functions which take as input a vector of weight α:

$$\begin{array}{rccc} \mathsf{WHW}: & \mathbb{R} \times \mathbb{F}_2^n & \longrightarrow & \mathbb{R} \\ & \left(\alpha, g\left(k^\star, T, R\right)^{(d)}\right) & \longmapsto & \sum_{b \leq n} \alpha_b \left[g\left(k^\star, T, R\right)^{(d)}\right]_b \end{array} \quad . \tag{1.7}$$

In a hardware implementation the classical model prediction is the Hamming Distance HD which is simply the HW of the Exclusive Or (XOR) of internal values.

1.3.2 Leakages descriptions

The SCA are closely linked to the measurements they exploit. In this section we present an overview of possible kinds of measurements. As already mentioned SCA exploit the additional information which can leak during the execution of the cipher through a side channel. In order to exploit the physical emanations the attacker acquired them using an *acquisition chain*. The chain represents all the tools which transform the physical leakages into the digital data which are exploited by the attackers.

Example 1 (DPA contest V4 (169) acquisition chain)**.** *An example of an acquisition chain for ElectroMagnetic emanation (EM) acquisitions can be as follows: the EM are acquired using an EM near-field probe, then the signal is increased using a Preamplifier and finally digitalized using an oscilloscope.*

After the step of digitalization each measurement can be seen as a vector where the components are the leakage samples.

The shape of the measurements will differ from an acquisition setup to another. They will also differ depending on the target. Indeed the measurements acquired from a hardware coprocessor may often be "smaller" and "noisier" than measurements acquired from a software implementation executed on smart-card. Indeed the length of software measurements can exceed the million of points whereas the hardware may not exceed thousands. Nevertheless in both cases the number of possible exploitable samples may be small compared to the size of the measurements.

The leakage function may vary from an experiment to another. Indeed it can be efficiently modeled by a modulation of a leakage model.

Definition 9 (Modulated Traces)**.** *Let us now define a* modulated trace *as a trace in which each time sample can be expressed as a modulation of a model (static in time) plus an independent noisy part:*

$$X = \left(\beta^{(d)}\widehat{\Psi}(g(k^\star, T, R)) + N^{(d)}\right)_{d\leq D} = \beta \cdot \widehat{\Psi}(g(k^\star, T, R)) + \left(N^{(d)}\right)_{d\leq D}, \tag{1.8}$$

where β is a vector in $\mathbb{R}^D$ and each $N^{(d)}$ is drawn from an independent identical distribution $\mathcal{N}$. In specific, the variance of the noise does not depend on the time sample $d \leq D$.

This notion is illustrated in Fig. 1.1.

Of course in other cases the same model does not appear in several time samples. Indeed the leakages may be impacted by many external factors such the activation of external peripheral which is typically the case on System on Chip (SOC). The values manipulated during the execution of the algorithm may be also manipulated by different combinational logic parts which have specific leakage functions.

1.3.3 Distinguisher

As already mentioned a common way to perform SCA is to compare a prediction model and the measurements. In these cases distinguishers are used. The different distinguishers can be divided into categories depending on the a priori knowledge they required. Nevertheless in some cases direct exploitations of the measurements are possible.

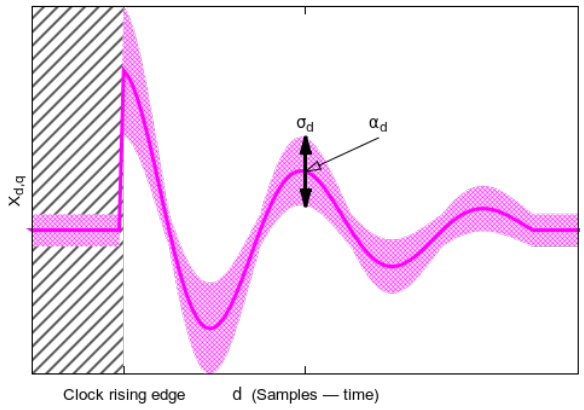


Figure 1.1: Example of a modulated trace

1.3.3.1 Simple Side Channel Analysis

In some cryptographic implementations the operations proceeded depend on the value of the key. In these cases a "simple" observation of the side channel can allow to recover the key. These attacks called Simple Side Channel Analysis (SSCA) can reveal directly the entire key. A first example of such attacks has been presented by Kocher in (85). In this attack the attacker takes advantage of the non constant time execution of operations of the algorithm.

Simple Power Analysis (SPA) will exploit the dependencies between the secret key and the power leakages. An example was proposed by Coron (38) where an attacker exploits the fact that a point addition is performed only if the current bit is one. As a consequence by identifying the sequence of points addition and points doubling the attacker is able to recover the scalar.

1.3.3.2 Profiled attacks

The profiled attacks assume a powerful attacker which has a full control of a clone device of the DUT often called open sample. Exploiting this clone device the attacker is able to build an accurate estimation of the leakage function. These attacks are generally multivariate ($D > 1$) and are seen as the most powerful kind of attacks. To do so profiled attacks are computed into two steps:

- *profiling phase*: during this phase the attacker uses his clone device to estimate the leakage function. In order to estimate this function he generally needs the knowledge of $k^\star$ and the ability to change it. It is in generally not possible on commercial devices.

- *attack phase*: during the attack phase the attacker exploits the knowledge of the leakage function to build a powerful attack to recover the secret key $k^\star$.

A first example of Profiled attack the so called *template attacks* has been given by Chari et al. in (31). They are known to be the most powerful attacks in an information theoretic sense; in the context of side-channel analysis, this means that they minimize the error probability, that is they maximize the probability of success (provided templates are estimated without error). They consist in estimating the likelihood for a key guess. In the learning phase the probability density function $p\left(X|T, K^\star\right)$ of the leakages X knowing T and $K^\star$ is computed. Then in the attack phase using the Bayes' theorem the likelihood is computed. In the original publication the noise N is assumed to be Gaussian, and as a consequence $p\left(X|T, K^\star\right)$ is the density of a Gaussian distribution.

Following this first example of Profiled Attacks several other Profiled Attacks have been proposed (75, 77, 93, 147).

Recently the optimal distinguisher has been presented depending on the knowledge of the attacker about the leakage function (74). When the attacker has a full knowledge of the leakage function the optimal distinguisher is the Maximum Likelihood (ML) of a Template attacks scenario.

Definition 10 (Maximum Likelihood Attacks ML)**.** *When the leakage function Ψ is known the optimal distinguisher is given by:*

$$\begin{array}{rccc} \mathrm{ML}: & \mathbb{F}_2^n \times \mathbb{R}^{D\times Q} & \to & \mathbb{F}_2^n \\ & \mathbf{y}, \mathbf{x} & \mapsto & \mathrm{argmax}_{k\in\mathcal{K}} \prod_{q=1}^{Q} p_N\left(x_q - y_q\right) \quad , \end{array} \tag{1.9}$$

where p_N is the density function of the noise, and where y_q is the qth realization of the prediction model Y_k (Eqn. (1.5)).

1.3.3.3 Non profiled Attacks

The non profiled attacks scenario assumes a less powerful attacker. Such attacks are generally univariate $D = 1$ and take into account all the leakage samples independently (i.e. the distinguisher is applied successively on each leakage sample). These attacks are generally applied on a set of traces and are often called *differential attacks*.

The first distinguisher which has been presented is the *difference of mean* used in the mono-bit DPA. The data set is split into two with respect to one bit of the result of the selection function. The key is recovered by taking the argmax of the distance between the mean of these two sets.

Definition 11 (Differential Power Analysis (DPA)). *The* DPA *attacking the* b^{th} *bit is :*

$$\begin{array}{rccl} \mathsf{DPA}: & \mathbb{F}_2 \times \mathbb{F}_2^n \times \mathbb{R}^Q & \rightarrow & \mathbb{F}_2^n \\ & b, \mathbf{t}, \mathbf{x} & \mapsto & \text{argmax}_{k \in \mathcal{K}} \frac{\sum_{q=1}^{Q} \mathbf{x}_q \times \mathbb{1}_{\left\{[g(k^\star, \mathbf{t}_q, R)]_b = 1\right\}}}{\sum_{q=1}^{Q} \mathbb{1}_{\left\{[g(k^\star, \mathbf{t}_q, R)]_b = 1\right\}}} - \frac{\sum_{q=1}^{Q} \mathbf{x}_q \times \mathbb{1}_{\left\{[g(k^\star, \mathbf{t}_q, R)]_b = 1\right\}}}{\sum_{q=0}^{Q} \mathbb{1}_{\left\{[g(k^\star, \mathbf{t}_q, R)]_b = 0\right\}}} \quad . \end{array} \tag{1.10}$$

Extended version of this attack has been proposed in order to take into account several bits (11, 104).

Brier et al. have presented in (18) the *Pearson Correlation Coefficient* ρ as distinguisher in the Correlation Power Analysis (CPA) which takes into account all the bits of the sensitive variables.

Definition 12 (Correlation Power Analysis (CPA)). *The* CPA *using the correlation coefficient as distinguisher is :*

$$\begin{array}{rccl} \text{CPA}: & \mathbb{F}_2^n \times \mathbb{R}^Q & \rightarrow & \mathbb{F}_2^n \\ & \mathbf{y}, \mathbf{x} & \mapsto & \text{argmax}_{k \in \mathcal{K}} \widehat{\rho}\,[\mathbf{x}, \mathbf{y}] \quad , \end{array} \tag{1.11}$$

where $\widehat{\rho}$ *denotes the estimator of the Pearson Correlation Coefficient.*

A classical model prediction for the CPA is the HW, or the HD. Nevertheless if the actual leakage function is different and close to WHW, the results of the CPA decrease.

As a consequence new distinguishers have been proposed to take into account this kind of leakage functions.

The Linear Regression Analysis (95) (LRA) uses the *coefficient of determination* as distinguisher. In this attack the weights which impact each bit of the sensitive value are recovered using a Liner Regression approach. The sketch of this attack is as follows, an attacker makes an estimation of the coefficients α for each key hypothesis k let us denote by $\widehat{\alpha}$ these estimated weights. These estimated weights are taken such that they minimize the euclidean distance. Finally the good key is the one maximizing the coefficient of determination.

Definition 13 (Linear Regression Analysis (LRA)). *The* LRA *using the coefficient of determination as distinguisher is given by:*

$$\begin{array}{rccl} \text{LRA}: & \mathbb{F}_2^n \times \mathbb{R}^Q & \rightarrow & \mathbb{F}_2^n \\ & \mathbf{t}, \mathbf{x} & \mapsto & \text{argmax}_{k \in \mathcal{K}}\, 1 - \frac{\|\mathbf{x} - \mathfrak{y} \cdot \widehat{\alpha}\|_2}{\|\mathbf{x} - \overline{\mathbf{x}}\|_2} \quad , \end{array} \tag{1.12}$$

where $\mathfrak{y}$ *denotes the* $Q \times n + 1$ *matrix which represents the bit decomposition of the sensitive variables (with a constant term), and* $\widehat{\alpha} = (\mathfrak{y}^{\mathsf{t}} \cdot \mathfrak{y})^{-1} \cdot \mathfrak{y} \cdot \mathbf{x}$.

Many other distinguishers have been presented depending on a partition variance as distinguisher (3, 159)

Another approach to relax the constraints linking the leakage function and the prediction model is the use of the Mutual Information (MI) as distinguisher. The use of the MI has been presented in the field of side channel analysis by Gierlichs et al. in (59) to build the so called Mutual Information Analysis (MIA). The main advantage of such approach is that the a priori knowledge needed to build $\widehat{\Psi}$ could be small, the only assumption needed on $\widehat{\Psi}$ is to be non-injective (59, 131).

The MI allows to evaluate the dependence between two random variables and is given by:

$$\mathsf{I}[X;Y] = \mathsf{H}[X] - \mathsf{H}[X|Y], \text{where} \tag{1.13}$$

$$\mathsf{H}[X] = \int_{-\infty}^{-\infty} p(x)\log p(x)\,\mathrm{d}x \text{ and where} \tag{1.14}$$

$$\mathsf{H}[X|y] = -\sum_{y}\int_{-\infty}^{-\infty} p(x,y)\log p(x|y)\,\mathrm{d}x \ . \tag{1.15}$$

Definition 14 (Mutual Information Analysis (MIA))**.** *The* MIA *exploiting the* MI *as distinguisher is given by:*

$$\begin{array}{rccc} \text{MIA}: & \mathbb{F}_2^n \times \mathbb{R}^Q & \to & \mathbb{F}_2^n \\ & \mathbf{y}, \mathbf{x} & \mapsto & \text{argmax}_{k\in\mathcal{K}}\, \widehat{\mathsf{I}}[\mathbf{x};\mathbf{y}] \ . \end{array} \tag{1.16}$$

Remark 1. *The computation of the empirical* MI *needs to compute the empirical probability density functions of* $\mathbf{x}$*.*

1.4 Samples Selection and Dimensionality Reduction

As presented in Sec. 1.3.2 the number of exploitable leakage samples may be small compared to the length of the measurements. As the Templates Attacks are naturally multivariate they can be applied directly on the whole traces but this leads to excessive computational loads and memory consumption (31). To reduce the number of points on which the attacks will be proceed a variety of methods exits. Such methods can be divided into two main categories: the *sample selection* methods and the *dimensionality reduction* methods.

Those two kinds of method are conceptually closed as they lead to smaller traces. The sample selection methods aim at finding in the traces the most relevant samples and select them. Sample selection methods lead as a consequence to a data selection. The dimensionality reduction methods aim at recombining the different samples in order to increase the possible exploitable

information contained in one point. The dimensionality reduction methods lead therefor to a data transformation. As a consequence the major difference occurs when monovariate distinguisher are applied on the reduced traces. Indeed in the cases of monovariate distinguishers as the ones presented in SubSect. 1.3.3.3 there is no need of sample selection as the distinguishers are applied independently on each time sample. Then the results of the attacks with sample selection will be similar to the results of attacks without sample selection. While, dimensionality reduction may be an interesting tool as it combines information spread over different leakage samples in an exploitable one and as a consequence better results can be expected.

1.4.1 Sample Selection

Different metrics for sample selection have presented in the field of SCA. They allow to classify the samples in order to identify the most relevant ones. The attacker selecting an arbitrary number of samples maximizing the sample selection metric.

In their original article on Template Attacks Chari et al. (31) used the difference of mean in order to recover the sample points which are linked to the key.

In the rest of this manuscript $\overline{\mathbf{x}}^{(d)}[y] = \frac{\sum_{q=1}^{Q} x_q^{(d)} \times \mathbb{1}_{\left\{g(k^\star, \mathbf{t}_q, R)^{(d)} = y\right\}}}{\sum_{q=1}^{Q} \mathbb{1}_{\left\{g(k^\star, \mathbf{t}_q, R)^{(d)} = y\right\}}}$.

Definition 15 (Difference of Mean (31) (DOM))**.** *The* DOM *sample selection method is given by:*

$$\begin{array}{rccc} \text{DOM}: & \mathbb{F}_2 \times \mathbb{F}_2^n \times \mathbb{R}^Q & \rightarrow & \mathbb{F}_2^n \\ & b, \mathbf{t}, \mathbf{x} & \mapsto & \text{argmax}_{d \leq D} \sum_{y \neq y'} \overline{\mathbf{x}}^{(d)}[y] - \overline{\mathbf{x}}^{(d)}[y'] \quad . \end{array} \tag{1.17}$$

An extension of this method based on Sum Of Square Difference has been proposed in (60), where the square of the difference of means is computed.

Definition 16 (Sum Of Square Difference (60) (SOSD))**.** *The* SOSD *sample selection method is given by:*

$$\begin{array}{rccc} \text{SOSD}: & \mathbb{F}_2 \times \mathbb{F}_2^n \times \mathbb{R}^Q & \rightarrow & \mathbb{F}_2^n \\ & b, \mathbf{t}, \mathbf{x} & \mapsto & \text{argmax}_{d \leq D} \sum_{y \neq y'} \left(\overline{\mathbf{x}}^{(d)}[y] - \overline{\mathbf{x}}^{(d)}[y']\right)^2 \quad . \end{array} \tag{1.18}$$

An extension which takes into account the variance the so called Sum Of Square pairwise T-differences (SOST) has also been proposed in (60).

One can notice that this selection method is closed to the DPA distinguisher. Similarly other distinguishers have been adapted to select samples in the context of Template Attacks.

Definition 17 (Pearson Correlation (98) (CPA))**.** *The* CPA *sample selection method is given by:*

$$\begin{array}{rccc} \text{CPA}: & \mathbb{F}_2^n \times \mathbb{R}^Q & \rightarrow & \mathbb{F}_2^n \\ & \mathbf{y}, \mathbf{x} & \mapsto & \text{argmax}_{d \leq D} \, \widehat{\rho} \left[\mathbf{x}^{(d)}, \mathbf{y} \right] \quad , \end{array} \tag{1.19}$$

where $\widehat{\rho}$ denotes the estimator of the Pearson Correlation Coefficient.

Similarly other distinguishers can be used as metric for sample selection. An overview of the different methods can be found in (51).

1.4.2 Dimensionality Reduction

While Sample Selection aims at selecting one or more relevant samples the Dimensionality Reduction aims at recombining the data in a way that information spread over multiple samples will be combined in only one. In the area of SCA such methods have been presented as preprocessing for Template Attacks (as the selection samples) but also for differential attacks.

Different dimensionality methods have been proposed. The Principal Component Analysis (PCA)) is a classical statistical tool for dimensionality reduction (79).

Definition 18. *The* PCA *is an orthonormal linear projection of the data, which maximizes the variance of the projected subspace of dimension $D' \leq D$. More formally, we search the projection which maximizes the variance of the projected data. For the first dimension of the subspace this leads to:*

$$\max_{\|u_1\|=1} Var[Xu_1] = \max_{\|u_1\|=1} {}^t u_1 S_T u_1,$$

where S_T is the covariance matrix given by ${}^t (\mathbf{x} - \overline{\mathbf{x}}) (\mathbf{x} - \overline{\mathbf{x}})$. For the second dimension, as we want an orthonormal projection, this yields:

$$\max_{\substack{\|u_2\|=1 \\ u_2 \cdot u_1 = 0}} {}^t u_2 S_T u_2.$$

The process is iterated for each dimension $D' \leq D$.

This method has been introduced in the area of SCA by Archambeau et al. in (2) in the context of the template attacks. It has been also presented in order to improve the results of differential attacks in (5).

Remark 2. *In the context of* SCA *the* PCA *is generally used to maximize the between-class variance. In this case the covariance matrix computed that is $\sum_{y \in F_2^n} \left(\overline{\mathbf{x}}^{(d)}[y] - \overline{\mathbf{x}}^{(d)} \right) \left(\overline{\mathbf{x}}^{(d)}[y] - \overline{\mathbf{x}}^{(d)} \right)^t$.*

Recently a new dimensionality reduction tool has been introduced in the context of SCA, the so called Linear Discriminant Analysis (158) (LDA). While the PCA aims at maximizing the variance of the projected subspace the LDA maximizes the ratio between the between-class variance and the within-class variance. This means that compared to the approach using the PCA is that the within-class is taken into account.

Definition 19. *The* LDA *is an orthonormal linear projection of the data, which maximizes the ratio between the between-class variance and the within-class variance of the projected subspace of dimension $D' \leq D$.*

For the first dimension of the subspace this leads to:

$$\max_{\|u_1\|=1} \frac{{}^t u_1 S_B u_1}{{}^t u_1 S_W u_1}.$$

where S_B and S_W are respectively the between-classes scatter matrix and the within-classes scatter matrix.

$$S_B = \sum_{y \in F_2^n} \left(\overline{\mathbf{x}}^{(d)}[y] - \overline{\mathbf{x}}^{(d)} \right) \left(\overline{\mathbf{x}}^{(d)}[y] - \overline{\mathbf{x}}^{(d)} \right)^t , \tag{1.20}$$

$$S_W = \sum_{y \in F_2^n} \sum_{q=1}^{Q} \left(x_q^{(d)} \times \mathbb{1}_{\left\{ g(k^\star, t_q, R)_d = y \right\}} - \overline{\mathbf{x}}^{(d)}[y] \right) \left(x_q^{(d)} \times \mathbb{1}_{\left\{ g(k^\star, t_q, R)_d = y \right\}} - \overline{\mathbf{x}}^{(d)}[y] \right)^t . \tag{1.21}$$

The process is iterated for each dimension $D' \leq D$.

Nevertheless even if these methods have already been compared (see for example (32)) there is no clear and systematic analysis of their overall behaviors in the context of SCA.

1.5 Protection methods

In order to protect cryptographic algorithms in embedded devices different countermeasures have been developed. A first approach is to mitigate the part depending on sensitive variables in the measured leakages.

Adding Noise. A designer can increase the noise in leakages by adding some operations in parallel of the execution of sensitive variables. As the sensitive variables, these dummy operations will leak. Another approach is to execute dummy operations between the execution of sensitive operations. The consequence is a misalignment in the traces which can be seen as noise.

Specific Logic. On the other hand a designer can reduced the signal available. A way to attend this goal is to use *dual rail logic*. In such implementations the same logic is implemented twice and during the execution of the algorithm the second logic takes as input the complementary values.

The two methods lead to a diminution of the Signal to Noise Ratio (SNR) which represents an evaluation of the ratio between information and noise.

Whereas the both methods provide security by increasing the number of traces needed to perform the attacks, their respective behaviors are device dependent. As a consequence a formal analysis of the security provided by these countermeasures is a difficult task. Therefor other countermeasures have been study for which the security characteristic can be formally grounded.

1.5.1 Data Masking Scheme

Data Masking schemes (29, 96) are one of the most used protection method against SCA as the provided security can be formally grounded. The aim of data masking is to make the sensitive data independent from the variables manipulated and then independent from the measured leakages. Intuitively, masking aims at increasing the order of the statistical moments (in the leakage distributions) that reveal sensitive information (30, 78).

Interestingly the consequence of such protections is closed to a noise addition as the exploitable information will be melt into more noisy information. As a consequence we will often see the SNR as a useful tool to study implementations and specifically protected implementations.

The rationale of masking schemes goes as follows: each sensitive variable is randomly split into Ω shares (using $\Omega-1$ masks), in such a way that any tuple of $\Omega-1$ shares manipulated during the masked algorithm is independent from any sensitive variable. A masking scheme which reaches this property is called *perfect* masking scheme. This splitting is done using an invertible operation $\perp$ and random values as masks. Let y be the sensitive values and $y_i, \Omega > i > 0$ being the $(\Omega - 1)$ random values drawn from a uniform law. Then the values manipulated in the data masking schemes are the masks and the masked value : $y_0 = y \perp y_1 \perp ... \perp y_{\Omega-1}$.

Different operations $\perp$ have been presented in the literature. The boolean maskings are the most classical one, they use the XOR denoted by $\oplus$ as invertible operation. They are often used in practice as the XOR operation is easy to implement both in hardware and in software, moreover in many of the classical block ciphers the permutation operations are linear with the XOR. The arithmetic or additive masking schemes (38) using the modular addition can also be used in order to protect cryptographic algorithms in which some operations are linear with the

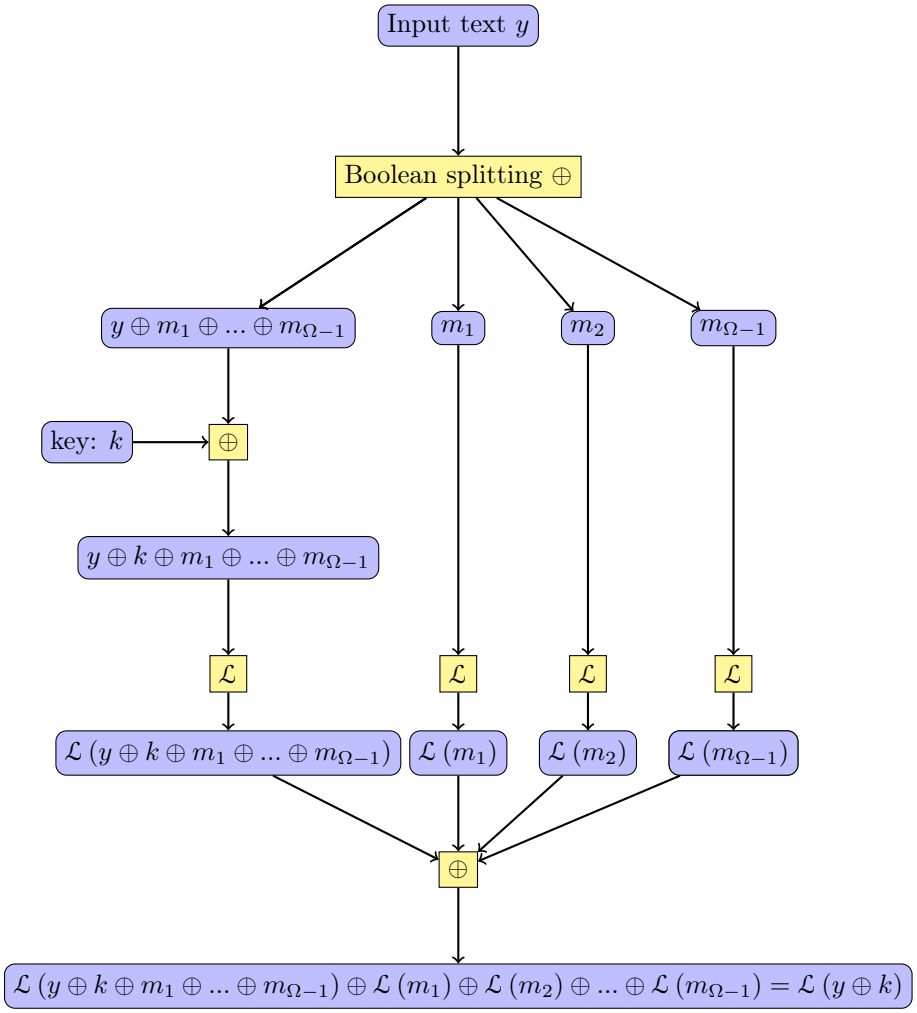


Figure 1.2: Schematic of a linear part of a masking scheme of a block cipher.

addition. It is for example the case with IDEA, SHA-1 or SHA-2. Another type of masking schemes are the multiplicative (61) ones using the modular multiplication. They can be used to compute the substitution part of some block ciphers. Affine masking schemes (57) combine a multiplicative and a boolean masking scheme.

The linear operations (with respect to $\perp$) are easy to implement. These operations are applied on all the shares and the results are combined by $\perp$ (see Fig. 1.2).

At the opposite it could be particularly challenging to compute non-linear parts of the algorithm, such as for example the S-Box of AES (a function from n bits to n bits).

A classical approach to compute the non-linear parts is to used masked tables. The idea

is to store a masked version of the non-linear part in a look up table which removes the input mask and remasks by the output mask. As those masks change for each encryption a possibility is to precompute off-line the table for each possible input and output mask in a global Look-up Table (130, 162) . This approach can be prohibitive (e.g. it could be necessary to store a $2^n \times 2^n \times 2^n \times 2^n$ table) and then is not often used in practice. An other approach for software implementations is to use precompute table (1, 29, 105). In such implementation the non linear parts are stored in a lookup table and at each encryption a new table is computed for the specific input and output masks. Recently, Coron presented at EUROCRYPT 2014 (39) a table recomputation scheme for any Ω. Since this countermeasure aims at high-order security ($\Omega > 1$), it requires one full table precomputation before every S-Box call.

Variety of hardware implementations can be found using specific mask gates (171) and the possible algebraic representation of the block cipher (16, 141). An example for AES is to use the tower field representation to reduce the inversion problem in a smaller field. The Threshold implementations (116) are a particular kind of hardware implementations which are designed to ensure the security up to a certain order in presence of glitches.

1.5.2 Protection for Asymmetric Cryptography

The Asymmetric Cryptography algorithms can be vulnerable to SPA indeed the classical algorithms used for the modular exponentiation used in RSA or the scalar multiplication for ECC use conditional branches depending on the exponent or the scalar. As an example the classical scalar multiplication algorithm performs a double and an add when the scalar bit is one or just a double when the scalar bit is zero. Therefor by simply looked at the sequence of operations it is possible to recover the entire key within one trace. It exists in the literature many different scalar multiplications/modular exponentiations.

To avoid SPA a first countermeasure makes the scalar multiplication regular meaning that the same operations are performed independently from the value of the current bit. Different algorithms have been proposed to attend this goal such as the Double and Add Always or the Montgomery Ladder algorithm.

Nevertheless this kind of countermeasure does not protect against differential attacks. An example of differential attacks has been proposed in (38). Similarly to the protected implementations of symmetric cryptographic algorithms, different randomization countermeasures have been developed, they will differ on the randomized values but their behaviors are similar. Indeed

for each execution of the cryptographic algorithm one value or more is randomized which makes this particular variable not attackable by differential means.

Different types of variable can be targeted and as a consequence different variables can be randomized.

Scalar/Exponent Splitting. In order to be protected the scalar (resp. the exponent) has to be randomized. For ECC a first example of scalar randomization has been proposed in (38): the group scalar randomization. This countermeasure takes advantage of the group structure of the EC. It consists in adding to the scalar a multiple of the order of the curve. Nevertheless this countermeasure is the target of the Carry Leakage Attacks (54). Therefor different other scalar randomization countermeasures have been proposed. They consist in splitting the scalar into two values. In the additive splitting (34) a scalar multiplication, with the secret scalar minus a random number as scalar, is first computed followed by the scalar multiplication with the random number as scalar. In multiplicative splitting (172) a first scalar multiplication is done with a random number. The result of this multiplication will play the role of the point in a second modular multiplication where the scalar is the secret scalar times the inverse of the random number. An other splitting often used is the Euclidean Splitting (33). Similarly to the multiplicative splitting a first scalar multiplication is performed using a random number. Its result serves as point for two other scalar multiplications. The first one used the rest of the euclidean division between the secret scalar and the random, the second scalar multiplication used the quotient of this euclidean division.

Similar countermeasures can be found in RSA implementation. Different scalar blindings have been proposed in the literature. An example of exponent blinding is provided in (84). In this case a random multiple of the Euler's function of the modulus is added to the exponent. The Exponent splitting (34) divides the exponent into two parts. Then two scalar multiplications are performed.

Data randomization. Specific attacks can target the point on the curve manipulated in the scalar multiplication rather than the scalar itself. To prevent such attacks data randomizations often called Points Blindings have been proposed. The first example of Points Blinding was given in (38). In this countermeasure a pseudo-random point on the curve is stored on the chip and updated at each iteration. The scalar multiplication is computed on the sum of the initial point and a random one. Other countermeasures have been proposed in order to protect the ECC

algorithms against the Doubling Attack (55) which consists in the comparison of two traces with the base point and the double of this point. Indeed the same value appears in the two traces if the scalar meets some properties. A first example of countermeasures is the Random Projective Coordinates (38). Using the Jacobian coordinate representation the point is randomized by multiplying each coordinate by a random number. A second countermeasure is the Random Curve Isomorphism (80). In this countermeasure the scalar multiplication is computed on an isomorphic random curve.

In the case of RSA the data randomization can be provided in two different ways. The first way is to randomize the modulus. This randomization can be achieved by multiplying the modulus by a random number. The second way is to randomize the base (84) which consists in a blinding of the message and the modulus. A random multiple of the modulus will be added to the message.

This list represents a non exhaustive presentation of possible attacks and countermeasures against ECC and RSA. A synthesis for ECC can be found in (115).

1.6 Attacks on the countermeasures

1.6.1 High Order Attacks

The masking schemes provide security against the SCA presented in Subsect. 1.3.3 nevertheless a particular kind of SCA has been developed in order to counter the masking schemes the so called High order Side Channel Attacks (HOSCA).

An $\Omega - 1$ order masking scheme will ensure that any combination of less than Ω values will not leak any information about the key.

The overall principle to counter the masking scheme is to combine the leakages of the Ω shares (29). Based on this principle a whole kind of attacks has been defined the HOSCA (29, 106, 133, 177).

1.6.1.1 Template Attacks

The Template Attacks have been extended to defeat the masking schemes in (121). The rational of the attacks is similar to the unprotected cases. In the first learning phase the attacker learns the density probability $p\left(X|T, K^\star, M\right)$ for all the possible $(T, K^\star, M)$.

Remark 3. *As the target is $\Omega - 1$ order masking schemes, X should be at least of size $D = \Omega$.*

Remark 4. *The profiling step is significantly easier if the masks are known during this step.*

In the attack phase the contribution of the masks is taken into account by computing the mean over the possible values of the masks.

$$p\left(K|X,T\right) = \sum_{m\in\mathcal{M}} p\left(X|T,K^{\star},m\right)p(m) \tag{1.22}$$

Recently the optimal distinguisher has been derived in the context of a template attacks against protected implementations. In the case of masking the optimal distinguisher which maximizes the success rate is given by (23) the Maximum Likelihood.

Definition 20 (Maximum Likelihood)**.** *When the $y\left(t,k,R\right)$ are known and the Gaussian noise N is i.i.d. across the queries (measurements) and independent across the dimension, then the optimal distinguisher is:*

$$\begin{array}{rccl} \text{OPT:} & \mathbb{R}^{DQ}\times\mathbb{R}^{DQ} & \longrightarrow & \mathbb{F}_2^n \\ & (\mathbf{x}, y\left(\mathbf{t},k,R\right)) & \longmapsto & \underset{k\in\mathbb{F}_2^n}{\operatorname{argmax}} \sum_{q=1}^{Q} \log \mathbb{E} \exp \dfrac{-\|\mathbf{x}_q - y(t_q,k,R)\|^2}{2\sigma^2} \end{array} \tag{1.23}$$

where the expectation operator $\mathbb{E}$ is applied with respect to the random variable $R\in\mathcal{R}$, and the norm is the Euclidean norm $\|\mathbf{x}_q - y(t_q,k,R)\|^2 = \sum_{d=1}^{D}(x_q^{(d)} - y^{(d)}(t_q,k,R))^2$.

1.6.1.2 Combination functions

While the Template attacks are multivariate, standard distinguishers are in general monovariate. Thus in order to combine the leakages of each share specific functions are used. These functions are called *combination functions*. The Higher Order Differential Power Analysis (HODPA) are based on a combination function and a distinguisher presented in Subsect. 1.3.3.3 (also called differential attacks). In the SCA literature several combination functions have presented taken conjointly with the distinguisher they completely defined the HODPA. They can be applied both on the leakage traces or the models. Formally we have that a combination function C is given by:

$$\begin{array}{rccc} C: & \mathbb{R}^D & \to & \mathbb{R} \\ & X & \mapsto & C\left(X\right) \ . \end{array} \tag{1.24}$$

Of course to have a sound HODPA we have that $D \geq \Omega$.

Several examples have been proposed in the case of a second order masking scheme. Let us denote by $X^{(0)}$ the leakage of the first share and $X^{(1)}$ the leakage of the second one i.e.

$$X^{(0)} = \Psi^{(0)}\left(\texttt{Sbox}[T\oplus M\oplus k^{\star}]\right) + N^{(0)} \ , \tag{1.25}$$

$$X^{(1)} = \Psi^{(1)}\left(M\right) + N^{(1)} \ . \tag{1.26}$$

Similarly we define by $Y^{(0)}$ and $Y^{(1)}$ their respective prediction model.

Definition 21 (Product Combining Function (30))**.** *The product combining function multiplies the shares:*

$$\begin{array}{rccc} C_p: & \mathbb{R}\times\mathbb{R} & \to & \mathbb{R} \\ & X & \mapsto & X^{(0)}\times X^{(1)} \ . \end{array} \tag{1.27}$$

Definition 22 (Absolute Difference Combining Function (107))**.** *The Absolute Difference Combining Function computes the absolute difference of the two shares:*

$$\begin{array}{rccc} C_{ad}: & \mathbb{R}\times\mathbb{R} & \to & \mathbb{R} \\ & X & \mapsto & |X^{(0)}-X^{(1)}| \ , \end{array} \tag{1.28}$$

In the two previous examples the combination function is only applied on the leakages. More exotic high order attack functions have been proposed where two different combination functions are used for the leakages and the prediction models (121).

Prouff et al. shown in (133) that the best approach, when the leakage function is the Hamming Weight, is to used the centered product combination function in order to combine the leakages and the prediction model.

Definition 23 (Centered Product Combining Function (133))**.** *The centered product combining function computes the centered product of the two shares:*

$$\begin{array}{rccc} C_{cp}: & \mathbb{R}\times\mathbb{R} & \to & \mathbb{R} \\ & X & \mapsto & \left(\left(X_0-\mathbb{E}\left[X^{(0)}\right]\right)\times\left(X^{(1)}-\mathbb{E}\left[X^{(1)}\right]\right)\right) \ , \end{array} \tag{1.29}$$

Additionally Prouff et al. shown in (133) that for a given combination function C_X applied on the measurements to maximize the absolute value of the correlation the prediction models have to be combined by:

$$\begin{array}{rccc} C_Y: & \mathbb{F}_2^n\times\mathbb{R} & \to & \mathbb{R} \\ & X & \mapsto & \mathbb{E}\left[\left(Y^{(0)}-\mathbb{E}\left[Y^{(0)}\right]\right)\times\left(Y^{(1)}-\mathbb{E}\left[Y^{(1)}\right]\right)\right] \ , \end{array} \tag{1.30}$$

where C_Y is the same function as C_X but defined over $\mathbb{F}_2^n$.

1.6.2 High Order Differential Attacks

Definition 24 (2O-CPA (133))**.** *We denote by* 2O-CPA *the* CPA *using the centered product as combination function. Namely:*

$$\begin{array}{rccc} \text{2O-CPA}: & \mathbb{R}^Q\times\mathbb{R}^Q\times\mathbb{R}^Q & \longrightarrow & \mathbb{F}_2^n \\ & (\mathbf{x_0},\mathbf{x_1},\mathbf{y}) & \longmapsto & \underset{k\in\mathbb{F}_2^n}{\operatorname{argmax}}\,\widehat{\rho}\left[\left(\mathbf{x}^{(0)}-\overline{\mathbf{x}^{(0)}}\right)\odot\left(\mathbf{x}^{(1)}-\overline{\mathbf{x}^{(1)}}\right),\mathbf{y}\right], \end{array} \tag{1.31}$$

where $\mathbf{y} = \mathbb{E}_M\left(\left(y^{(0)}\left(\mathbf{t}, k, R\right) - \mathbb{E}\left[y^{(1)}\left(\mathbf{t}, k, R\right)\right]\right) \odot \left(y^{(1)}\left(\mathbf{t}, k, R\right) - \mathbb{E}\left[y^{(1)}\left(\mathbf{t}, k, R\right)\right]\right)\right)$, $\odot$ *is the element wise product and* $\widehat{\rho}$ *is an estimator of the Pearson coefficient.*

Then this definition can be straightforward extended to any masking scheme of order Ω.

Definition 25. *The "classical"* ΩO-CPA *is the* HOCPA *built by combining the* Ω *shares using the centered product combination function.*

$$\begin{array}{rccc} \Omega\text{O-CPA}: & \mathbb{R}^{\Omega} \times \mathbb{R} & \longrightarrow & \mathbb{F}_2^n \\ & \left(\left(\overline{\mathbf{x}^{(i)}}\right)_{i \in [\![0,\Omega-1]\!]}, Y\right) & \longmapsto & \underset{k \in \mathbb{F}_2^n}{\operatorname{argmax}}\, \rho\left[\prod_{i=0}^{\Omega-1} \overline{\mathbf{x}^{(i)}}, Y\right] \end{array} ,$$

where $\mathbf{y} = \mathbb{E}_M\left(\prod_{i=0}^{d-1}\left(y^{(i)}\left(\mathbf{t}, k, R\right) - \mathbb{E}\left[y^{(i)}\left(\mathbf{t}, k, R\right)\right]\right)\right)$, *and the product is the element wise product and* $\widehat{\rho}$ *is an estimator of the Pearson coefficient.*

As already mentioned a HOSCA is completely defined by the distinguisher and the combination function. Then all the properties of these attacks can be expressed using the properties of the combination function or the distinguisher.

Definition 26 (Attack order)**.** *The order of an* HODPA *is given by the polynomial degree of its combination function.*

Remark 5. *In this definition the order of* ΩO-CPA *is the number of shares combined that is* Ω.

Proposition 1. *The number of measurements needed to recover the secret key increases exponentially with respect to the order of the masking schemes.*

Proof. A first proof of this proposition was given in (108) for a Gaussian noise and DPA distinguisher. □

Therefor the general assumption is that the better attack against a masked scheme is the minimal order attack.

Recently several HOSCA have been presented which exploit additional leakages (additionally to the masks and the masked sensitive values). In (24, 124, 173) presented an attack which takes into account the leakages which occur during the table recomputation of a masked block cipher with table recomputation steps. Exploiting the multivariate leakages of such algorithms, the results of the HOSCA are greatly increased.

Let us define the leakages of the table recomputation:

$$X^{(2)} = \Psi^{(2)}(M) + N^{(2)} \tag{1.32}$$

$$X^{(3)} = \Psi^{(3)}(M \oplus 1) + N^{(3)} \tag{1.33}$$

$$\vdots$$

$$X^{(2^n+1)} = \Psi^{(2^n+1)}(M \oplus (2^n - 1)) + N^{(2^n+1)} \ . \tag{1.34}$$

One example of an attack exploiting these leakages has been exposed in (174), which we label as 2-stage CPA attack.

Definition 27 (2-stage CPA attack (174))**.**

$$2{\times}\text{CPA}^{mt}(\mathbf{x},\mathbf{t}) = \arg\max_{k\in\mathcal{K}} \widehat{\rho}(\mathbf{x}^{(0)}, y^{(0)}(\mathbf{t},k,\widehat{\mathbf{m}})), \tag{1.35}$$

where $\forall i\ \widehat{m}_i$ is the mask that maximizes the correlation between $x_i^{(\omega)}$ and $y_i^{(\omega)} = \omega \oplus m_i$ for $\omega \in [2, 2^{n+1}]$. This attack is a synergy between a horizontal and a vertical attack. For each trace (separately $\forall i$), the first attack in Eq. (4.7) consists in recovering the mask during the precomputation. Second, a regular CPA using a model in which both the plain-text t and the mask m are assumed as public knowledge is launched. Even if the mask $\widehat{m}$ is not recovered correctly for each trace (since 2^n leakage samples during the precomputation can be seen as small), it can be expected that the value of the mask is recovered by the first horizontal attack probabilistically well enough for it to be biased, i.e., better guessed than random.

In order to mitigate the impact of these attacks some countermeasures have been developed (124) , they are based on random shuffles. These shuffles make the loop index ω unknown by the attacker. This kind of masking are called in this manuscript shuffled table recomputation masking scheme.

1.6.3 Dimensionality parameters of the attacks

The SCA often deal with the combinations of several leakage samples. It is the case of SCA with dimensionality reduction methods for differential attacks or template attacks. The HOSCA often combine different leakage samples to recover key depend values. In this subsection we present three definitions (numbered 28, 29 and 30) which allow to classify the attacks depending on the way they exploit the multiple leakages.

Definition 28 (Dimension)**.** *The dimension D of the attack is the number of leakage samples jointly used to establish one key guess.*

Let X be a leakage measurement. We have that $X = \Psi\left(g\left(k^\star, T, R\right)\right) + N$. The set of all distinct sensitive variables is given by $\Delta = \{g\left(k^\star, T, R\right)^{(i)} | \forall j \neq i,\ g\left(k^\star, T, R\right)^{(i)} \neq g\left(k^\star, T, R\right)^{(j)}\}$.

Definition 29 (δ-variate attacks)**.** *An attack is said δ-variate when $\delta = \#\Delta$.*

Notice that this definition does not take into account the possible duplicate variables but highlights the diversity of exploitable variables.

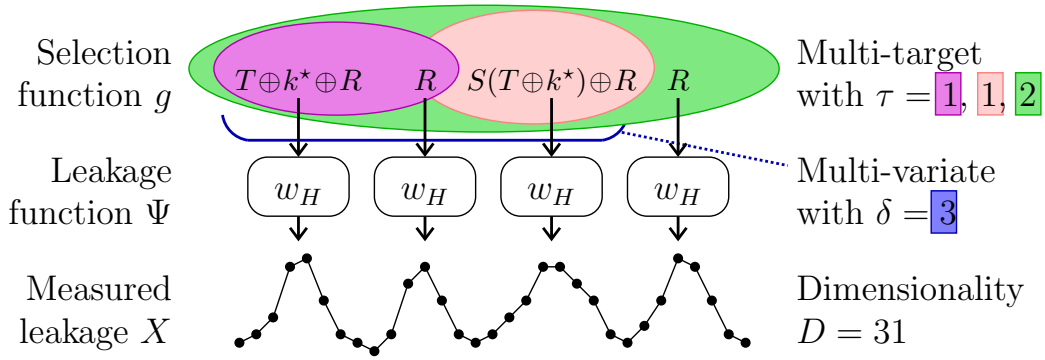


Figure 1.3: Example of multiple leakages

Another approach which combines leakage samples, is the exploitation of the leakages of different target operations. For example an attacker can recover a byte of key during the AddRoundKey, SubBytes or ShiftRows and combined the different key guesses obtained in order to improve the results of these attacks. Such attacks presented in (101) are called Multi-target. In this thesis we give a closed definition:

Definition 30 (τ-target attacks)**.** *An attack is said τ-variate when τ is the number of subset of Δ which depends on the secret key.*

In other words this means that an attacker can build τ different and independent attacks. Of course a better approach is to exploit all these results in one to improve the success probability of the attacks.

An overview of the three previous definitions is given in Fig. 1.3. An attacker can target two different operations the AddRoundKey or SubBytes separately or jointly using three different variables: the random values, the mask value during the AddRoundKey and the mask value during SubBytes. These variables may leak through many different time samples (31 in this example).

In order to illustrate this property let us give some examples in Tab. 1.1.

In this thesis we will investigate how to improve the results of the SCA by increasing these three parameters.

1.6.4 Horizontal Attack

Similarly to the symmetric case some new attacks have been developed to counter the protections in the case of asymmetric cryptography. A classical kind of methods to defeat countermeasure

Attack	Dimension	δ	τ
CPA	1	1	1
CPA with dimensionality reduction	D	1	1
2O-CPA	2	2	1
ΩO-CPA	Ω	Ω	1
Multi-target DPA	$\widetilde{\tau}$	$\widetilde{\tau}$	$\widetilde{\tau}$

Table 1.1: Dimension parameter

is the so called *Horizontal Attacks*. These attacks recover the secret key in only one trace and as a consequence are not impacted by the randomization countermeasures. They exploit the regularity in the power laddering to extract information within one trace.

Remark 6. *In presence of scalar/exponent* Horizontal Attacks *blinding allows to recover the blinded secret key. As this value allows to forge correct outputs it is sufficient for an attacker point of view to recover only this value.*

A first example of Horizontal Attack the so called Big Mac Attack can be found in (179). This attack on RSA implementation consists in finding if the same operand appears into two different multiplications. The rational is as follows, the attacker first select two leakage windows $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ corresponding to two different multiplications. If the Euclidean distance is small then the same operand are manipulated in the two multiplications. Formally if we have the $\|\mathbf{X}^{(1)} - \mathbf{X}^{(2)}\|_2 < t$ where t is a threshold we have the same operand. This attack has been improved in (37) where the Pearson coefficient is used instead of the Euclidean distance. This attack has also been extended to ECC in (7).

Of course Template Attacks are also available against this protection but they require the knowledge of the secret values during the profiling phase (102).

An other approach exploiting cluster algorithms has recently been presented. This kind of attacks allows to recover the secret key in one trace (76, 127). These attacks allow to target directly the bit-values of the secret key.

1.7 Attacks evaluation

As already mentioned different distinguishers and preprocessing methods can be used to build SCA. Then the question arises to know which distinguisher or which combination distinguisher/preprocessing method leads to the better efficiency. Two approaches can be chosen. A

first way is to exploit actual results of attacks to build the comparisons. The other possibility is to theoretically evaluate the results of the attacks based on the expression of the distinguishers.

1.7.1 Empirical Evaluations

In this approach the different attacks are compared using their respective results. Different metrics have been proposed in order to exploit theses results to build comparisons. These metrics are often based on the results of several repetitions of the same attacks over different sets of traces. This means that the empirical distinguisher $\widehat{\mathcal{D}}$ is applied I times on I different sets of traces ${}_i\mathbf{X}$ and ${}_i\mathbf{T}$ with $i \leq I$ build with the same secret key $k^\star$. Notice that the index is before the capital bold letter to avoid ambiguity with the column notation. In the rest of this section let us denote by ${}_i\widehat{k}$ the result of the i-th one with $i \leq I$ i.e. ${}_i\widehat{k} = \text{argmax}_k \, \widehat{\mathcal{D}}\left({}_i\mathbf{X}, {}_i\mathbf{T}\right)$.

Definition 31 (Empirical Success Rate (161) (SR))**.** *Let ${}_i\widehat{k}, i \leq I$ be set of I results of I independent attacks. The empirical Success Rate for an empirical distinguisher $\widehat{\mathcal{D}}$ is given by:*

$$\text{SR}\left[\widehat{\mathcal{D}}\right] = \frac{1}{I} \sum_{1 \leq i \leq I} \mathbb{1}_{k^\star = {}_i\widehat{k}} \ . \tag{1.36}$$

An approach is to compute Guessing Entropy (GE). The GE is given by the mean rank of the secret key.

Definition 32 (Rank)**.** *The rank of the secret key is given by:*

$$\mathit{rank}\left(k^\star\right) = \min_{\mathcal{K}_S \in \mathcal{K}} \left(\#\mathcal{K} - \#\{\mathcal{K}_S | k^\star = \underset{k \in \mathcal{K}_S}{\text{argmax}} \, \widehat{\mathcal{D}}\left({}_i\mathbf{X}, {}_i\mathbf{T}\right)\} + 1 \right) \ . \tag{1.37}$$

Definition 33 (Guessing Entropy (89) (GE))**.** *Let ${}_i\widehat{k}, i \leq I$ be set of I results of I independent attacks. The empirical Guessing Entropy for an empirical distinguisher $\widehat{\mathcal{D}}$ is given by:*

$$\text{GE}\left[\widehat{\mathcal{D}}\right] = \frac{1}{I} \sum_{1 \leq i \leq I} \mathit{rank}\left(k^\star\right) \ . \tag{1.38}$$

When the attack is followed by a key enumeration (175) an attacker may used the *ranking entropy* which takes into account the orders of magnitude of the rank more than the rank of the key itself (100).

Definition 34 (Ranking Entropy (100) (RE))**.** *Let ${}_i\widehat{k}, i \leq I$ be set of I results of I independent attacks. The empirical Ranking Entropy for an empirical distinguisher $\widehat{\mathcal{D}}$ is given by:*

$$\text{RE}\left[\widehat{\mathcal{D}}\right] = \frac{1}{I} \sum_{1 \leq i \leq I} \log \mathit{rank}\left(k^\star\right) \ . \tag{1.39}$$

These empirical metrics provide an efficient way to compare the results of different distinguishers. A distinguisher will be said better than another one if its SR is higher or if its GE is lower when applied on the same set of traces.

1.7.2 Theoretical comparison

The empirical validations do not provide any feedback to explain the behaviors of the success SCA. In order to get a better understanding on relevant parameters, theoretical evaluation can be build. The success of an attack can be expressed using the theoretical Success Rate (SR)

Definition 35 (Theoretical Success Rate (SR))**.** *The theoretical Success Rate for a given distinguisher $\mathcal{D}$ is:*

$$\mathrm{SR}\left[\mathcal{D}\right] = p(k^\star = \widehat{k}) \ . \tag{1.40}$$

Exploiting this formula different analysis have been performed to study the relevant parameters (53, 140, 170) of the SR. In (53) Fei et al. provided a closed form expression of the SR for the DPA distinguisher. Exploiting this expression they showed that the SR depends on three main parameters:

- The number of measurements Q,
- The Signal to Noise Ratio SNR,
- The confusion coefficient κ.

The confusion coefficient is a parameter which expresses the relationship between the correct key and incorrect key hypothesis for a given leakage model. In the initial publication (53) as only the DPA was taken into account the leakage model was the one bit one.

Thillard et al. extended in (170) the notion of confusion coefficient for any leakage models and then give the closed form expression of the SR for the CPA. Recently an approach based (63) on the Success Exponent (SE) allows to derive the closed form expression of the SR for different distinguishers (DPA MIA and CPA). In this manuscript we used their definition of the confusion coefficient.

Definition 36 (Confusion Coefficient (63))**.** *The confusion coefficient between the secret key $k^\star \in \mathcal{K}$ and any key hypothesis $k \in \mathcal{K}$ for a leakage model is given by:*

$$\kappa\left(k^\star, k\right) = p(Y\left(k^\star\right) = Y\left(k\right)) = \mathbb{E}\left[\frac{\left(Y\left(k^\star\right) - Y\left(k\right)^2\right)}{2}\right]. \tag{1.41}$$

Recently these approaches have been extended in the case of HOSCA against masking scheme (46, 94).

1.8 Contributions of the Thesis

1.8.1 Contributions

This thesis is about the security of the protected implementation of cryptographic algorithms. Especially we investigate the possible attack path using specific HOSCA designed to target these implementations. In this context we show that the level of security provided by analysis which take into account only monovariate leakages is overestimated.

In particular starting from the observation that many sensitive leakages appear in the side channel measurements we investigate what is their impact in terms of security. Which can be rewritten as:

How far an attack can go by exploiting multiples leakages?

The answer of this question is provided by exploring the different meaning that the term "multiple" can take in the SCA context.

The Fig. 1.4 represents a summary of the contributions of this thesis. Each arrow highlights a way to improve the success of the SCA. The black marks represent the initial attacks which exploit the minimum number of points. In this manuscript we show that increasing the number of leakage samples used for the attacks by adding similar variables, the D axis in Fig. 1.4 is a powerful tool to increase the probability of success. We show theoretically the best approach in case of a first order attack represented by a green arrow. In the case of a second order attack, represented by a blue arrow, against protected implementations we will present a dimensionality reduction tool which improves the results of the attacks. The gain of these methods increases with the order of the implementations. We can easily see in Fig. 1.4 that the previous methods exploit the multiplicity of leakages with respect with only one axis. The other attacks will exploit the two remaining axes. We give an example of a multi-target attack exploiting the table recomputation step of a masking scheme in Chap. 4. We theoretically prove that this method is the optimal way to exploit the multiple leakages. In Chap. 6 we present a new attack which exploits the leakages along the δ axis. This attack presents the interesting property of not being of minimal order as it is a third order attack represented by a red arrow. Nevertheless this attack exploits only two axes. The last attack exploits the three axes and provides better results compare to the previous chapter assuming a full knowledge of the leakage function. Finally Fig. 1.4 shows the different axes to improve the results of SCA. Indeed the classification of the

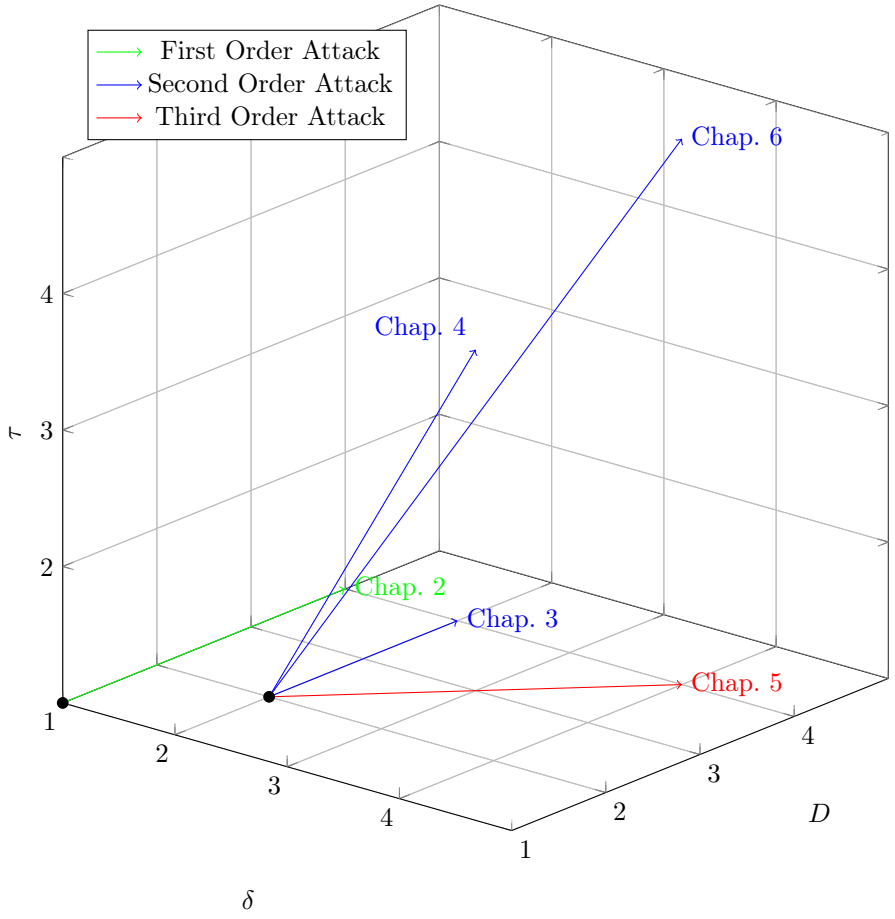


Figure 1.4: Summary of the contributions.

attack allows an attacker to select the axis in which he could improve the attacks. We can notice that these axes are numerous, multiple leakages, multiple variables and multiple targets.

1.8.2 Outlines

In a first Part I of this manuscript we investigate how to increase the SR by exploiting the multiple leakages of the same sensitive variable. This question is often presented in the SCA literature and can be expressed into two ways. How to turn multivariate leakages in monovariates ones and how to do this in the best way. The first question was often answered by applying *dimension reduction* tools. Regarding the classification provided in Subsect. 1.6.3 in this part we will present methods to increase the *dimension* of the attack.

In Chapter 2 we tackle the question of optimality. We theoretically expressed the best

dimensionality. In the case of modulated traces with full knowledge of the modulation we gives the optimal dimensionality reduction.

In Chap. 3 we present new dimensionality reduction tools in the context of HOCPA. We detail the impact of multiple leakages in the case of HOCPA. In particular we show that as the number of possible points increases with the order the impact of dimensionality reduction jointly increases with order. Moreover we present a new tool to go from multivariate leakages to monovariate ones. Relaxing the knowledge of the attacker we present an optimal dimensionality reduction for a particular types of noises.

In the second Part II we investigate how to enhance results of HOSCA by taking into account different variables. In particular regarding the classification provided in Subsect. 1.6.3 we will build highly variate and highly target attacks. In other words we increase the parameter τ and δ of the attacks.

In Chap. 4 we extend the first attack against masking schemes with table recomputation step. In this scenario we derive the optimal distinguisher and show that this approach leads to better results than the state-of-the-art attacks.

It is known that such attacks targeting the table recomputation are a critical threat against these implementations. As a consequence countermeasures have been developed. As already mentioned these countermeasures are based on the randomization of the order of execution of the table recomputation.

In Chap. 5 we build an HOSCA which allows to attacks such countermeasures better than any attacks of the state-of-the-art. While being a higher order attack this new attack gives better results than the minimal order ones. Indeed we show that it is possible to combine the multiple different leakages in one to increase the SNR and thus the SR of the attacks. As a consequence we show a first example in which the minimal order monovariate attacks is not the good tool to asset the security of these shuffled table recomputation implementations of masked block cipher.

In Chap. 6 we investigate theoretically the behaviors of the Maximum Likelihood distinguisher in the context of highly multivariate leakages. For instance we take as example the protected table recomputation algorithm. We show that in this case the ML cannot be computed. Additionally we present a new attack with better results in this context than the attacks of the state-of-the-art. In particular this new attack shows how to combine different order leakages to build efficient HOSCA.

Part I

Dimensionality Reduction a case study in presence of masking

CHAPTER 2

Optimal Dimensionality Reduction with Profiling.

The results presented in this chapter have been published in collaboration with Sylvain Guilley, Annelie Heuser, Damien Marion, and Olivier Rioul in the international Workshop on Cryptographic Hardware and Embedded Systems (CHES 2015) (22).

Contents

2.1 Introduction

The large number of samples to feed into the model has always been a problematic issue for multi-dimensional side-channel analysis. One solution is to use techniques to select *points of interest*. Most of them, such as sum-of-square differences (SOSD) and t-test (SOST) (60), are *ad hoc* in that they result from a criterion which is independent from the attacker's key extraction objective. Recent criteria, such as leakage maximization by sensitive value (2), avoid this problem. Other formal criteria, related to *non-profiled* attacks, have also been proposed (69, 120).

Therefore, there seems to be a converging effort, in both non-profiled and profiled attacks, to *reduce the dimensionality* of multi-dimensional measurements. This desirable property of dimensionality reduction achieves several goals simultaneously:

- it simplifies the side-channel problem (to a single multivariate pdf);
- it concentrates the information (to distinguish using fewer traces); and
- it improves computational speed.

It can be argued, however, that like every preprocessing technique, dimensionality reduction would lose information.

Contributions. In this chapter, we tackle this problem of dimensionality reduction from a theoretical viewpoint. Provided that the attacker has full knowledge of the leakage model, we find that "less is more": the advantages of dimensionality reduction can come with no impact on the attack success probability, while improving computational speed.

We derive that the optimal dimensionality reduction process consists in a *linear combination* of samples, which we explicit as a projection on a specific one-dimensional space. For white noise, it turns out that the improved signal-to-noise ratio (SNR) *after* projection is simply the *sum* of the signal-to-noise ratios at the various samples *before* projection.

Finally, we show that the optimal dimensionality reduction technique asymptotically matches the linear discriminant analysis (LDA) preprocessing. We find that LDA generally outperforms principal component analysis (PCA) for which the SNR increases to a lesser extend than LDA, except in the case of white homoscedastic noise where PCA and LDA become equivalent.

We also validate in practice those results on the DPA CONTEST V2 traces (168).

Review of the state-of-the-art. Dimensionality reduction is part and parcel of profiled attacks. The seminal paper on template attacks (31) is motivated by keeping covariance matrices involved in the training phase sufficiently well conditioned. Manual selection of *relevant leaking points* was discussed in (122) as *educated guesses*. Several automated techniques were proposed, such as sum-of-square differences (SOSD) and t-test (SOST) (60), and also wavelet transforms (44).

Several related metrics were proposed for *leakage detection*. The ANOVA (ANalysis Of VAriance) *F-test* is a ratio between the explained variance and the total variance—see e.g. (32, 42) and (13) where it is named *Normalized Inter-Class Variance* (NICV). Also used for linear

regression analysis, it is known as the *coefficient of determination*, denoted by the symbol "R^2". It is employed in the context of side-channel analysis in (167) as *multivariate regression analysis* in the presence of white noise, and in (155), where it is used as a distinguisher and as a linearity metric.

PCA has been used to compact traces in (5) and templates in (2). The eigenvalues of PCA can be viewed as a security metric (62) or even as a distinguisher (156). This technique is particularly attractive as it can be easily and accurately computed with no divisions involved. It is advocated in (81) that PCA aims at maximizing the inter-class variance, yet it is also important to take the intra-class variance into account. For this reason, LDA has been promoted as an improved alternative. Empirical comparisons were investigated in (137, 158, 165). Unfortunately, despite some differences in terms of qualitative efficiency, there is no clear rationale to prefer one method over the other. In fact, it is unclear which of the intrinsic virtue of statistical tools, their implementation, or the dataset is actually responsible for the performance of dimensionality reduction.

Other works attempted to consider different *objective functions*. In (120), the correct key correlation is taken as the objective to be maximized. A similar goal is pursued in (66, 68, 69, 71). Still other dimensionality reduction techniques exist, such as quadratic discriminant analysis, but have not been studied in the side-channel literature. We mention that similar questions have also been raised in the presence of masking countermeasures (20, 49, 138).

Outline. The remainder of the chapter is as follows. The optimal dimensionality reduction is derived theoretically in Section 2.2. Section 2.3 provides illustrative examples. A comparison with state-of-the-art techniques such as PCA, and LDA (158) is given in Section 2.4. Practical validations on real traces are in Section 2.5. Section 2.6 concludes.

2.2 Theoretical Solution in the Presence of Gaussian Noise

2.2.1 Notations

We adopt a matrix notation. The different queries are indexed by $q = 1, \ldots, Q$, where Q is the number of traces. The different samples in a given trace are indexed by $d = 1, \ldots, D$. Any matrix containing D samples from Q queries is denoted by:

$$\mathbf{M}^D = (M_{d,q})_{d,q} \ ,$$

where $d = 1, \ldots, D$ is a row index and $q = 1, \ldots, Q$ is a column index. For clarity reason we add in this chapter the index D. Two matrices noted side-by-side are implicitly multiplied.

In this chapter in order to derive the optimal attack, it is assumed that the leakage model Ψ is perfectly known to the attacker. We also assumed without loss of generality that is model is centered.

The actual leakage can be written as

$$\mathbf{X}^D = \alpha^D \mathbf{Y}(k^\star) + \mathbf{N}^D, \tag{2.1}$$

where the components of α are not all zero, $k^\star$ is the (unknown) correct key, and $\mathbf{N}^D$ is some random measurement noise. The α and noise distribution are assumed known to the attacker.

We make the stationarity assumption that the noise distribution does not depend on the particular query, that is, the N_q^D are independent and identically distributed independently of the value of q. For a given q, however, the noise samples of N_q^D can be correlated. We assume that N_q^D follows a D-dimensional zero-mean Gaussian distribution $\mathcal{N}(0, \Sigma)$, where covariance matrix Σ is a symmetric positive definite $D \times D$ matrix. Therefore, there exists a matrix $\Sigma^{1/2}$, which is such that $\Sigma^{1/2}\Sigma^{1/2} = \Sigma$. We assume that the matrix Σ is known by the attacker.

2.2.2 Optimal Attack

We focus on the optimal attack as part of our scientific approach to the problem. It is always possible that for some peculiar reason a suboptimal attack actually performs better in the presence of dimensionality reduction. But by the *data processing theorem* (41) any preprocessing like dimensionality reduction can only decrease information about the secret, and, therefore, degrade performance of the *optimal* attack. As a result, it does make sense to minimize the impact of dimensionality reduction on the success rate for this optimal attack so as not to be biased by performance loss or gain due to other factors.

The optimal attack, also known as the template attack (31), consists in applying the *maximum likelihood* principle (74). Having collected Q traces of dimensionality D in a matrix $\mathbf{x}^D$, where each trace x_q^D corresponds to a known plaintext t_q, the best key guess that maximizes the

probability of success is given by

$$\mathcal{D}(\mathbf{x}^D, \mathbf{t}) = \arg\max_k \; p(\mathbf{x}^D | \mathbf{t}, k^\star = k) \tag{2.2}$$

$$= \arg\max_k \; p_{\mathbf{N}^D}(\mathbf{x}^D - \alpha^D \mathbf{y}(k)) \tag{2.3}$$

$$= \arg\max_k \; \prod_{q=1}^{Q} p_{\mathbf{N}^D}(\mathbf{x}^D - \alpha^D y_q(k)) \tag{2.4}$$

where

$$p_{\mathbf{N}_q^D}(z^D) = \frac{1}{\sqrt{(2\pi)^D |\det \Sigma|}} \exp\Big(-\frac{1}{2}(z^D)^{\mathrm{t}} \Sigma^{-1} z^D\Big). \tag{2.5}$$

We have used the independence of the queries in (2.4) and the assumption that at each query, the noise distribution is the same in (2.5).

Notice that, the optimal attack can as well be a *simple power attack* (if $Q = 1$) or a *differential power attack* (if $Q > 1$), using the terminology from (86). Still, in the sequel, we focus on attacks which require many traces ($Q \gg 1$).

2.2.3 Optimal Dimensionality Reduction

We state our main result in the following Theorem 2.2.1:

Theorem 2.2.1. *The optimal attack on the multivariate traces $\mathbf{x}^D$ is equivalent to the optimal attack on the monovariate traces $\widetilde{x}^Q$, obtained from $\mathbf{x}^D$ by the formula:*

$$\widetilde{x}_q = \frac{\big(\alpha^D\big)^{\mathrm{t}} \Sigma^{-1} x_q^D}{(\alpha^D)^{\mathrm{t}} \Sigma^{-1} \alpha^D} \qquad (q = 1, \ldots, Q). \tag{2.6}$$

Proof. By taking the logarithm of the expression to be maximized in Eqns. (2.2)–(2.5), the optimal distinguisher $\mathcal{D}(\mathbf{x}^D, \mathbf{t})$ rewrites

$$\mathcal{D}(\mathbf{x}^D, \mathbf{t}) = \arg\min_k \sum_{q=1}^{Q} \big(x_q^D - \alpha^D y_q(k)\big)^{\mathrm{t}} \Sigma^{-1} \big(x_q^D - \alpha^D y_q(k)\big). \tag{2.7}$$

For each trace index q, the terms in the sum expand to

$$\underbrace{(x_q^D)^{\mathrm{t}} \Sigma^{-1} x_q^D}_{\text{cst. } C \text{ independent of } k} - 2(\alpha^D)^{\mathrm{t}} y_q(k) \Sigma^{-1} x_q^D + (y_q(k))^2 (\alpha^D)^{\mathrm{t}} \Sigma^{-1} \alpha^D$$

$$= C - 2y_q(k)\big[(\alpha^D)^{\mathrm{t}} \Sigma^{-1} x_q^D\big] + (y_q(k))^2 \big[(\alpha^D)^{\mathrm{t}} \Sigma^{-1} \alpha^D\big]$$

$$= \big[(\alpha^D)^{\mathrm{t}} \Sigma^{-1} \alpha^D\big] \left(y_q(k) - \frac{(\alpha^D)^{\mathrm{t}} \Sigma^{-1} x_q^D}{(\alpha^D)^{\mathrm{t}} \Sigma^{-1} \alpha^D}\right)^2 + C'.$$

The latter division is valid since Σ is positive definite and α^D is a nonzero vector. Therefore,

$$\begin{aligned}\mathcal{D}(\mathbf{x}^D, \mathbf{t}) &= \arg\min_k \sum_{q=1}^{Q} \left(y_q(k) - \frac{(\alpha^D)^{\mathsf{t}} \Sigma^{-1} x_q^D}{(\alpha^D)^{\mathsf{t}} \Sigma^{-1} \alpha^D} \right)^2 \left[(\alpha^D)^{\mathsf{t}} \Sigma^{-1} \alpha^D \right] \\ &= \arg\min_k \sum_{q=1}^{Q} \frac{\left(\widetilde{x}_q - y_q(k) \right)^2}{\widetilde{\sigma}^2}, \end{aligned} \tag{2.8}$$

where

$$\begin{cases} \widetilde{x}_q &= \dfrac{(\alpha^D)^{\mathsf{t}} \Sigma^{-1} x_q^D}{(\alpha^D)^{\mathsf{t}} \Sigma^{-1} \alpha^D}, \\ \widetilde{\sigma} &= \left((\alpha^D)^{\mathsf{t}} \Sigma^{-1} \alpha^D \right)^{-1/2}. \end{cases} \tag{2.9}$$

We have shown that (2.7) and (2.8) are equivalent expressions for the same optimal distinguisher, computed either:

- on multivariate traces x_q^D, with a noise covariance matrix Σ, or:
- on monovariate (i.e., scalar) traces $\widetilde{x}_q$, with scalar noise of variance $\widetilde{\sigma}^2$.

□

Theorem 2.2.1 shows that in fact, the optimal attack already integrates an optimal dimensionality reduction. The maximal success rate is not altered.

Definition 37 (Projection vector)**.** *Let V^D be a column of D elements. We call the projection of an acquisition campaign $\mathbf{x}^D$ on V^D the new mono-sample traces $(V^D)^{\mathsf{t}} \mathbf{x}^D$. That is, every trace X_q^D $(1 \leq q \leq Q)$ of the initial campaign is summarized as one sample $(V^D)^{\mathsf{t}} X_q^D = \langle V^D \mid X_q^D \rangle$.*

Based on this definition, Theorem 2.2.1 can be interpreted as follows.

Corollary 1. *The optimal dimensionality reduction is made by a linear combination of the samples where each multivariate trace is projected on the vector $V^D = \frac{\Sigma^{-1}\alpha^D}{(\alpha^D)^{\mathsf{t}}\Sigma^{-1}\alpha^D}$, of size $D \times 1$.*

Proof. By Theorem 2.2.1,

$$\underbrace{\widetilde{x}^Q}_{1 \times Q \text{ matrix}} = \underbrace{\frac{(\alpha^D)^{\mathsf{t}} \Sigma^{-1}}{(\alpha^D)^{\mathsf{t}} \Sigma^{-1} \alpha^D}}_{1 \times D \text{ matrix } (V^D)^{\mathsf{t}}} \underbrace{\mathbf{x}^D}_{D \times Q \text{ matrix}} .$$

□

In addition, after this projection, the leakage becomes scalar and can be characterized by a signal-to-noise ratio as shown in the following

Corollary 2. *After optimal dimensionality reduction, the signal-noise-ratio is given by*

$$\frac{1}{\widetilde{\sigma}^2} = (\alpha^D)^{\mathsf{t}} \Sigma^{-1} \alpha^D.$$

Proof. This is in line with Eqn (2.8). The random leakage $\mathbf{x}^D$ is protected onto V^D to yield $\widetilde{X}_q = Y_q(k) + \widetilde{N}$ $(q = 1, \ldots, Q)$ where $\widetilde{N}$ is an additive white Gaussian noise (AWGN) distributed as $\mathcal{N}(0, ((\alpha^D)^{\mathsf{t}} \Sigma^{-1} \alpha^D)^{-1})$. Recall that the variance of the leakage model has been assumed normalized $= 1$. Therefore, the signal-to-noise ratio equals

$$\frac{\mathrm{Var}(Y_q(k))}{\mathrm{Var}(\widetilde{N})} = \frac{1}{((\alpha^D)^{\mathsf{t}} \Sigma^{-1} \alpha^D)^{-1}} = (\alpha^D)^{\mathsf{t}} \Sigma^{-1} \alpha^D \ .$$

□

The SNR is an interesting metric on its own, because it quantifies how much the signal has been concentrated (its power increased) for a given noise level. Furthermore, the SNR directly relates to the success rate of optimal attacks (53).

2.2.4 Discussion

It is interesting to note that the optimal dimensionality reduction does not depend on the actual distribution of $Y^D(k)$, the deterministic part of the leakage model. This means that irrespective of the leakage function Ψ, the best dimensionality reduction depends only on signal weights α^D and on noise covariance Σ.

Similarly, the optimal dimensionality reduction does not depend on the *confusion coefficient* of the leakage model (53): for identical weight and noise distribution, the optimal linear combination of leakages is the same whether an XOR or a substitution box operation is targeted.

2.3 Examples

2.3.1 White Noise

One interesting situation is when the noise samples are uncorrelated (see for instance (167) for an experimental setup). The covariance matrix Σ is diagonal:

$$\Sigma = \begin{pmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_D^2 \end{pmatrix}.$$

Proposition 2. *For white noise, the optimal dimensionality reduction takes the form:*

$$\widetilde{x}_q = \frac{\displaystyle\sum_{d=1}^{D} \frac{\alpha_d}{\sigma_d^2} x_q^{(d)}}{\displaystyle\sum_{d=1}^{D} \frac{\alpha_d^2}{\sigma_d^2}} \qquad (q = 1, \ldots, Q) \tag{2.10}$$

Proof. Apply Theorem 2.2.1, where Σ^{-1} is diagonal with diagonal entries $1/\sigma_d^2$. □

Let $\mathrm{SNR}_d = \alpha_d^2/\sigma_d^2$ be the initial signal-to-noise ratio at the dth sample *before* dimensionality reduction.

Proposition 3. *For white noise, the equivalent signal-to-noise ratio* after *optimal dimensionality reduction is given by the sum*

$$\widetilde{\mathrm{SNR}} = \sum_{d=1}^{D} \mathrm{SNR}_d \,. \tag{2.11}$$

Proof. By Corollary 2, $\widetilde{\mathrm{SNR}} = (\alpha^D)^{\mathsf{t}}\Sigma^{-1}\alpha^D = \sum_{d=1}^{D} \frac{\alpha_d^2}{\sigma_d^2} = \sum_{d=1}^{D} \mathrm{SNR}_d$. □

Thus, combining independent multidimensional samples within one trace increases the signal-to-noise as if those samples were captured in D independent traces. In this case having Q traces of D samples each is simply the same as having $Q \times D$ independent monovariate traces.

2.3.2 Correlated Autoregressive Noise

A more general situation is when the samples are correlated like an autoregressive process. More precisely, assume that all samples share the same noise distribution of variance σ^2, and that two consecutive noise samples have correlation factor equal to $\rho \in]-1, +1[$. The correlation factors ρ typically models an autoregressive low-pass filtering of the acquisition setup (see Sec. 2.5.2 for a real-world example). The noise covariance matrix takes the Toeplitz form:

$$\Sigma = \sigma^2 \begin{pmatrix} 1 & \rho & \rho^2 & \rho^3 & \cdots & \rho^{D-2} & \rho^{D-1} \\ \rho & 1 & \rho & \rho^2 & \cdots & \rho^{D-3} & \rho^{D-2} \\ \rho^2 & \rho & 1 & \rho & \cdots & \rho^{D-4} & \rho^{D-3} \\ \rho^3 & \rho^2 & \rho & 1 & \cdots & \rho^{D-5} & \rho^{D-4} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \rho^{D-2} & \rho^{D-3} & \rho^{D-4} & \rho^{D-5} & \cdots & 1 & \rho \\ \rho^{D-1} & \rho^{D-2} & \rho^{D-3} & \rho^{D-4} & \cdots & \rho & 1 \end{pmatrix} = \left(\sigma^2 \rho^{|d-d'|}\right)_{1 \le d,d' \le D}.$$

We emphasize that $|\rho|$ is strictly smaller than one in keeping with the assumption that Σ be positive definite. When $\rho = 0$, the noise becomes white as in the preceding subsection.

Proposition 4. *For autoregressive noise, the optimal dimensionality reduction takes the form:*

$$\widetilde{x}_q = \tfrac{1}{\sigma^2(1-\rho^2)}\Big[(\alpha_1 - \rho\alpha_2)x_{q,1} + \textstyle\sum_{d=2}^{D-1}((1+\rho^2)\alpha_d - \rho(\alpha_{d-1} + \alpha_{d+1}))x_{d,q} \\ + (\alpha_D - \rho\alpha_{D-1})x_{q,D}\Big]. \tag{2.12}$$

Proof. It can easily be checked that Σ^{-1} is tridiagonal:

$$\Sigma^{-1} = \frac{1}{\sigma^2(1-\rho^2)} \begin{pmatrix} 1 & -\rho & 0 & 0 & \cdots & 0 & 0 \\ -\rho & 1+\rho^2 & -\rho & 0 & \cdots & 0 & 0 \\ 0 & -\rho & 1+\rho^2 & -\rho & \cdots & 0 & 0 \\ 0 & 0 & -\rho & 1+\rho^2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1+\rho^2 & -\rho \\ 0 & 0 & 0 & 0 & \cdots & -\rho & 1 \end{pmatrix}.$$

Then apply Theorem 2.2.1:

$$\widetilde{x}_q = \tfrac{1}{\sigma^2(1-\rho^2)} \begin{pmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_{D-1} & \alpha_D \end{pmatrix} \begin{pmatrix} 1 & -\rho & \cdots & 0 & 0 \\ -\rho & 1+\rho^2 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1+\rho^2 & -\rho \\ 0 & 0 & \cdots & -\rho & 1 \end{pmatrix} \begin{pmatrix} x_{q,1} \\ x_{q,2} \\ \vdots \\ x_{q,D-1} \\ x_{q,D} \end{pmatrix}$$

and expand. □

Notice that in the optimal dimensionality reduction, each leakage sample $x_q^{(d)}$ is not only weighted by its corresponding α_d but also by its two neighbor weights $\alpha_{d\pm1}$, provided the latter exist.

Proposition 5. *For autoregressive noise, the equivalent signal-to-noise ratio after optimal dimensionality reduction is given by*

$$\widetilde{\mathrm{SNR}} = \tfrac{1}{\sigma^2(1-\rho^2)} \big[\alpha_1^2 + (1+\rho^2)\textstyle\sum_{d=2}^{D-1} \alpha_d^2 + \alpha_D^2 - 2\rho \sum_{d=1}^{D-1} \alpha_d \alpha_{d+1}\big]. \tag{2.13}$$

Proof. Apply Corollary 2:

$$\widetilde{\mathrm{SNR}} = \tfrac{1}{\sigma^2(1-\rho^2)} \begin{pmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_{D-1} & \alpha_D \end{pmatrix} \begin{pmatrix} 1 & -\rho & \cdots & 0 & 0 \\ -\rho & 1+\rho^2 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1+\rho^2 & -\rho \\ 0 & 0 & \cdots & -\rho & 1 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{D-1} \\ \alpha_D \end{pmatrix}$$

and expand. □

Corollary 3. *For equal weights $\alpha_1 = \cdots = \alpha_D = \alpha$, i.e., when initial signal-to-noise ratios $\mathrm{SNR}_1 = \cdots = \mathrm{SNR}_D = \mathrm{SNR}$ are the same, one has*

$$\widetilde{\mathrm{SNR}} = \mathrm{SNR} \times \frac{D(1-\rho) + 2\rho}{1+\rho}. \tag{2.14}$$

Proof. Proposition 5 reduces to

$$\begin{aligned}\widetilde{\mathrm{SNR}} &= \frac{\alpha^2}{\sigma^2(1-\rho^2)}\left(2+(D-2)(1+\rho^2)-2\rho(D-1)\right)\\ &= \frac{\alpha^2}{\sigma^2(1-\rho)(1+\rho)}\left((1-\rho)(D-\rho(D-2))\right)\\ &= \frac{\alpha^2}{\sigma^2}\frac{1}{1+\rho}\left(D-\rho(D-2)\right) = \mathrm{SNR}\times\frac{D(1-\rho)+2\rho}{1+\rho}.\end{aligned}$$

□

In other words, optimal dimensionality reduction has the effect of multiplying the monovariate SNR by the factor $\frac{D-\rho(D-2)}{1+\rho}$. This gain factor is of course equal to 1 for dimension $D=1$, but becomes strictly greater than 1 for larger dimensions, since $\frac{D-\rho(D-2)}{1+\rho} > \frac{D-(D-2)}{2} = 1$ where we have used that $\rho > -1$ or $\frac{1}{1+\rho} > \frac{1}{2}$.

For very small values of correlation ρ, Taylor expansion about $\rho = 0$ gives $\frac{D-\rho(D-2)}{1+\rho} = D - 2(D-1)\rho + \mathcal{O}(\rho^2)$. The SNR gain is equal to the dimension D at first order, which is consistent with Proposition 3. In addition, that gain is never greater than D, since $\frac{D(1-\rho)+2\rho}{1+\rho} \leq \frac{D(1-\rho)+2D\rho}{1+\rho} = D$. Therefore, when $\mathrm{SNR}_1 = \ldots = \mathrm{SNR}_D$, nonzero values of correlation ρ decrease the efficiency of dimensionality reduction, the most favorable situation being the case of white noise samples.

2.4 Comparison with PCA and LDA

When the attacker does not precisely know the model given by Eqn. (2.1), the optimal dimensionality reduction cannot be applied directly. In this section, we analyse theoretically two well-known engineering solutions to reduce the dimensionality: PCA and LDA. Both techniques are based on eigen decompositions.

2.4.1 Principal Components Analysis (PCA)

Principal components analysis aims at identifying directions in the *centered* data set. The directions of PCA are the eigenvectors of ${}^t(\mathbf{x}-\overline{\mathbf{x}})(\mathbf{x}-\overline{\mathbf{x}})$.

Proposition 6. *Asymptotically as* $Q \longrightarrow +\infty$,

$${}^t(\mathbf{x}-\overline{\mathbf{x}})(\mathbf{x}-\overline{\mathbf{x}}) \longrightarrow \alpha^D(\alpha^D)^{\mathsf{t}} + \Sigma. \tag{2.15}$$

Proof. By the law of large numbers,

$${}^t(\mathbf{x}-\overline{\mathbf{x}})(\mathbf{x}-\overline{\mathbf{x}}) \longrightarrow \mathrm{Cov}(X_q^{(d)}, X_q^{(d')})$$

almost surely, where the covariance term can be computed as: $\mathrm{Cov}(X_q^{(d)}, X_q^{(d')}) = \mathrm{Cov}(\alpha_d Y_q + N_q^{(d)}, \alpha_{d'} Y_q + N_q^{(d')})$. When expanding this expression, cross terms disappear by independence of $\mathbf{Y}$ and $\mathbf{N}^D$. There remains:

$$\alpha_d \alpha_{d'} + \Sigma_{d,d'}$$

where we have used the hypothesis that Y_q has unit variance. □

The classical PCA has the drawback that ${}^t(\mathbf{x} - \overline{\mathbf{x}})(\mathbf{x} - \overline{\mathbf{x}})$ depends both on the *signal* and on the *noise*. *Inter-class PCA* has been introduced in (2). The matrix used in the PCA is traded for a more simple matrix $Z^{D,\#Y}$, where each column, indexed by y, is the centered column $\frac{1}{\sum_{\substack{1\leq q\leq Q \\ \widetilde{Y}_q = y}} 1} \sum_{\substack{1\leq q\leq Q \\ \widetilde{Y}_q = y}} X_q^D$. One advantage of this method is that it explicitly takes into account the sensitive variable Y.

It can be easily checked, that, asymptotically, each column Z_y^D tends to $\alpha^D y$ when $Q \longrightarrow +\infty$. Therefore, $Z^{D,\#Y}(Z^{D,\#Y})^{\mathsf{t}}$ tends to a $D \times D$ matrix proportional to $\alpha^D(\alpha^D)^{\mathsf{t}}$. Here, the noise has been averaged away in each class y, which is a second advantage. Therefore, in the sequel, we shall refer to the inter-class PCA of (2) simply as PCA.

We have the following spectral characterization of the asymptotic PCA:

Proposition 7. *Asymptotically, PCA has only one principal direction, namely the vector α^D.*

Proof. By Proposition 6, the PCA matrix tends asymptotically to $\alpha^D(\alpha^D)^{\mathsf{t}}$. This $D \times D$ matrix has rank one, because all its columns are multiple of α^D. Since

$$(\alpha^D(\alpha^D)^{\mathsf{t}})\alpha^D = \alpha^D((\alpha^D)^{\mathsf{t}}\alpha^D) = \left\|\alpha^D\right\|_2^2 \times \alpha^D,$$

α^D is the eigenvector with corresponding nonzero eigenvalue $= \|\alpha^D\|_2^2$. □

Notice that the uniqueness of the eigenvector for PCA holds in our model (2.1). However, Proposition 7 would not hold if e.g., the noise were correlated to the signal.

Remark 7. *The classical PCA has the same eigenvector α^D if the noise is* isotropic*, i.e., white and of same variance in every dimension.*

The paper (2) presents an optimization procedure to find the eigenelements.

Proposition 8. *The asymptotic signal-to-noise ratio after projection using PCA is equal to $\frac{\left\|\alpha^D\right\|_2^4}{(\alpha^D)^{\mathsf{t}}\Sigma\alpha^D}$.*

Proof. After projection on the (asymptotic) eigenvector α^D, the leakage becomes: $(\alpha^D)^{\mathsf{t}}\alpha^D Y_q(k^\star) + (\alpha^D)^{\mathsf{t}} N_q^D$. The projected signal is $((\alpha^D)^{\mathsf{t}}\alpha^D)Y_q(k^\star)$. The projected noise is $(\alpha^D)^{\mathsf{t}} N_q^D$, which

remains centered. Its variance is equal to the expectation of its square:

$$\begin{aligned}\mathrm{Var}((\alpha^D)^{\mathsf{t}} N_q^D) &= \mathbb{E}\left((\alpha^D)^{\mathsf{t}} N_q^D\right)^2 = \mathbb{E}\left((\alpha^D)^{\mathsf{t}} N_q^D (N_q^D)^{\mathsf{t}} \alpha^D\right) \\ &= (\alpha^D)^{\mathsf{t}} \mathbb{E}\left(N_q^D (N_q^D)^{\mathsf{t}}\right) \alpha^D = (\alpha^D)^{\mathsf{t}} \Sigma \alpha^D.\end{aligned}$$

Therefore,

$$\mathrm{SNR}_{\mathrm{PCA}} = \frac{\mathrm{Var}(((\alpha^D)^{\mathsf{t}} \alpha^D) Y_q(k^\star))}{\mathrm{Var}((\alpha^D)^{\mathsf{t}} N_q^D)} = \frac{\mathrm{Var}(\left\|\alpha^D\right\|_2^2 Y_q(k^\star))}{(\alpha^D)^{\mathsf{t}} \Sigma \alpha^D} = \frac{\left\|\alpha^D\right\|_2^4}{(\alpha^D)^{\mathsf{t}} \Sigma \alpha^D}.$$

□

Example 2. *For white noise (Sec. 2.3.1)*

$$\mathrm{SNR}_{PCA} = \frac{\left(\sum_{d=1}^D \alpha_d^2\right)^2}{\sum_{d=1}^D \alpha_d^2 \sigma_d^2}. \tag{2.16}$$

Example 3. *For autoregressive noise (Sec. 2.3.2)*

$$\mathrm{SNR}_{PCA} = \frac{\sum_{d=1}^D \alpha_d^2}{\sigma^2} \frac{1}{1 + \frac{2}{\sum_{d=1}^D \alpha_d^2} \sum_{d=1}^{D-1} \rho^d \sum_{d'=1}^{D-d} \alpha_{d'} \alpha_{d'+d}}. \tag{2.17}$$

We can now compare the performance of the asymptotic PCA to the optimal dimensionality reduction.

Theorem 2.4.1. *The SNR of the asymptotic PCA is smaller than the SNR of the optimal dimensionality reduction.*

Proof. By assumption the noise covariance matrix is symmetric positive definite, hence there exists a matrix $\Sigma^{1/2}$, which is such that $\Sigma^{1/2}\Sigma^{1/2} = \Sigma$. By Cauchy-Schwarz inequality,

$$\left(\langle \Sigma^{-1/2}\alpha^D \mid \Sigma^{1/2}\alpha^D \rangle\right)^2 \leq \left\|\Sigma^{-1/2}\alpha^D\right\|_2^2 \cdot \left\|\Sigma^{1/2}\alpha^D\right\|_2^2.$$

Therefore, $\mathrm{SNR}_{\mathrm{PCA}} = \frac{((\alpha^D)^{\mathsf{t}}\alpha^D)^2}{(\alpha^D)^{\mathsf{t}}\Sigma\alpha^D} \leq (\alpha^D)^{\mathsf{t}}\Sigma^{-1}\alpha^D = \widetilde{\mathrm{SNR}}$. □

Corollary 4. *The asymptotic PCA has the same SNR as the the optimal dimensionality reduction if and only if α^D is an eigenvector of Σ. In this case, both dimensionality reductions are equivalent.*

Proof. Equality holds in Theorem 2.4.1 if and only if there exists a nonzero real number λ such that $\Sigma^{1/2}\alpha^D = \lambda\Sigma^{-1/2}\alpha^D$, i.e., $\Sigma\alpha^D = \lambda\alpha^D$, i.e., α^D is an eigenvector of Σ.

In this case, the optimal protection is on the vector $\Sigma^{-1}\alpha^D = \frac{1}{\lambda}\alpha^D$, which is proportional to the projection vector belonging to the asymptotic PCA. □

Remark 8. *Assume white noise (Sec. 2.3.1) where all values σ_d^2 $(1 \leq d \leq D)$ are different. Then, by Corollary 4, the asymptotic PCA is optimal only if $\alpha^D = (0, 0, \cdots, 0, 1, 0, \cdots, 0)$, which we may consider unrealistic since only one sample out of D would leak secret information.*

In contrast, if $\sigma_1 = \cdots = \sigma_D = \sigma$, the covariance matrix has only one eigenvalue, namely $(1, 1, \cdots, 1)$, which has multiplicity D. Thus, for white homoscedastic noise, PCA is asymptotically optimal if and only if $\alpha_1 = \cdots \alpha_D = \alpha$, that is, the SNR is the same for each sample.

Still in the case of white noise, we can lower bound the SNR of the asymptotic PCA:

Lemma 1. *For white noise, the SNR of the asymptotic PCA is not less than the worst SNR among the samples, but can be strictly smaller than the higher SNR among the samples.*

Proof. We have

$$\sum_{d=1}^{D} \alpha_d^2 \sigma_d^2 = \sum_{d=1}^{D} \frac{\sigma_d^2}{\alpha_d^2} \alpha_d^4 \leq \left(\max_{d=1}^{D} \frac{\sigma_d^2}{\alpha_d^2} \right) \sum_{d=1}^{D} \alpha_d^4.$$

Since $\left(\max_{d=1}^{D} \frac{\sigma_d^2}{\alpha_d^2} \right)^{-1} = \min_{d=1}^{D} \frac{\alpha_d^2}{\sigma_d^2} = \min_{d=1}^{D} \mathrm{SNR}_d$, the expression of the SNR of the asymptotic PCA given by Eqn. (2.16) is such that

$$\mathrm{SNR}_{\mathrm{PCA}} = \frac{\left(\sum_{d=1}^{D} \alpha_d^2 \right)^2}{\sum_{d=1}^{D} \alpha_d^2 \sigma_d^2} \geq \frac{\left(\sum_{d=1}^{D} \alpha_d^2 \right)^2}{\sum_{d=1}^{D} \alpha_d^4} \min_{d=1}^{D} \mathrm{SNR}_d \geq \min_{d=1}^{D} \mathrm{SNR}_d \tag{2.18}$$

where we have used Cauchy-Schwarz inequality $\sum_{d=1}^{D} \alpha_d^2 \alpha_d^2 \leq \left(\sum_{d=1}^{D} \alpha_d^2 \right)^2$.

Conversely, we can give an example for which $\mathrm{SNR}_{\mathrm{PCA}} < \max_{d=1}^{D} \frac{\alpha_d^2}{\sigma_d^2}$. Take $D = 2$, $\alpha_1 = \alpha_2 = 1$, $\sigma_1 = 1$ and $\sigma_2 = 10$. Then $\mathrm{SNR}_{\mathrm{PCA}} = 4/(1 + 10^2) = 4/101$, which is strictly smaller than $\alpha_1^2/\sigma_1^2 = 1$. □

2.4.2 Linear Discriminant Analysis (LDA)

LDA has been introduced in side-channel analysis in (158). With respect to inter-class PCA, it computes the eigenvectors of the matrix $S_w^{-1} S_b$, where:

- S_w is the *within-class scatter matrix*, asymptotically equal to Σ, and
- S_b is the *between-class scatter matrix*, equal to $\alpha^D (\alpha^D)^{\mathsf{t}}$.

We have the following spectral characterization of the asymptotic LDA:

Proposition 9. *Asymptotically, LDA has only one principal direction, namely the vector $\Sigma^{-1} \alpha^D$.*

Proof. The matrix $S_w^{-1} S_b = \Sigma^{-1} \alpha^D (\alpha^D)^{\mathsf{t}}$ has rank one. Indeed, $\alpha^D (\alpha^D)^{\mathsf{t}}$ has rank one, and multiplying by an invertible matrix (namely Σ^{-1}) keeps the rank unchanged. Since

$$(\Sigma^{-1} \alpha^D (\alpha^D)^{\mathsf{t}}) \Sigma^{-1} \alpha^D = \Sigma^{-1} \alpha^D ((\alpha^D)^{\mathsf{t}} \Sigma^{-1} \alpha^D) = \left((\alpha^D)^{\mathsf{t}} \Sigma^{-1} \alpha^D \right) \times \Sigma^{-1} \alpha^D,$$

$\Sigma^{-1}\alpha^D$ is the unique eigenvector with corresponding eigenvalue $(\alpha^D)^{\mathsf{t}}\Sigma^{-1}\alpha^D > 0$. This eigenvalue is equal to the SNR of the asymptotic LDA. □

By Corollary 2, the SNR of the asymptotic LDA is equal to the SNR of the optimal dimensionality reduction, denoted by $\widetilde{\mathrm{SNR}}$. In fact, we have the following.

Theorem 2.4.2. *The asymptotic LDA computes exactly the optimal dimensionality reduction.*

Proof. Compare Theorem 2.2.1 with Proposition 9: in both cases, the projection vector is collinear with $\Sigma^{-1}\alpha^D$. □

2.4.3 Numerical Comparison Between Asymptotic PCA and LDA

Numerical comparison between asymptotic PCA and LDA is given in Fig. 2.1(a) and (b), for $D = 6$ samples. The noise is chosen autoregressive, with $\sigma = 1$ and different values for ρ (Sec. 2.3.2). The vector α^D is chosen equal to $(1, 1, 1, 1, 1, 1)^{\mathsf{t}}$ in Fig. 2.1(a) and to $\sqrt{6.0/6.4}\cdot(1.0, 1.1, 1.2, 1.3, 0.9, 0.5)^{\mathsf{t}}$ in Fig. 2.1(b), such that $\widetilde{\mathrm{SNR}} = 6$ when $\rho = 0$. The SNR of the asymptotic LDA is that of the optimal dimensionality reduction (cf. Corollary 2), and that of the asymptotic PCA can be found in Example 3. The first case (Fig. 2.1(a)) fits the situation depicted in Corollary 3. The asymptotic PCA and LDA are almost similar. Besides, when $\rho \to 1^-$, both SNRs tend to 1 (recall Eqn. (2.17) and (2.14)). But, when the SNR varies over the D samples (Fig. 2.1(b)), the asymptotic LDA can be significantly better than the asymptotic PCA. The sample-wise extremal SNRs ($\mathrm{SNR}_d = \alpha_d^2/\sigma^2$) are also represented: the SNR of the PCA can be smaller than the largest SNR, namely $\max_{1\leq d\leq D}\mathrm{SNR}_d$, (recall Lemmas 1), which is not the case of the SNR of the LDA. Actually, the SNR of LDA increases to infinity because $\widetilde{\mathrm{SNR}} \approx 0.164/(1-\rho)$ when $\rho \to 1^-$ (see Eqn. (2.13)).

2.5 Practical Validation

In this section, we investigate real traces. Experiments are carried out on the DPA CONTEST v2 (168) traces. One clock cycle lasts $D = 200$ samples. As traces are captured from a hardware implementation of an AES, we consider the Hamming distance leakage model (in accordance with most attacks reported on the analyzed device (36), namely a SASEBO-GII board with a Xilinx XC5VLX30 FPGA (146)). In the sequel, we focus on the Hamming distance between the byte 0 of the last round and that of the cipher text. That is, the function Ψ in Eqn. (1.2) is a normalized Hamming weight; precisely, $\Psi : z \in \mathbb{F}_2^n \mapsto \frac{2}{\sqrt{n}}\left(w_H(z) - \frac{n}{2}\right)$, where $n = 8$, because

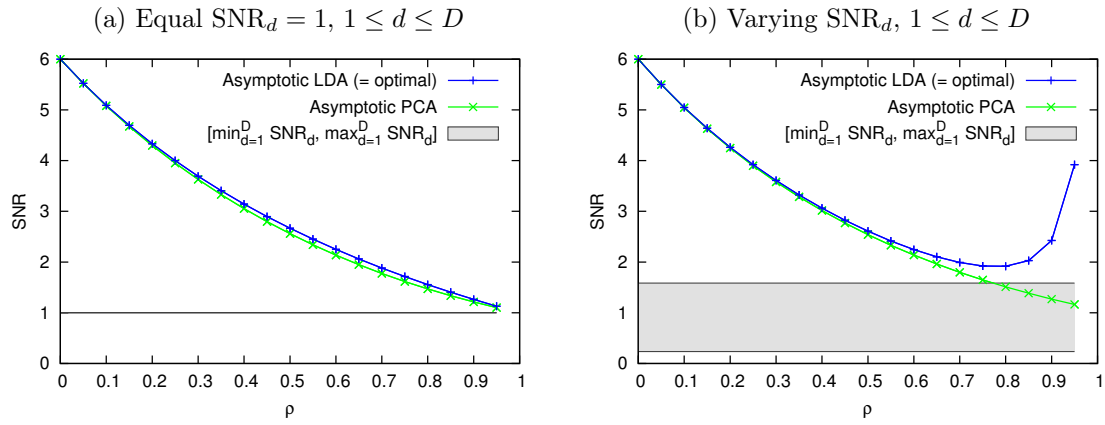


Figure 2.1: Comparison of the SNR of asymptotic LDA (optimal) and of asymptotic PCA

AES is a byte-oriented block cipher. In addition, we emphasize that our model (Eqn. (2.1)) is indeed suitable to leakage dimensionality reduction within one clock period.

2.5.1 Precharacterization of the Model Parameters α^D and Σ

In order to characterize the model, we need to recover the column matrix α^D and the $D \times D$ covariance matrix Σ of the noise.

Proposition 10. *The parameters of the model* (2.1) *which minimize the fitting error are given by*

$$\widehat{\alpha}^D = \frac{\mathbf{x}^D(\mathbf{y})^{\mathsf{t}}}{\mathbf{y}(\mathbf{y})^{\mathsf{t}}}.$$

Proof. The goal (minimizing the fitting error) is similar to that of the optimal distinguisher, namely maximize the probability of $p_{\mathbf{N}^D}(\mathbf{x}^D - \alpha^D \mathbf{y})$ (Eqn. (2.4)). But in the context of characterization, the correct key is known. Therefore, we wish to minimize in α^D and Σ the following objective function:

$$\text{objective}(\alpha^D, \Sigma) = \sum_{q=1}^{Q} \left\{ \left(x_q^D - \alpha^D y_q(k^\star)\right)^{\mathsf{t}} \Sigma^{-1} \left(x_q^D - \alpha^D y_q(k^\star)\right) \right\}, \tag{2.19}$$

which reminds of Eqn. (2.7) (except that now, the key $k = k^\star$ is known). We use the notation $(\widehat{\alpha}^D, \widehat{\Sigma}) = \text{argmin}_{(\alpha^D, \Sigma)} \text{objective}(\alpha^D, \Sigma)$.

We fix Σ and minimize only on α^D. The gradient of $\text{objective}(\alpha^D, \Sigma)$ w.r.t. $(\alpha^D)^{\mathsf{t}}$ writes:

$$\frac{\partial}{\partial (\alpha^D)^{\mathsf{t}}} \text{objective}(\alpha^D, \Sigma) = \sum_{q=1}^{Q} -2 y_q(k^\star) \left(\Sigma^{-1} x_q^D - y_q(k^\star) \Sigma^{-1} \alpha^D\right). \tag{2.20}$$

The objective function is extremal in $\widehat{\alpha}^D$ if and only if its derivative is equal to zero at this point. Let $\mathbf{y}$ be an abbreviation for $\mathbf{y}(k^\star)$. This condition takes the form of a *normal equation*

$$\widehat{\alpha}^D = \frac{\sum_{q=1}^{Q} y_q x_q^D}{\sum_{q=1}^{Q} y_q^2} = \frac{\mathbf{x}^D(\mathbf{y})^\mathsf{t}}{Y^Q(\mathbf{y})^\mathsf{t}}. \tag{2.21}$$

where the numerator is the inter-covariance matrix of $\mathbf{x}^D$ and $\mathbf{y}$, and the denominator is the covariance matrix of $\mathbf{y}$. □

Interestingly, the most likely value $\widehat{\alpha}^D$ of α^D does not depend on the noise covariance matrix. As $\mathbf{N}^D = \mathbf{x}^D - \widehat{\alpha}^D\mathbf{y}$ has zero mean, the latter can be evaluated on its own as the well-known unbiased estimator of Σ:

$$\widehat{\Sigma} = \tfrac{1}{Q-1}(\mathbf{x}^D - \widehat{\alpha}^D\mathbf{y})(\mathbf{x}^D - \widehat{\alpha}^D\mathbf{y})^\mathsf{t}. \tag{2.22}$$

By plugging Eqn. (2.21) into Eqn. (2.22), one obtains

$$\begin{aligned}
\widehat{\Sigma} &= \tfrac{1}{Q-1}\left(\mathbf{x}^D - \mathbf{x}^D\frac{(\mathbf{y})^\mathsf{t}\mathbf{y}}{Y^Q(\mathbf{y})^\mathsf{t}}\right)\left(\mathbf{x}^D - \mathbf{x}^D\frac{(\mathbf{y})^\mathsf{t}\mathbf{y}}{Y^Q(\mathbf{y})^\mathsf{t}}\right)^\mathsf{t} \\
&= \tfrac{1}{Q-1}\mathbf{x}^D\left(I^{Q,Q} - \frac{(\mathbf{y})^\mathsf{t}\mathbf{y}}{Y^Q(\mathbf{y})^\mathsf{t}}\right)^2(\mathbf{x}^D)^\mathsf{t} &(2.23)\\
&= \tfrac{1}{Q-1}\mathbf{x}^D\left(I^{Q,Q} - \frac{(\mathbf{y})^\mathsf{t}\mathbf{y}}{Y^Q(\mathbf{y})^\mathsf{t}}\right)(\mathbf{x}^D)^\mathsf{t} &(2.24)\\
&= \tfrac{1}{Q-1}\left(\mathbf{x}^D(\mathbf{x}^D)^\mathsf{t} - \frac{\mathbf{x}^D(\mathbf{y})^\mathsf{t}\mathbf{y}(\mathbf{x}^D)^\mathsf{t}}{Y^Q(\mathbf{y})^\mathsf{t}}\right).
\end{aligned}$$

In Eqn. (2.23), $I^{Q,Q}$ denotes the $Q \times Q$ identity matrix, and we use in Eqn. 2.24 the fact that the matrix $I^{Q,Q} - (\mathbf{y})^\mathsf{t}\mathbf{y}/(Y^Q(\mathbf{y})^\mathsf{t})$ is idempotent, i.e., equal to its square.

Remark 9. *We have the following remarkable identity:*

$$\mathbf{x}^D(\mathbf{x}^D)^\mathsf{t} \quad = \quad \widehat{\alpha}^D(\widehat{\alpha}^D)^\mathsf{t}\,\mathbf{y}(\mathbf{y})^\mathsf{t} \quad + \quad (Q-1)\widehat{\Sigma}.$$

This equation is the non-asymptotic version of Proposition 6.

2.5.2 Computation of SNRs on the AES Traces from DPA Contest v2 Last Round

The values $\widehat{\alpha}^D$ and $\widehat{\Sigma}$ are represented in Fig. 2.2. We obtain:

- $\max_{d=1}^{D} \widehat{\alpha}_d^2/\widehat{\Sigma}_{d,d} = 1.69 \cdot 10^{-3}$ (no dimensionality reduction)
- $\mathrm{SNR}_{\mathrm{PCA}} = \frac{((\widehat{\alpha}^D)^\mathsf{t}\widehat{\alpha}^D)^2}{(\widehat{\alpha}^D)^\mathsf{t}\widehat{\Sigma}\widehat{\alpha}^D} = 1.36 \cdot 10^{-3}$ (PCA)

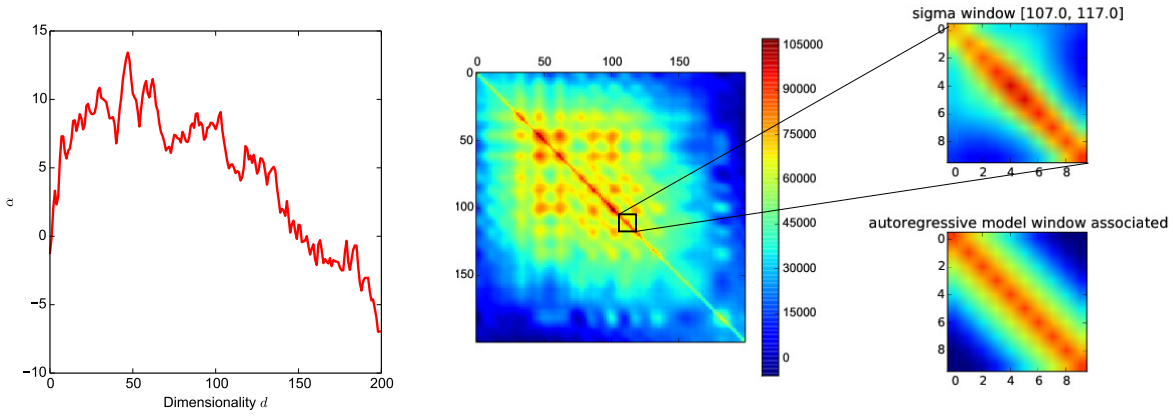


Figure 2.2: Estimated $\widehat{\alpha}^D$ (*left*) and $\widehat{\Sigma}$ (*right*), for $Q = 10,000$ traces

- $\mathrm{SNR}_{\mathrm{LDA}} = (\widehat{\alpha}^D)^{\mathrm{t}}\widehat{\Sigma}\widehat{\alpha}^D = 12.78 \cdot 10^{-3}$ (LDA)

Therefore, the LDA has the largest SNR: it is about seven times larger than the maximum sample-wise SNR. The PCA has, in this example, an SNR smaller than the maximum univariable SNR (see Lemma 1).

Interestingly, one can see in Fig. 2.2 that the noise is locally autoregressive, for instance between samples 107 and 117.

2.6 Conclusions and Perspectives

Dimensionality reduction is common practice in side-channel analysis. This pre-processing technique has many virtues, such as an elegant multivariate description of the leakages, the concentration of information which reduces the required number of measurements to extract the key, and the increase of computational efficiency. Nonetheless, as any processing, dimensionality reduction can only reduce some information.

Using a mathematical approach, we have shown that dimensionality reduction is actually part of the optimal attack. This proves rigorously that dimensionality reduction can be achieved without loss in terms of attack success probability in extracting a secret key. As it turns out, the optimal dimensionality reduction consists in a linear projection of the trace samples.

We have also shown that the linear discriminant analysis asymptotically achieves the same projection, and hence becomes optimal as the number of traces increases. When the various samples are weakly correlated, we have found that PCA is nearly equivalent to the optimal

dimensionality reduction and to LDA. Thus, in realistic contexts, state-of-the-art dimensionality reduction techniques are actually close to the optimal method.

Finally, we show how to estimate the model parameters (modulation vector α^D and noise covariance matrix Σ), and compute them on a real traces. An SNR gain factor of 7 can be obtained with respect to sample-wise SNR, which stresses the practical interest of dimensionality reduction.

As a perspective, we note that it should also be possible to obtain similar results when the noise is non-Gaussian (e.g., uniform). It is also desirable to compare dimensionality reduction based on linear projections to machine-learning techniques which are also multidimensional, such as SVM, random forests, K-means, etc.

CHAPTER 3

Dimensionality Reduction a case study in presence of masking.

The results presented in this chapter have been published in collaboration with Jean-Luc Danger, Sylvain Guilley, Annelie Heuser, and Yannick Teglia in the international conference on Security, Privacy and Applied Cryptography Engineering (SPACE 2014) (20).

Contents

Multi-variate side-channel attacks allow to break higher-order masking protections by combining several leakage samples. But how to optimally extract all the information contained in all possible Ω-tuples of points? In this chapter, we introduce preprocessing tools that answer this question. We first show that maximizing the higher-order CPA coefficient is equivalent to finding the maximum of the covariance. We apply this equivalence to the problem of trace dimensionality reduction by linear combination of its samples. Then we establish the link between this problem and the Principal Component Analysis. In a second step we present the optimal solution for the problem of maximizing the covariance. We also theoretically and empirically compare these methods. We finally apply them on real measurements, publicly

available under the DPA Contest v4, to evaluate how the proposed techniques improve the second-order CPA (2O-CPA).

3.1 Introduction

In some particular masking implementations, the two shares (117) depending on the same mask leak at different time samples (e.g., in software). Second-order attacks that combine two different time samples are called *bi-variate* SCA. When the masking scheme uses Ω shares, *multi-variate* SCA are still able to reveal the secret key by combining leakage samples corresponding to each of the Ω shares. Note that, depending on the implementation and the measurement setup each share may leak in multiple samples.

To enhance the results of SCA several preprocessing tools can be used. In the case of *bi-variate* SCA it is particularly interesting to take into account all the information spread over the time. Indeed, the number of possible pairs increases quadratically in the number of leakage samples. For example, if the first share leaks over D_1 samples and the second share over D_2 samples, we could perform a *bi-variate* SCA on $D_1 \times D_2$ possible combined points. So, taking into account all these leaks may undoubtedly increase the efficiency of an attack.

More generally, to break $(\Omega - 1)$-order masking schemes the attacker needs to combine Ω samples corresponding to Ω shares. So, if D_i is the number of samples which leak the i-th share then the attacker could perform *multi-variate* SCA on $\prod_{1 \leq i \leq d} D_i$ different Ω-tuples. In other words, the number of possible Ω-tuples to perform *multi-variate* SCA is in $O(D^{\Omega})$ where D is the number of samples each share leaks (and assuming that each share is leaking the same number of samples, i.e., $\forall i \in [\![1, \Omega]\!], D_i = D$).

In order to break $(\Omega - 1)$-order masking schemes an attacker should therefor combine first the leakages of each share, this leads to an increase of the number of exploitable samples. In a second times to increase the success of its attack an attacker can exploit all theses leakages in one by combining them. It will be the case when dimensionality reduction methods are applied.

Many methods have been presented in the area of SCA to combine the information spread over time: the Principal Component Analysis (PCA) for dimensionality reduction (2) for Template attacks (31) (see Chapt. 2 for an analysis of dimensionality reduction in a Template Attacks scenario) but also as a preprocessing tool (5) for DPA (87). Recently in (70) Hajra and Mukhopadhyay present an approach based on match filters to find the optimal preprocessing. Other methods have been designed to combine samples from different acquisitions ((154, 158)).

Additionally, PCA has also been used as a distinguisher in (157). Some other methods could be applied like the Canonical Correlation Analysis (120) to improve CPA (19). Interestingly, all these methods lead to a dimensionality reduction.

Another approach to improve the efficiency of SCA is to find the optimal model. A *linear-regression* approach may be used. In (120) Oswald and Paar introduce optimization algorithms to determine numerically the optimal linear combination before CPA. By choosing a different objective we can give a formal expression for the result of the optimization problem, and then have an optimal method without any utilization of sophisticated optimization algorithms that would require "parameter settings", which could be costly in time. Still, we notice that the approach in (120) and our could be advantageously combined.

Contributions.

In this chapter we tackle the question *how to optimally combine the information spread over multiple time samples, for* HOCPA *attacks of arbitrary order?* Namely we extend the results of Chapt. 2 in a context of less powerful attacker and attacks against protected implementations. Specifically in this chapter we assume that the attacker is not able to completely profiled the leakage function. We present the optimal preprocessing method and express it as a generic synthetic formula. By linking the PCA to the problem of maximizing the result of the CPA we are able to evaluate the presented method. We compare these two methods theoretically and prove that they are optimal under some assumptions. We then compare these methods empirically as preprocessing tools to boost 2O-CPA attacks on a first-order masking scheme. In particular, we test these methods on real measurements (DPA contest v4 (169)). In summary, we show that taking into account all possible pairs of leakage points will significantly improve the effectiveness of 2O-CPA, in one attack.

Outline of the chapter.

The rest of the chapter is organized as follows. In Sect. 3.2 we present our case study and a theoretical comparison between PCA and the covariance method as a method to obtain the optimal preprocessing for second-order CPA. The attacks are then applied on a real masked implementation in Sect. 3.3. Sect. 3.4 provides another case study to apply these methods as preprocessing tools. Finally, conclusions and perspectives are drawn in Sect. 3.5.

3.2 Theoretical optimal preprocessing function

3.2.1 Case study

Let us assume that each measurement trace can be seen as a vector of points. So the leakage of the measurements can be defined as: $X = \left(X^{(d)}\right)_{1\leq d\leq D}$ where $X^{(d)} = S^{(d)} + N^{(d)}$, $S^{(d)} = \Psi^{(d)}\left(g\left(k^\star, T, R\right)\right)$ being the part of the leakage which is linked to the internal operation processed on the target component and $N^{(d)}$ being the noise that assumed to be independent of $S^{(d)}$. It can be noted that, we simply refer to interval $[\![1, D]\!]$ as D, whenever there is no risk of confusion. It can also be assumed that these traces are centered and also reduced, i.e., $\mathbb{E}\left[X^{(d)}\right] = 0\ \forall d$ and $\mathrm{Var}\left[X^{(d)}\right] = 1\ \forall d$. Note that, the attacker is always able to center by removing the empirical mean and reduce by dividing the empirical standard deviation.

Let Z be the internal variable (depending on the sensitive variable) manipulated during the algorithm and let $\widehat{\Psi}$ defines the leakage model. In the case of CPA, a transformation of the initial data (preprocessing) may increase the correlation coefficient. To consider all information contained in X an option would be to use a linear transformation as a prepossessing. Note that, combining all points by a weighted sum leads to a dimensionality reduction. More precisely,

$$\max_{\alpha} |\rho\left[\alpha \cdot X, \widehat{\Psi}(Z)\right]|, \tag{3.1}$$

where ρ is the Pearson coefficient, α is a vector in $\mathbb{R}^D$ and $\cdot$ the scalar product.

Remark 10. *The solution of* $\max_{\alpha} |\rho\left[\alpha \cdot X, \widehat{\Psi}(Z)\right]|$ *is also a solution of* $\max_{\alpha} \rho\left[\alpha \cdot X, \widehat{\Psi}(Z)\right]^2$.

Remark 11 (EIS (Equal Images under the Same key) assumption (147))**.** *The only part of the correlation that allows to distinguish the key is the covariance.*

After the preprocessing we do not need to normalize by the variance of the traces, because we compare key guesses between each other for a given time sample not on a direct scale. So, as seen in Remark 11 the normalization by the variance does not impact the way we distinguish the key. Thus, we can simply focus on maximizing the following equation:

$$\max_{\|\alpha\|=1} \mathsf{Cov}\left[\alpha \cdot X; \widehat{\Psi}(Z)\right]^2. \tag{3.2}$$

As the covariance is not bounded we introduce the constraint $\|\alpha\| = 1$ where $\|\cdot\|$ is the Euclidean norm, namely $\|\alpha\| = \sqrt{\alpha \cdot \alpha}$.

In this section we assume that the attacker has a “learning device" with a fixed key on which he is unrestricted in the number of acquisitions, which is typically more than the required

number to successfully perform the attack. As a consequence we can reasonably assume that the attacker knows the key on the learning device and he is able to identify the zones of interest in $[\![1, D]\!]$ where the internal variable leaks. Moreover, he is able to estimate the weights of the linear combination (see Eq. (3.2)) on the learning device. In the rest of this study we call this step the "learning phase". In the final step the attacker targets another device that is expected to leak in a similar way as the learning one. However, on the device under attack he is no longer able to acquire an unlimited amount of traces. In particular, in this "attack phase" his main goal is to retrieve the secret key using only the minimum number of traces.

3.2.2 Principal component analysis

A classical way to recombine information with linear combinations is to apply PCA (79).

Remark 12. *In general, most of the variance lays within a few dimensions (i.e., much less than D).*

Proposition 11. *The solution of the problem in Def. 18 is the D' eigenvectors of X associated to the D' maximal eigenvalues.*

Proof. The proof can be found in (79). □

As the problem of maximizing the covariance depends on the expected leakage model the preprocessing is defined such that it takes $\widehat{\Psi}(Z)$ into account. This implies that the given preprocessing methods are model-dependent. We can explicit the Proposition 11:

Proposition 12. *If we link our measurements X to their conditional expectations $\mathbb{E}\left[X|\widehat{\Psi}(Z)\right]$ knowing a model $\widehat{\Psi}(Z)$, then the* PCA *yields the principal direction:*

$$\max_{\|\alpha\|=1} Var\left[\alpha \cdot \mathbb{E}\left[X|\widehat{\Psi}(Z)\right]\right].$$

This result means that the eigenvector of the largest eigenvalue is the projection that maximizes the inter-class variance.

Proof. Let $\widehat{\Psi}_1, \widehat{\Psi}_2, \ldots, \widehat{\Psi}_N$ the values that $\widehat{\Psi}(Z)$ can take. Then, the lines of matrix X are $\mathbb{E}\left[X|\widehat{\Psi}(Z) = \widehat{\Psi}_1\right]$, $\mathbb{E}\left[X|\widehat{\Psi}(Z) = \widehat{\Psi}_2\right]$, …, $\mathbb{E}\left[X|\widehat{\Psi}(Z) = \widehat{\Psi}_N\right]$. Apply Proposition 11. □

3.2.3 Preprocessing on modulated side channel traces

In this section we investigate the behaviors of preprocessing methods on modulated side channel traces. Let us first recall the definition of modulated traces given in Def. 9:

Definition 38 (Modulated Traces)**.** *Let us now define a* modulated trace *as a trace in which each time sample can be expressed as a modulation of a model (static in time) plus an independent noisy part:*

$$X = \left(\beta^{(d)}\widehat{\Psi}(Z) + N^{(d)}\right)_{d\leq D} = \beta \cdot \widehat{\Psi}(Z) + \left(N^{(d)}\right)_{d\leq D}, \tag{3.3}$$

where β is a vector in $\mathbb{R}^D$ and each $N^{(d)}$ is drawn from an independent identical distribution $\mathcal{N}$. In specific, the variance of the noise does not depend on the time sample $d \leq D$.

Theorem 3.2.1. *In the case of modulated traces the solution of* PCA *is equivalent to maximizing the covariance (Eqn.* (3.2)*). More precisely, if $X = \left(\beta^{(d)}\widehat{\Psi}(Z) + N^{(d)}\right)_{d\in D}$ then*

$$\alpha \in \underset{\|\alpha\|=1}{\operatorname{argmax}} \mathsf{Cov}\left[\alpha \cdot X; \widehat{\Psi}(Z)\right]^2 \iff \alpha \in \underset{\|\alpha\|=1}{\operatorname{argmax}} Var\left[\mathbb{E}\left[\alpha \cdot X | \widehat{\Psi}(Z)\right]\right].$$

Proof. The proof is given in Appendix A.1. □

Notice that, in Theorem 3.2.1, we consider that many vectors α can maximize the covariance: so, the return value of the argmax operator is a set.

In a particular case of Theorem 3.2.1 we can explicitly describe α.

Lemma 2. *If α and β are linearly dependent, we have:*

$$\frac{\beta}{\|\beta\|} \in \underset{\|\alpha\|=1}{\operatorname{argmax}} \mathsf{Cov}\left[\alpha \cdot X; \widehat{\Psi}(Z)\right]^2 . \tag{3.4}$$

Proof. The proof is given in Appendix A.2. □

After projection into the new reduced space the covariance matrix will be zero everywhere except at $(0, 0)$. Moreover, all the variance should be contained in the first principal direction, thus, we do not need to take the other eigenvectors into consideration.

As β does not depend on a particular model we also maximize the covariance for wrong keys in the same proportion as the covariance for the good key. Thus we do not change the way we distinguish the good key from the wrong ones (the relative distinguishing margin is unchanged (181)). However, the dimensionality reduction leads to an improvement of the attack by increasing the signal-to-noise ratio (SNR). We define the SNR as the variance of the signal divided by the variance of the noise. This definition of SNR coincides with the Normalized Inter-Class Variance (NICV (12, 13)).

Lemma 3. *If the noise $N^{(d)}$ is identically distributed (i.d.) for all d, then the noise is unchanged by any linear combination of unitary norm.*

Proof. By hypothesis, $\mathrm{Var}\left[\alpha \cdot \left(N^{(d)}\right)_{d\in D}\right] = \|\alpha\|^2 \mathrm{Var}\left[\mathcal{N}\right] = \mathrm{Var}\left[\mathcal{N}\right]$. □

Remark 13. *In Lemma 3 the traces are modulated (see Def. 9) as a consequence the noise is independent in the time samples. The assumption of modulated traces is a strong assumption, nevertheless the noise can always be standardized by multiplying the traces with the inverse of the noise covariance matrix.*

Proposition 13. *If the noise $N^{(d)}$ is i.d. for all d, then the signal-to-noise ratio is maximum after the projection:*

$$\frac{\max\limits_{d\in D} Var\left[\beta^{(d)}\widehat{\Psi}(Z)\right]}{Var[\mathcal{N}]} \leq \frac{\max\limits_{\|\alpha\|=1} Var\left[\alpha\cdot\mathbb{E}\left[X|\widehat{\Psi}(Z)\right]\right]}{Var[\mathcal{N}]}.$$

Proof. By definition of α we have $\max\limits_{d\in D}\mathrm{Var}\left[\beta^{(d)}f(Z)\right] \leq \max\limits_{\|\alpha\|=1}\mathrm{Var}\left[\alpha\cdot\mathbb{E}\left[X|\widehat{\Psi}(Z)\right]\right]$. Besides, by lemma 3, the numerator of the SNR does not depend on our preprocessing, since is satisfies $\|\alpha\| = 1$. □

Remark 14. *In the case of modulated traces the* PCA *gives the solution of a matched-filter (109).*

3.2.4 Covariance vector as a preprocessing method

In the general case when the model is not known or in the presence of noise, the variance may not only be contained in the first eigenvector (5). Therefore, it may be useful to also take the other directions of the PCA into account. Note that, we still obtain an optimal function to reduce the dimensionality before conducting a CPA under the same leakage model assumption.

Proposition 14 (Covariance method)**.**

$$\left(\frac{\mathsf{Cov}\left[X^{(d)};\widehat{\Psi}(Z)\right]}{\|\left(\mathsf{Cov}\left[X^{(d)};\widehat{\Psi}(Z)\right]\right)_{d\in D}\|}\right)_{d\in D} \in \underset{\|\alpha\|=1}{\operatorname{argmax}}\,\mathsf{Cov}\left[\alpha\cdot X;\widehat{\Psi}(Z)\right]^2$$

Proof. The proof is given in Appendix A.3. □

So, the normalized covariance is the optimal preprocessing method to maximize the value of the covariance when using linear combinations of traces points. In the rest of this study we call this method the “covariance method" and the result the “covariance vector".

Remark 15. *Note that, the model of the actual leakage of the traces is not used in the proof of Appendix A.3. The results are therefore applicable for any leakage model such as the one presented in (67).*

3.2.5 Discussion

The previous subsection shows that the projection of the differential traces on the covariance vector gives a solution to the problem of maximizing the covariance after dimensionality reduction (i.e., after having learned the best linear form). This method is better than the state-of-the-art, where each tuple of samples is processed on its own (see the *big picture* in Fig. 3.1); it can be seen as a generalization to higher-order attacks of (70). Some other preprocessing tools have been proposed to reduce the dimensionality and enhance the quality of the CPA. The PCA (5) is a known way to preprocess the data to reduce the dimension and increase the efficiency of attacks. As defined in Sect. 3.2.3, PCA is directly linked to the maximization problem, which is also underlined by our empirical results given in Sect. 3.3.

Oswald and Paar showed in (120) that the best linear combination ("best" in the sense of separating the highest peaks from the nearest rival) can be approached by numerical resolutions. The model presented in (70) is not totally applicable to our study case. If we are in the case of modulated traces, the expectation over each sample of the combined traces could be null. In this case the method presented is not directly suitable.

The point of this study is not to exhibit a better method for dimensionality reduction but to show that we can solve this problem in an easier way by using the vector of covariance.

Other preprocessing methods can be used before any dimensionality reduction such as reduction filtering using a Fourier or a Hartley transform (8). However, when the transformation is linear and invertible, the covariance method applies in a strictly equivalent way. The next subsection clarifies this point on the example of the Fourier transform.

3.2.6 Time vs Frequency domains

Let X a signal in time domain, i.e., $X = (X^{(d)})_{d \in D}$. The representation of X in the frequency domain is the discrete Fourier transform $\mathcal{F}(X)$.

Definition 39 (Discrete Fourier transform)**.** *Let ß be a square root of -1 in $\mathbb{C}$. The discrete Fourier transform of X is a vector $\mathcal{F}(X)$ of same length, defined as $\mathcal{F}(X)_f = \sum_{d \in [\![1,D]\!]} X^{(d)} \cdot e^{-2\pi ß f d/D}$, for all f in the interval $[\![1, F]\!]$ (where $F = D$).*

Proposition 14 can also be applied on $\mathcal{F}(X)$ instead of X. We then have the following Corollary.

Corollary 5 (Covariance method in the frequency domain)**.** *The covariance method in frequency domain yields covariance vectors equal to the Fourier transform of the covariance vectors in the time domain.*

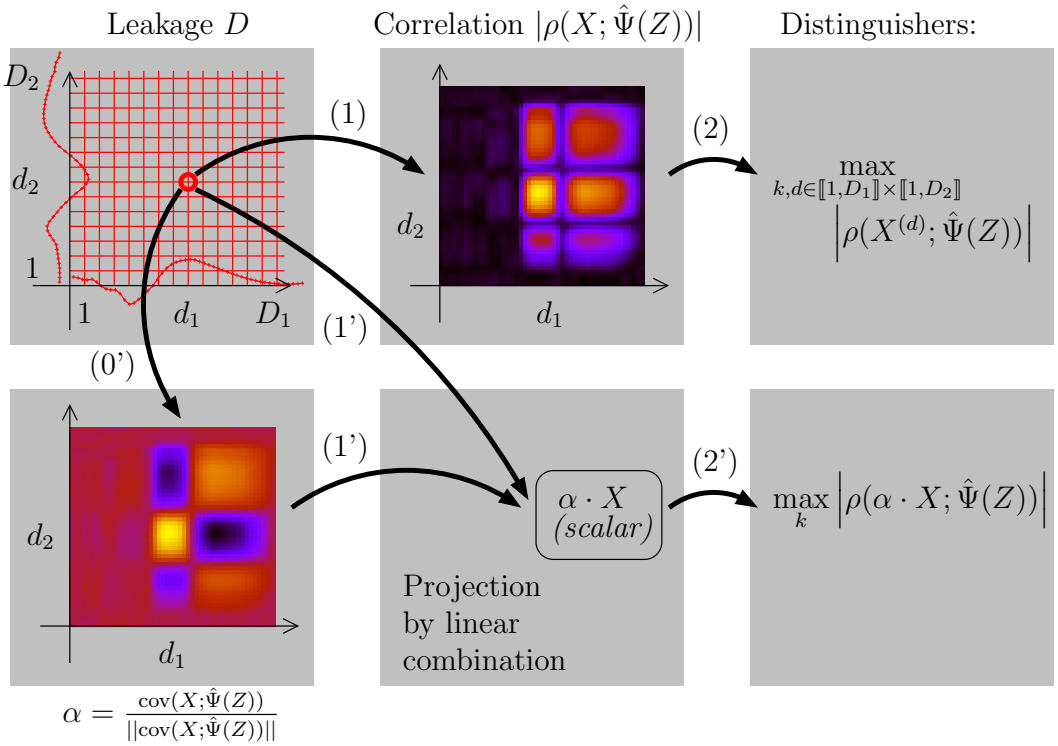


Figure 3.1: Big picture of the "covariance method". The usual 2O-CPA computes a correlation for each pair (d_1, s_2) of leakage (step (1)), and then searches for a maximum over the keys and the time instances (step (2)). Our new method obtains a "covariance vector" (termed α) on a "learning device" (step (0')), and then first projects the leakage X on α (step (1')), before looking for the best key only while maximizing the distinguisher (step (2')). Notice that the model $\widehat{\Psi}(Z)$ depends implicitly on the key guess k.

Proof. We have $\alpha \cdot \mathcal{F}(X) = \mathcal{F}(\alpha) \cdot X$, by interversion of the sums on f and d. Besides, Parseval's theorem states that $||\mathcal{F}(\alpha)||^2 = ||\alpha||^2$. Thus, the application of Proposition 14 on $\mathcal{F}(X)$ instead of X yields $\mathcal{F}(\alpha)$, where α are the covariance vectors obtained in the time domain. □

3.3 Empirical results

In Sect. 3.2 we defined two preprocessing methods (the PCA and the "covariance method"). They were described in general, but can also apply to second-order CPA; the only difference is that the interval $[\![1, D]\!]$ where samples live is replaced by the Cartesian product $[\![1, D_1]\!] \times [\![1, D_2]\!]$, where D_1 and D_2 are the window lengths containing the leakage of the two shares. Accordingly, the leakage X is the suitable combination (e.g., the centered product (133)) of samples from each window, which is reflected in the model (See for instance Eqn. (3.5) and (3.6)). We

will now compare these two methods on real measurements. These methods combine in one point the information spread over several points. The more samples to combine, the more the dimensionality reduction increases the success of the attacks.

3.3.1 Implementation of the masking scheme

To evaluate these two methods we use the publicly available traces of the DPA contest v4 (169), which uses a first order low-entropy masking protection applied on AES called Rotating S-box Masking (RSM). In RSM only sixteen Substitution boxes (S-boxes) are used and all the sixteen outputs of SubBytes are masked by a different mask. We take great in this chapter to exploit second-order leakage (in particular, we avoid the first-order leakage identified by Moradi et al. (112)). Moreover, the same mask is used for the AddRoundKey operation where it is XORed to one plain-text byte T and in the SubBytes operation where it is XORed with the S-box output depending on another plain-text byte T'. As a consequence a bi-variate CPA can be built by combining these two leaks knowing T and T'. The leakage model in this case is given by:

$$\widehat{\Psi}(Z) = \mathbb{E}\left[(\mathrm{HW}[T \oplus M] - 4) \cdot (\mathrm{HW}[\mathtt{Sbox}[T' \oplus K] \oplus M] - 4) \,|T, T', K\right], \tag{3.5}$$

where T, T', K are two bytes of the plaintext and a byte of the key respectively, together noted $Z = (T, T', K)$, and where HW is the Hamming weight function and the expectation is taken over M. We denote this combination as (XOR, S-Boxes).

Moreover, we also define another way to combine points in order to compensate the mask. As only sixteen different masks in RSM are used, also a link between the masks used at the output of the S-boxes exists. Accordingly, the combination of the outputs of two different S-boxes are not well balanced and we could consider an attack depending on two different S-Boxes which use two different masks. In this case the leakage model for the bi-variate CPA is:

$$\widehat{\Psi}(Z) = \mathbb{E}\left[(\mathrm{HW}[\mathtt{Sbox}[T \oplus K] \oplus M] - 4) \cdot (\mathrm{HW}[\mathtt{Sbox}[T' \oplus K'] \oplus M'] - 4) \,|T, T', K, K'\right]. \tag{3.6}$$

In this equation, which we denote as (S-Boxes, S-Boxes), T and K (resp. T' and K') are the plain-text and key bytes entering the first (resp. the second) S-Box, and Z is a shortcut for the quadruple (T, T', K, K'). We notice that there exists a deterministic link between M and M'; M and M' belong to some subset $\{m_0, m_1, \ldots, m_{15}\}$ of $\mathbb{F}_2^8$. We assume that M enters S-box $0 \leq i \leq 15$ and M' S-box $0 \leq i' \leq 15$. Then when $M = m_{\mathsf{offset}}$ for some $0 \leq \mathsf{offset} \leq 15$, we have that $M' = m_{\mathsf{offset}+i'-i \mod 16}$.

3.3.2 Leakage analysis

We assume that the adversary is able to identify the area where the two operations leak during the “learning phase". In order to analyze the leakage of the two operations, we first calculate the covariance of the traces when the mask is known using 25000 measurements.

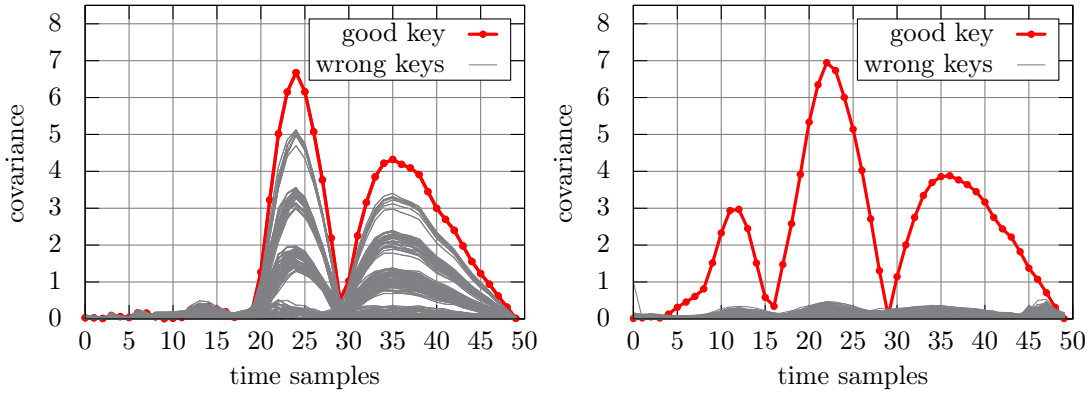
Figure 3.2a presents the absolute value of the covariance between the points where the mask is XORed with the plain-text and the leakage model $\mathrm{HW}[T \oplus M \oplus K] - 4$. The covariance is computed for all key guesses K, where the wrong keys are plotted in gray and the correct key in red. Note that, as we target a XOR operation the maximum of the absolute value of the covariance is reached for two key guesses, namely the correct one and its opposite. It is quite clear, in Fig. 3.2a, that the traces are reasonably modulated (as per Def. 9); consequently, the relative distinguishing margin is constant over all the whole trace (as underlined in Sec. 3.2.3). In the sequel, we use as leakage for the first share $\mathrm{HW}[T \oplus M] - 4$ instead of $\mathrm{HW}[T \oplus M \oplus K] - 4$. As the second share is key-dependent, this choice allows us to restrict ourselves to one key search instead of two.

Figure 3.2b presents the covariance between the points where the output of an S-box leaks and the leakage model $\mathrm{HW}[\mathtt{Sbox}[T' \oplus K] \oplus M] - 4$.

In both cases the mask leaks over several points; 50 samples represent less than 1 clock cycle. In this case the leakage is not uniformly spread over the points, thus it is reasonable to use a weighted sum to reduce the dimensionality of the data.

As the two leakages do not depend on the same operations their shapes are different. Interestingly, the distance between the correct key (red) and the next rival (grey) is much smaller in Figure 3.2a than in Figure 3.2b, Indeed the covariance plotted in Figure 3.2a is computed using a leakage depending on AddRoundKey, whereas the covariance plotted in Figure 3.2b is computed using a leakage caused by SubBytes. More precisely, the second plot corresponds to a time window when the value of the S-box output is moved during the ShiftRows operation that follows SubBytes.

Figure 3.3a (resp. 3.3b) presents the covariance between the points where the output of an S-box leaks and the leakage model $\mathrm{HW}[\mathtt{Sbox}[T \oplus K] \oplus M] - 4$ (resp. $\mathrm{HW}[\mathtt{Sbox}[T' \oplus K'] \oplus M'] - 4$). It can be noticed that the leakages of two different S-boxes indeed differ. The reason of this difference is that the two leakages are not due to the execution of the same operations. Figure 3.3b shows the covariance between the leakage of the S-box output due to the ShiftRows operation that follows and the corresponding model, whereas Figure 3.3a shows the covariance between



(a) Leakage caused by AddRoundKey

(b) Leakage caused by ShiftRows

Figure 3.2: Covariance absolute value, for (a) XOR and (b) S-box

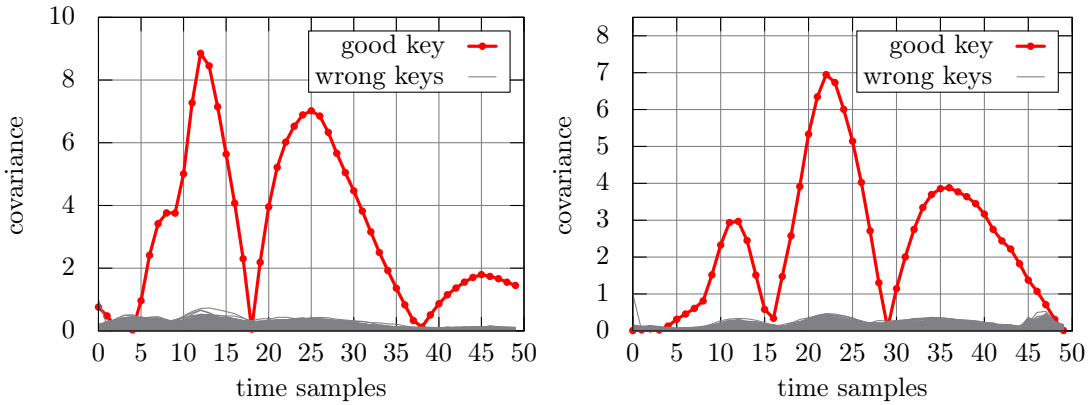
the leakage due to the SubBytes operations and the corresponding model. As *looking-up* and *moving* a byte are different operations, they leak differently.

3.3.3 Experimental protocol

In this experiment we select two windows of 50 points corresponding to the leakage of the two shares. Then all possible pairs of points have been combined using the centered product function (133). In all the experiments, the preprocessing method and the 2O-CPA are applied on these “combined" traces. We compare 2O-CPA with and without preprocessing.

We used the 50000 first traces of the DPA contest v4 for the learning phase and the remaining for the attack phase. To compute the success rate we repeated the experiment as many times as we could due to the restricted amount of traces.

Note that, several attacks using profiling or semi-profiling have been published in the Hall of Fame of the DPA contest v4. Most of these attacks are specially adapted to the vulnerabilities of the provided implementation or the particularities of RSM. However, our proposed preprocessing tools do not particularly target RSM, moreover, they are generic and could be applied to any masking scheme leaking two shares.



(a) Leakage caused by SubBytes

(b) Leakage caused by ShiftRows

Figure 3.3: Covariance absolute value, for (a) S-box and (b) S-box+ShiftRows

3.3.4 Comparison of the two preprocessing methods and classical second-order CPA

First of all, for the (XOR, S-Boxes) combination we see in Fig. 3.4 that the preprocessing improves the efficiency of the attacks. We need less than 200 measurements to reach 80% of success with the covariance or PCA preprocessing while we need more than 275 measurements for the classical 2O-CPA, which gives an improvement of 30%.

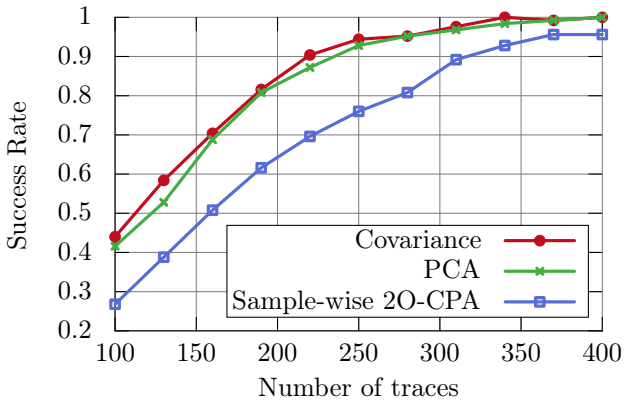
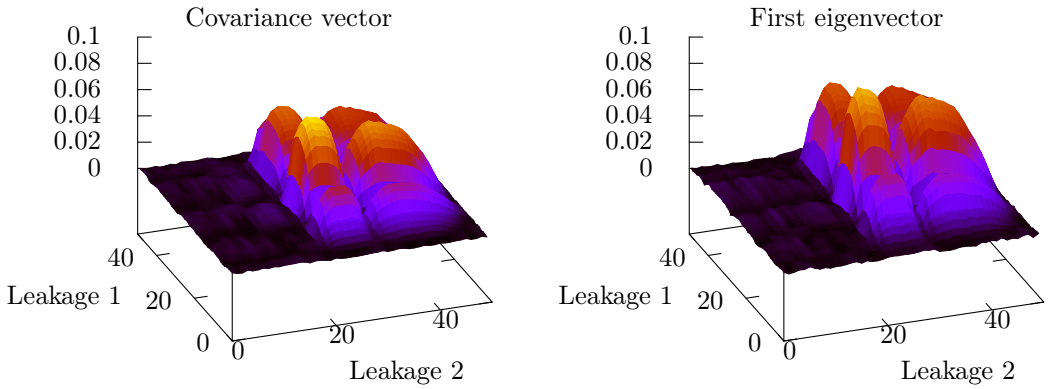


Figure 3.4: Comparison between the classical second-order CPA and second-order CPA with preprocessing using (XOR, S-Boxes)

Figure 3.5 shows a 3-D representations of the vectors using the PCA (which returns the



(a) 2-dimensional covariance vector

(b) 2-dimensional first eigenvector

Figure 3.5: Comparison between the covariance vector and the first eigenvector

first eigenvector) and the covariance method (which returns the covariance vector). The larger the value on the z-axis of Fig. 3.5 and 3.7, the higher the contribution (weight) of this point. The axes “leakage 1" and “leakage 2" represent the part depending on the two leakages of XOR (Fig. 3.2a) and S-box (Fig. 3.2b) operations in the combined traces. We can see in Figure 3.5 that the two methods highlight the same points of the combined traces and have the same shape (approximately the same values). Thus, the two methods give similar results in terms of success rate, which is confirmed by Figure 3.4.

As it can be seen in Figure 3.6, in case of the (S-Boxes, S-Boxes) combination we need around 275 traces to reach 80% of success for the 2O-CPA after the two preprocessing methods, while the raw 2O-CPA needs around 550 traces to succeed. So, using the preprocessing method decreases the number traces to perform the attack by 50%. It can be seen that the two methods yield apparently exactly the same results, which means that we are precisely in the framework of Theorem 3.2.1: the display traces that are almost perfectly modulated by one static leakage model.

One explanation of the effectiveness of the preprocessing can be found in Figure 3.7. There are much more leaking points in the same window size when we combine two S-boxes. It can be seen in Sect. 3.3.5 that another explanation can be the fact that when we apply these preprocessing methods the attacks are less sensitive to the noise.

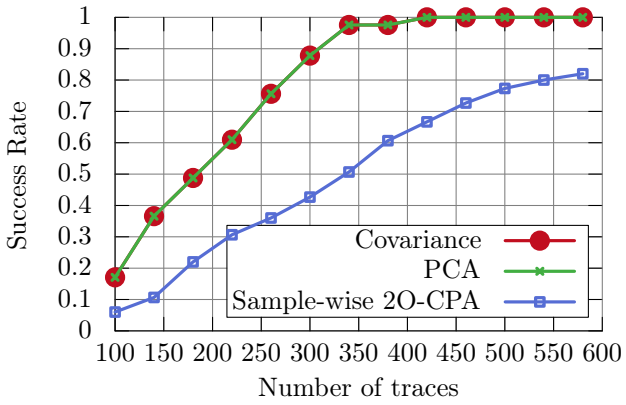


Figure 3.6: Comparison between the classical second-order CPA and second-order CPA with preprocessing using (S-boxes, S-Boxes)

3.3.5 How is the preprocessing linked to the noise?

We have theoretically shown in Proposition 13 that the presented preprocessing methods improve the SNR. We now empirically verify this results. In each point we add Gaussian noise to mimic real noisy measurements. We perform this experiment on the same points and with the same model as used for Figure 3.4.

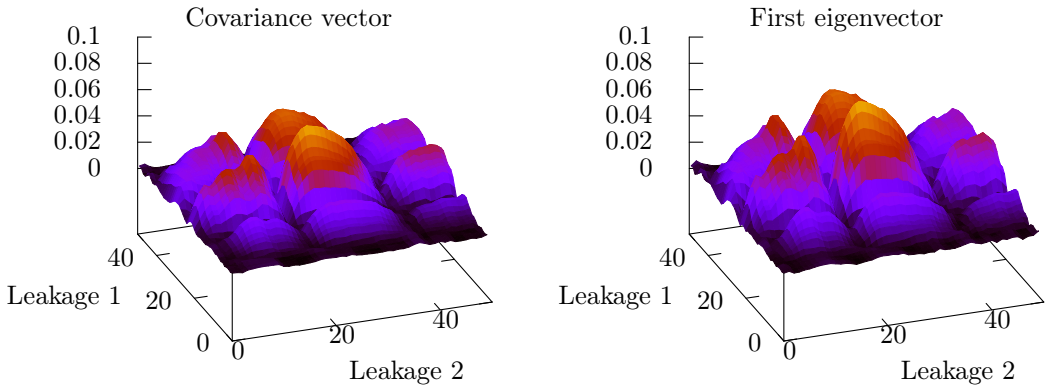
Figure 3.8a shows that using preprocessing methods improves second-order CPA in presence of noise. In this case we added Gaussian noise with a standard deviation of 3. The attacks after preprocessing need around 225 measurements to reach 80% of success whereas the 2O-CPA needs more than 550 measurements. Thus, preprocessing leads to a gain over 50%. As shown in Figure 3.4 the gain was close to 30% without noise.

In Figure 3.8b we can see that for Gaussian noise with a standard deviation of 5 the gain is more than 75%. Indeed the 2O-CPA after preprocessing needs around 250 traces reach 80% of success rate whereas for 2O-CPA 1000 measurements are not sufficient.

So this kind of preprocessing by dimensionality reduction is well designed against noisy implementation where the noise is not correlated with the time or the data.

3.4 On the fly preprocessing

We have defined a case study when the attacker owns a “learning device". As a consequence he is able to acquire a sufficient number of measurements to well estimate the covariance matrix for



(a) 2-dimensional covariance vector

(b) 2-dimensional first eigenvector

Figure 3.7: Comparison between the covariance vector and the first eigenvector

the PCA and the covariance vectors. However, the attacker might not always have this powerful tool.

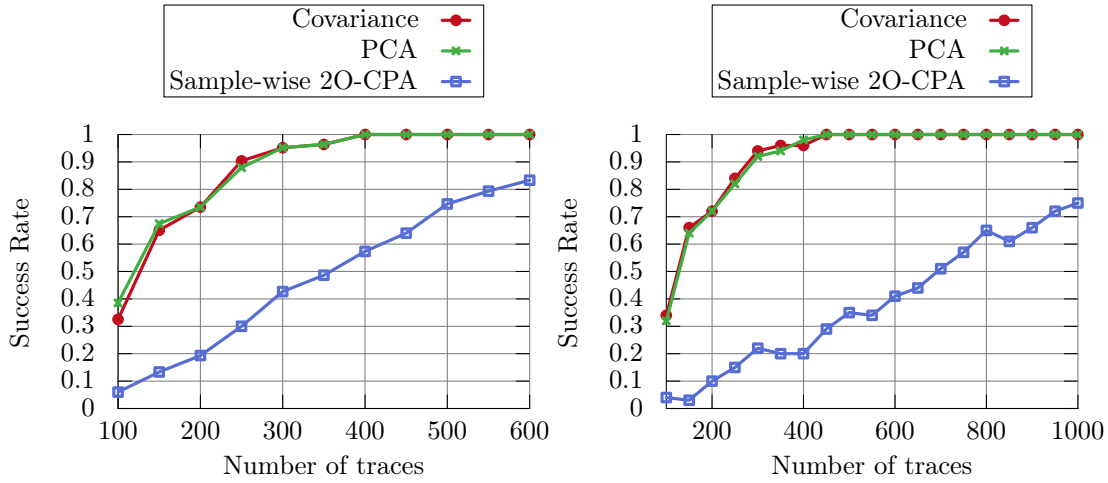
As seen in Subsect. 3.3.4 even for a small number of traces for the learning phase we have a significant improvement when we use preprocessing methods. We therefore evaluate these tools also as “on the fly" preprocessing methods.

3.4.1 Case study

We now model a less powerful attacker who does not have a “learning device" and estimates the value of the coefficient of the linear transformation directly on the traces used for the attack. Because the key is unknown the preprocessing method has to be computed for each key hypothesis. Finally, the adversary applies the covariance between the transformed data and the model depending on the key hypothesis. In this experiment we use the 10000 first traces of the DPA contest to compute the success rate which results in 25 repetitions.

3.4.2 Empirical results

Figure 3.9 illustrates the success rate after preprocessing for different sizes of the learning set for PCA (green) and the covariance vector (red). One can observe that the covariance method performs better than PCA when a low number of traces is used during the learning phase,



(a) Standard deviation of 3

(b) Standard deviation of 5

Figure 3.8: Comparison between 2O-CPA with preprocessing method and without in presence of Gaussian noise, with a standard deviation of 3 for (a) with a standard deviation of 5 for (b)

accordingly, this method is a good choice as a “on the fly" preprocessing method. The reason why the PCA method needs more measurements for the learning than the covariance method to reach its maximum efficiency during the attack phase could be the fact that the covariance matrix (see the term ${}^tXX$ in Def. 18) needs more traces to be well estimated.

Figure 3.10 shows that with the “on the fly" preprocessing we can perform 2O-CPA using 225 measurements. This represents a gain of 18% compared to raw (sample-wise) 2O-CPA.

3.5 Conclusions and Perspectives

In this chapter we presented the covariance method as an optimal preprocessing method for second-order CPA. By using all possible leakage points our method improves the efficiency of the attacks and as the number of combined leakage points grow quadratically, thus our preprocessing method is well adapted for *bi-variate* CPA. We further theoretically linked the PCA to the problem of maximization of the covariance. We demonstrated theoretically the result of the covariance method to be the optimal linear combination for maximizing the covariance and underlined empirically that this method improves the result of *bi-variate* CPA.

Compared to 2O-CPA, the attack based on the optimal covariance method is significantly improved, particularly in presence of noise and when the number of leaking points is important.

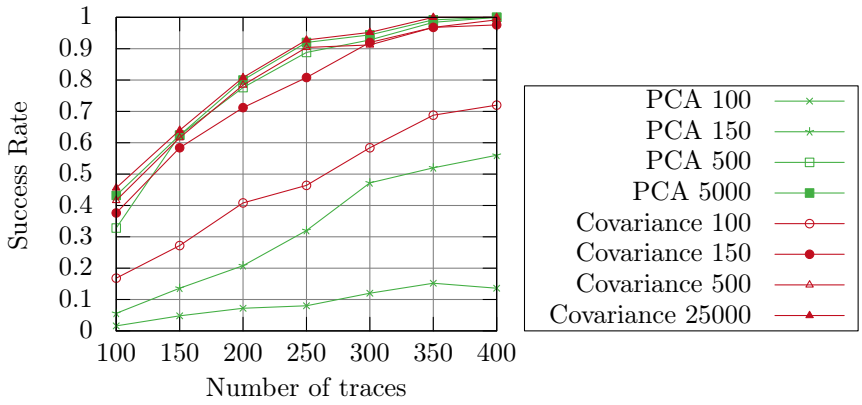


Figure 3.9: Comparison between covariance and PCA depending on the size of the learning base

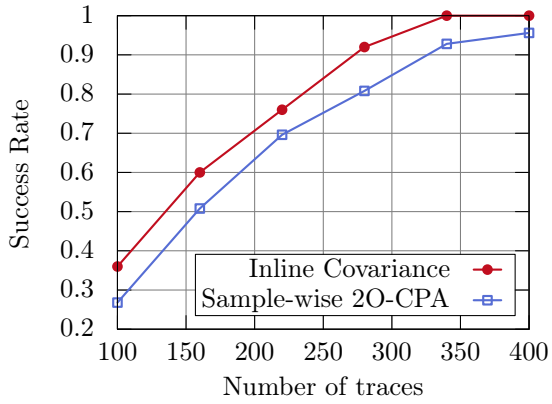


Figure 3.10: Comparison between covariance in line preprocessing and 2O-CPA

This is partly explained by the fact the optimal covariance considers all the relevant sampling points, whereas the 2O-CPA considers only the best pair of samples and does not exploit the other interesting pairs.

We have also shown that the optimal covariance method is more efficient than PCA when the learning phase is performed on the fly. All the results have been validated by experiences on real traces corresponding to masking implementation of the DPA contest v4. As a consequence dimensionality reduction by linear combination is well adapted to the case of *multi-variate* CPA. Moreover, the higher the order of masking, the more efficient the attack after preprocessing.

We could extend the previous results on other implementations which are less favorable to attacker, e.g., with more noise. Also we plan to compare the method presented in this chapter and the method presented in (70) in these cases.

Following the results of the previous chapter we could extend these results in a profiled scenario. In this context we could look at the optimal dimensionality reduction in terms of success rate in the case of masking. Following the notations introduced in the introduction we will look at the optimal dimensionality reduction in the case of bi-variate attacks. Then a generalization could be done to extend this result at any order.

Part II

Multivariate Leakages of a Masking Scheme with Table Recomputation

CHAPTER 4

Optimal Distinguisher against Masking Table

The results presented in this chapter have been published in collaboration with Sylvain Guilley, Annelie Heuser and Olivier Rioul in the international Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2014) (23).

Contents

Masking schemes based on tables recomputation are classical countermeasures against high-order side-channel attacks. Recently Tunstall, Whitnall and Oswald at FSE 2013 have exhibit a new attack which exploits the multiple leakages linked to one mask during the recomputation of tables. This attack is highly multi-variate as it exploits many different leakages depending on many different variables. As a consequence following our nomenclature this attack has a high parameter δ. Based on these results we investigate in this chapter the optimal attack in this context. We formally compute the optimal distinguisher against table recomputation masking scheme when the attacker has a full knowledge of the leakage function. This new distinguisher will provide better result in term of probability of success than the attacks of the state-of-the-

art. Moreover our analysis gives better understanding on the behavior of the attack. Indeed we show that for high noises the optimal distinguisher is closed to a sum of several 2O-CPA. As a consequence this new attack is closed to a highly multi-target attack.

4.1 Algorithm of masking tables

In the implementation of masking schemes, it is particularly challenging to compute non-linear parts of the algorithm, such as for example the S-Box of AES (a function from n bits to n bits). To solve this difficulty different methods have been proposed which can be classified in three categories (97).

- Algebraic methods (16, 141). The outputs of the S-Box will be computed using the algebraic representation of the S-Box.
- Global Look-up Table (130, 162) method. A table is precomputed off-line for each possible input and output masks.
- Table recomputation methods which precompute a masked S-Box stored in a table (1, 29, 105). Here, the full table is recomputed despite not all entries will be called. Such tables can be recomputed only once per encryption to reach first-order security. More recently, Coron presented at EUROCRYPT 2014 (39) a table recomputation scheme secure against Ωth-order attacks. Since this countermeasure aims at high-order security ($\Omega > 1$), it requires one full table precomputation before every S-Box call.

The principle of masking tables is illustrated in Alg. 1. Instead of showing a complete masked AES, only the masked computation of AddRoundKey followed by SubBytes is shown.

We have indicated the words length of all data as n, typically, $n = 8$ bit for AES. Two random masks m and m' are drawn initially from $\mathbb{F}_2^n$ and all the data manipulated by the algorithm will be exclusive-ored with one of the two masks.

Passing additively masked data through the Sbox is not obvious, as this operation is non-linear. Therefore, the Sbox is recomputed masked, as shown on lines 2 to 5: a new table S', that has also size $2^n \times n$ bits, is required for this purpose. It can be seen that the addresses are accessed sequentially in the ordinal order. In the Sbox precomputation step, the key byte k is not manipulated. The leakage only concerns the mask.

Then, the masked computation which involves the masked table follows, in lines 7 to 10. Masking the plaintext is straightforward (see line 7). Key addition can be done safely as a

second step, as the plaintext is already masked (see line 8). Of course, if the whole AES were to be evaluated, the demasking step (line 10) would be deferred till the end of the AES.

Algorithm 1: Beginning of a block cipher masked by Sbox precomputation

input : t, one byte of plaintext, and k, one byte of key
output : The application of AddRoundKey and SubBytes on t, i.e., $S(t \oplus k)$

1 $m \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$, $m' \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$ // Draw of random input and output masks
2 **for** $\omega \in \{0, 1, \ldots, 2^n - 1\}$ **do** // Sbox masking
3 $\quad z \leftarrow \omega \oplus m$ // Masked input
4 $\quad z' \leftarrow S[\omega] \oplus m'$ // Masked output
5 $\quad S'[z] \leftarrow z'$ // Creating the masked Sbox entry
6 **end**
7 $t \leftarrow t \oplus m$ // Plaintext masking
8 $t \leftarrow t \oplus k$ // Masked AddRoundKey
9 $t \leftarrow S'[t]$ // Masked SubBytes

// ... normally, the full AES is computed here ...

10 $t \leftarrow t \oplus m'$ // Demasking
11 **return** t

4.2 Classical Attacks

We now consider the attack of a masking scheme using Sbox recomputation as described in (87).

It is noteworthy that the traditional approach to reduce the multiplicity of leakage samples by a combination $C_X : \mathbb{R}^d \to \mathbb{R}$ actually *would fail* in the setup of masking tables. Indeed, the combination functions are usually considered symmetric into its arguments, meaning that any swap of the inputs does not affect the combination. This (tacit) hypothesis has been made, for instance, for

- the absolute difference $C_{ad}(X) = |X^{(0)} - X^{(1)}| = |X^{(1)} - X^{(0)}|$, and
- the centered product $C_{cp}(X) = (X^{(0)} - \mathbb{E}\left[X^{(0)}\right])(X^{(1)} - \mathbb{E}\left[X^{(1)}\right]) = (X^{(1)} - \mathbb{E}\left[X^{(1)}\right])(X^{(0)} - \mathbb{E}\left[X^{(0)}\right])$.

We assume here that the attacker applies the combination function on the leakages occurring

during the Sbox recomputation (see Alg. 1), i.e., the attacker gains 2^n leakages

$$X^{(0)} = \Psi^{(0)}(M) + N^{(0)} \tag{4.1}$$

$$X^{(1)} = \Psi^{(1)}(M \oplus 1) + N^{(1)} \tag{4.2}$$

$$\vdots$$

$$X^{(2^n-1)} = \Psi^{(2^n-1)}(M \oplus (2^n - 1)) + N^{(2^n-1)} \ , \tag{4.3}$$

and would apply[1] e.g., $C_{ad}(X)$ or $C_{cp}(X)$. Additionally, he measures the leakage $X^{(2^n)} = \Psi^{(2^n)}(T \oplus k \oplus M) + N^{(2^n)}$ and finally combines it with the previous combined leakages as $\bar{C}(X^{(2^n)}, C(X^{(0)}, \ldots, X^{(2^n-1)}))$.

Following the methodology in (133) and assuming an equal leakage function on each share[2], i.e., $\Psi = \Psi^{(0)} = \cdots = \Psi^{(2^n)}$, the optimal function to combine the predictions would then be

$$\begin{aligned} C_Y &= \mathbb{E}\{\bar{C}_X(C_X(\Psi(M), \Psi(M \oplus 1), \ldots, \Psi(M \oplus (2^n - 1))), \Psi(t \oplus k \oplus M))\} && (4.4)\\ &= \frac{1}{2^n} \sum_{m \in \mathbb{F}_2^n} \bar{C}_X(C_X(\Psi(m), \Psi(m \oplus 1), \ldots, \Psi(m \oplus (2^n - 1))), \Psi(t \oplus k \oplus m)) \\ &= \frac{1}{2^n} \sum_{m' \in \mathbb{F}_2^n} \bar{C}_X(C_X(\Psi(m' \oplus k), \ldots, \Psi(m' \oplus k \oplus (2^n - 1))), \Psi(t \oplus m')) && (4.5)\\ &= \frac{1}{2^n} \sum_{m' \in \mathbb{F}_2^n} \bar{C}_X(C_X(\Psi(m'), \Psi(m' \oplus 1), \ldots, \Psi(m' \oplus (2^n - 1))), \Psi(t \oplus m')). && (4.6) \end{aligned}$$

In Eq. (4.5), we change m for $m' = m \oplus k$ and in Eq. (4.6), the input terms at position ζ are replaced with those at position $\zeta \oplus k$ (because of the symmetry property of c). Accordingly, C_Y does not depend on the key k and is even constant as the same operation can be done on $t \oplus k$, therefore higher-order CPA fails.

Of course, the Sbox precomputation masking scheme can be attacked by the classic means, that ignore the precomputation stage (i.e., lines 7 to 10 in Alg. 1 are already vulnerable alone). More precisely, if several Sboxes are computed with different plaintext bytes and different key bytes, then *collision attacks* are possible. Also a second-order attack with a combination function can be achieved, for instance, between the addition of the mask to the plaintext and the Sbox call, i.e., between lines 7 and 9 in Alg. 1. Additionally, such second-order attack can also be

[1] The centered product combination function naturally extends from two to any number of inputs. However, the absolute difference is inherently a binary combination function. A possible generalization in the context of arity d could thought of as: $C_{ad}(X) = \sum_{\substack{0 \le \omega < d \\ 0 \le \omega' < d}} |X^{(\omega)} - X^{(\omega')}|$. Such expression remains unchanged under permutation of the inputs.

[2] This assumption is reasonable for software implementation, which is the adequate scenario for masking tables.

achieved between one leakage from the Sbox recomputation (say line 3 in Alg. 1. and line 8 (or preferably with line 9 for a better contrast (9)).

However, a better attack would consist in using altogether all the leakages from the Sbox recomputation with one (or more) of the samples used during the computation proper (starting from line 8, when the key is involved). One example of such strategy has been exposed in (174), which we label as 2-stage CPA attack.

Definition 40 (2-stage CPA attack (174))**.**

$$2{\times}\mathrm{CPA}^{mt}(\mathbf{x},\mathbf{t}) = \arg\max_{k\in\mathcal{K}} \widehat{\rho}(\mathbf{x}^{(2^n)}, y^{(2^n)}(\mathbf{t},k,\widehat{\mathbf{m}})), \tag{4.7}$$

where $\forall i\ \widehat{m}_i$ is the mask that maximizes the correlation between $x_i^{(\omega)}$ and $y_i^{(\omega)} = \omega \oplus m_i$ for $\omega \in [0, 2^n[$. This attack is a synergy between a horizontal and a vertical attack and as a consequence we will call these attacks Horizontal Vertical (HV) attacks. For each trace (separately $\forall i$), the first attack in Eq. (4.7) consists in recovering the mask during the precomputation (lines 2 to 5 in Alg. 1). Second, a regular CPA using a model in which both the plaintext t and the mask m are assumed as public knowledge is launched. Even if the mask $\widehat{m}$ is not recovered correctly for each trace (since 2^n leakage samples during the precomputation can be seen as small), it can be expected that the value of the mask is recovered by the first horizontal attack probabilistically well enough for it to be biased, i.e., better guessed than random. This gives a rough idea of the proof of soundness for this attack.

Nonetheless, this attack is probably not the most efficient, as it uses separately the information available from the Sbox precomputation and from the leakage of the AES algorithm proper. The next subsection investigates the optimal attack and gives approximation for high and low noise.

4.3 High Order Optimal Distinguisher for Precomputation Masking Tables

When using masking tables (Alg. 1) the attacker first has all leaking samples during the precomputation, i.e., $y_i^{(\omega)} = \Psi(\omega \oplus m)$ that are independent of i for $0 \leq \omega \leq (2^n - 1)$, and, second, the leakage arising from the combination of the mask m, plaintext t_i and the key, i.e., $y_i^{(2^n)} = \Psi(t_i \oplus k \oplus m)$. Thus, all terms for $\omega \neq 2^n$ do not depend on the key and the higher-order optimal distinguisher from Def. 20 can be further deduced.

Theorem 4.3.1 (OPT for masking tables)**.** *When $\Psi : \mathbb{F}_2^n \to \mathbb{R}$ is known, $N_i^{(\omega)} \sim \mathcal{N}(0, \sigma_\omega^2)$ and i.i.d. across values of q and independent across the values of $\omega = \{0, \ldots, 2^n\}$, then the higher-order optimal distinguisher against masking tables takes the form*

$$\begin{aligned}&\mathrm{OPT}(\mathbf{x}, \mathbf{t}) = \\ &\arg\max_{k \in \mathcal{K}} \sum_{i=1}^{q} \log \left\{ \sum_{m \in \mathbb{F}_2^n} \exp \left\{ \sum_{\omega \in \mathbb{F}_2^n} \frac{1}{{\sigma^{(\omega)}}^2} \left(x_i^{(\omega)} \Psi(\omega \oplus m) - \frac{1}{2} \Psi(\omega \oplus m)^2 \right) \right.\right. \\ &\left.\left. + \frac{1}{{\sigma^{(2^n)}}^2} \left(x_i^{(2^n)} \Psi(t_i \oplus m \oplus k) - \frac{1}{2} \Psi(t_i \oplus m \oplus k)^2 \right) \right\} \right\}. \end{aligned} \tag{4.8}$$

Proof. Straightforward computation from Eq. (1.23) yields

$$\arg\max_{k \in \mathcal{K}} \prod_{i=1}^{q} \sum_{m \in \mathbb{F}_2^n} \prod_{\omega \in \mathbb{F}_2^n} \exp \left\{ \frac{1}{{\sigma^{(\omega)}}^2} \left(x_i^{(\omega)} y_i^{(\omega)} - \frac{1}{2} {y_i^{(\omega)}}^2 \right) \right\} \tag{4.9}$$

$$= \arg\max_{k \in \mathcal{K}} \sum_{i=1}^{q} \log \left\{ \sum_{m \in \mathbb{F}_2^n} \exp \left\{ \sum_{\omega \in \mathbb{F}_2^n} \frac{1}{{\sigma^{(\omega)}}^2} \left(x_i^{(\omega)} y_i^{(\omega)} - \frac{1}{2} {y_i^{(\omega)}}^2 \right) \right\} \right\} \tag{4.10}$$

Now plugging the respective leakages as described in Subsect. 4.3 gives

$$\begin{aligned}&= \arg\max_{k \in \mathcal{K}} \sum_{i=1}^{q} \log \left\{ \sum_{m \in \mathbb{F}_2^n} \exp \left\{ \sum_{\omega \in \mathbb{F}_2^n} \frac{1}{{\sigma^{(\omega)}}^2} \left(x_i^{(\omega)} \Psi(\omega \oplus m) - \frac{1}{2} \Psi(\omega \oplus m)^2 \right) \right.\right. \\ &\left.\left. + \frac{1}{{\sigma^{(2^n)}}^2} \left(x_i^{(2^n)} \Psi(t_i \oplus m \oplus k) - \frac{1}{2} \Psi(t_i \oplus m \oplus k)^2 \right) \right\} \right\}. \end{aligned} \tag{4.11}$$

□

Proposition 15 (HOOD for masking tables for low SNR)**.** *For large Gaussian noise (or low* SNR*) the distinguisher becomes*

$$\begin{aligned}&\mathrm{OPT\text{-}high}(\mathbf{x}, \mathbf{t}) = \\ &\arg\max_{k \in \mathcal{K}} \sum_{\omega \in \mathbb{F}_2^n} \frac{1}{{\sigma^{(\omega)}}^2} \sum_{i=1}^{q} \begin{pmatrix} x_i^{(\omega)} x_i^{(2^n)} \sum_m \Psi(\omega \oplus m) \Psi(t_i \oplus k \oplus m) \\ -\frac{1}{2} x_i^{(2^n)} \sum_m \Psi(t_i \oplus k \oplus m) \Psi(\omega \oplus m)^2 \\ -\frac{1}{2} x_i^{(\omega)} \sum_m \Psi(\omega \oplus m) \Psi(t_i \oplus k \oplus m)^2 \\ +\frac{1}{4} \sum_m \Psi(\omega \oplus m)^2 \Psi(t_i \oplus k \oplus m)^2 \end{pmatrix}. \end{aligned} \tag{4.12}$$

Proof. Due to the lack of space we neglect the term $\arg\max_{k \in \mathcal{K}}$ in front of each line. Starting from Eq. (1.23) we use again the first-order Taylor expansion $\exp\{\varepsilon\} = 1 + \varepsilon + O(\varepsilon^2)$. So,

$$\prod_{i=1}^{q} \sum_{m \in \mathbb{F}_2^n} \prod_{\omega=0}^{2^n} \Big(1 + \frac{1}{{\sigma^{(\omega)}}^2} \big(x_i^{(\omega)} y_i^{(\omega)} - \frac{1}{2} {y_i^{(\omega)}}^2 \big) + \frac{1}{2{\sigma^{(\omega)}}^4} \big(x_i^{(\omega)} y_i^{(\omega)} - \frac{1}{2} {y_i^{(\omega)}}^2 \big)^2 \Big).$$

Furthermore, an expansion at second-order gives

$$\prod_{i=1}^{q}\sum_{m\in\mathbb{F}_2^n}\Big(1+\sum_{\omega=0}^{2^n}\frac{1}{\sigma^{(\omega)^2}}\big(x_i^{(\omega)}y_i^{(\omega)}-\frac{1}{2}{y_i^{(\omega)}}^2\big)+\frac{1}{2\sigma^{(\omega)^4}}\big(x_i^{(\omega)}y_i^{(\omega)}-\frac{1}{2}{y_i^{(\omega)}}^2\big)^2 + \sum_{\omega\neq\omega'}^{2^n}\frac{1}{\sigma^{(\omega)^2}\sigma^{(\omega')^2}}\big(x_i^{(\omega)}y_i^{(\omega)}-\frac{1}{2}{y_i^{(\omega)}}^2\big)\big(x_i^{(\omega')}y_i^{(\omega')}-\frac{1}{2}{y_i^{(\omega')}}^2\big)\Big). \tag{4.13}$$

From the perfect masking condition (see (22, Proposition 1)) the first-order term

$$\sum_{m\in\mathbb{F}_2^n}\sum_{\omega=0}^{2^n}\frac{1}{\sigma^{(\omega)^2}}\big(x_i^{(\omega)}y_i^{(\omega)}-\frac{1}{2}{y_i^{(\omega)}}^2\big)=\sum_{\omega=0}^{2^n}\frac{1}{\sigma^{(\omega)^2}}\big(x_i^{(\omega)}\sum_{m\in\mathbb{F}_2^n}y_i^{(\omega)}-\frac{1}{2}\sum_{m\in\mathbb{F}_2^n}{y_i^{(\omega)}}^2\big)$$

is constant as well as

$$\sum_{m\in\mathbb{F}_2^n}\sum_{\omega=0}^{2^n}\frac{1}{2\sigma^{(\omega)^4}}\big(x_i^{(\omega)}y_i^{(\omega)}-\frac{1}{2}{y_i^{(\omega)}}^2\big)^2 \tag{4.14}$$

$$=\sum_{\omega=0}^{2^n}\frac{1}{2\sigma^{(\omega)^4}}\big({x_i^{(\omega)}}^2\sum_{m\in\mathbb{F}_2^n}{y_i^{(\omega)}}^2+\frac{1}{4}\sum_{m\in\mathbb{F}_2^n}{y_i^{(\omega)}}^4-x_i^{(\omega)}\sum_{m\in\mathbb{F}_2^n}{y_i^{(\omega)}}^3\big). \tag{4.15}$$

The other terms in ω,ω' can be written as

$$2\sum_{\omega<\omega'}^{2^n}\frac{1}{\sigma^{(\omega)^2}\sigma^{(\omega')^2}}\Big(x_i^{(\omega)}x_i^{(\omega')}\sum_{m\in\mathbb{F}_2^n}y_i^{(\omega)}y_i^{(\omega')}-\frac{1}{2}x_i^{(\omega')}\sum_{m\in\mathbb{F}_2^n}y_i^{(\omega')}{y_i^{(\omega)}}^2 -\frac{1}{2}x_i^{(\omega)}\sum_{m\in\mathbb{F}_2^n}y_i^{(\omega)}{y_i^{(\omega')}}^2+\frac{1}{4}\sum_{m\in\mathbb{F}_2^n}{y_i^{(\omega)}}^2{y_i^{(\omega')}}^2\Big). \tag{4.16}$$

Moreover, all terms involving only combinations of $\omega<d=2^n$ do not depend on the key, thus we can further simplify to the required equation

$$\sum_{\omega\in\mathbb{F}_2^n}\frac{1}{\sigma^{(\omega)^2}}\left(\sum_{i=1}^{q}x_i^{(\omega)}x_i^{(2^n)}\sum_{m\in\mathbb{F}_2^n}y^{(\omega)}y^{(2^n)}-\frac{1}{2}x_i^{(2^n)}\sum_{m\in\mathbb{F}_2^n}y^{(2^n)}{y^{(\omega)}}^2\right. \tag{4.17}$$

$$\left.-\frac{1}{2}x_i^{(\omega)}\sum_{m\in\mathbb{F}_2^n}y^{(\omega)}{y^{(2^n)}}^2+\frac{1}{4}\sum_{m\in\mathbb{F}_2^n}{y^{(\omega)}}^2{y^{(2^n)}}^2\right). \tag{4.18}$$

□

Proposition 16 (Relationship between HOOD and CPA for masking tables)**.** *When all noise variances are equal, i.e., $\sigma=\sigma^{(\omega)}\ \forall\omega$, Eq. (4.12) further simplifies to*

$$\text{OPT-high}(\mathbf{x},\mathbf{t})=\arg\max_{k\in\mathcal{K}}\sum_{\omega\in\mathbb{F}_2^n}\sum_{i=1}^{q}\Big(x_i^{(\omega)}x_i^{(2^n)}\sum_{m\in\mathbb{F}_2^n}\Psi(\omega\oplus m)\Psi(t_i\oplus k\oplus m) -\frac{1}{2}x_i^{(\omega)}\sum_{m\in\mathbb{F}_2^n}\Psi(\omega\oplus m)\Psi(t_i\oplus k\oplus m)^2\Big), \tag{4.19}$$

which becomes close to a combination of higher-order CPAs, i.e.,

$$\text{C-CPA}(\mathbf{x},\mathbf{t}) = \arg\max_{k\in\mathcal{K}} \sum_{\omega\in\mathbb{F}_2^n} \rho(c_X^{n\text{-}prod}(\mathbf{x}^{(\omega)},\mathbf{x}^{(2^n)}), c_Y^{opt}(\mathbf{y}^{(\omega)},\mathbf{y}^{(2^n)})) \\ - \frac{1}{2}\rho(\mathbf{x}^{(\omega)}, c_Y^{opt}(\mathbf{y}^{(\omega)},\mathbf{y}^{(2^n)^2})). \tag{4.20}$$

Proof. If all the variances are equal we have

$$\sum_{\omega\in\mathbb{F}_2^n} \frac{\Psi(\omega\oplus m)^2}{\sigma^{(\omega)}} = \frac{1}{\sigma}\sum_{\omega\in\mathbb{F}_2^n}\Psi(\omega\oplus m)^2 = \frac{1}{\sigma}\sum_{\omega\in\mathbb{F}_2^n}\Psi(\omega)^2. \tag{4.21}$$

So, regarding the second term in Eq. (4.12) we have

$$\sum_{\omega\in\mathbb{F}_2^n} \frac{1}{\sigma^{(\omega)^2}} \sum_{i=1}^{q} x_i^{(2^n)} \sum_{m\in\mathbb{F}_2^n} \Psi(t_i\oplus k\oplus m)\Psi(\omega\oplus m)^2 \tag{4.22}$$

$$= \sum_{i=1}^{q} x_i^{(2^n)} \sum_{m\in\mathbb{F}_2^n} \Psi(t_i\oplus k\oplus m) \sum_{\omega\in\mathbb{F}_2^n} \frac{1}{\sigma^{(\omega)^2}}\Psi(\omega\oplus m)^2 \tag{4.23}$$

$$= \sum_{i=1}^{q} x_i^{(2^n)} \sum_{m\in\mathbb{F}_2^n} \Psi(t_i\oplus k\oplus m) \sum_{\omega\in\mathbb{F}_2^n} \frac{1}{\sigma^{2}}\Psi(\omega)^2 \tag{4.24}$$

$$= \sum_{i=1}^{q} x_i^{(2^n)} \sum_{m\in\mathbb{F}_2^n} \Psi(t_i\oplus m) \sum_{\omega\in\mathbb{F}_2^n} \frac{1}{\sigma^{2}}\Psi(\omega)^2, \tag{4.25}$$

which clearly does not depend on the key k. The same goes for the fourth term, which proofs the first part. Now, rewriting Eq. (4.19) gives

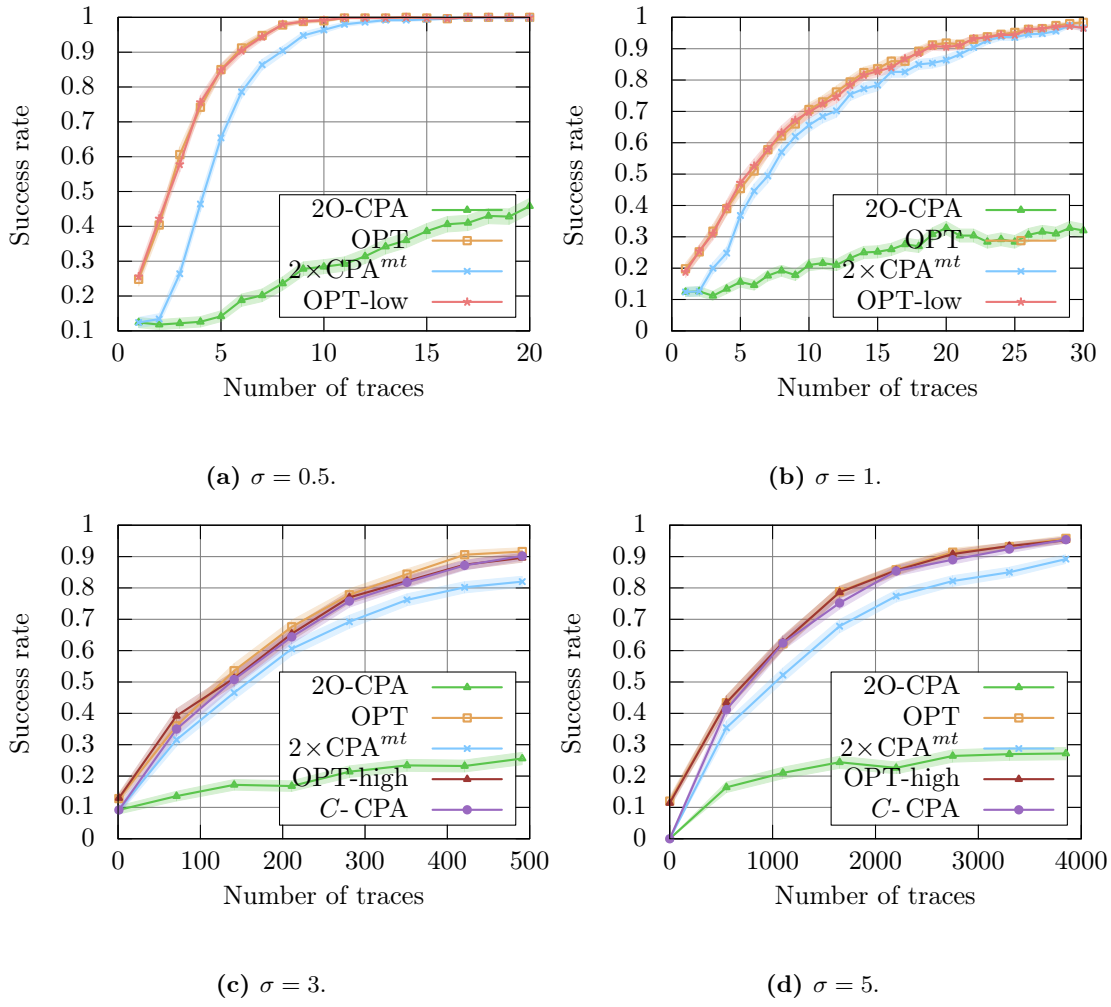
$$\arg\max_{k\in\mathcal{K}} \sum_{\omega\in\mathbb{F}_2^n} \langle \mathbf{x}^{(\omega)}\mathbf{x}^{(2^n)} \mid \sum_{m\in\mathbb{F}_2^n} \Psi(\omega\oplus m)\Psi(\mathbf{t}\oplus k\oplus m)\rangle \\ -\langle \frac{1}{2}\mathbf{x}^{(\omega)} \mid \sum_{m\in\mathbb{F}_2^n} \Psi(\omega\oplus m)\Psi(\mathbf{t}\oplus k\oplus m)^2\rangle, \tag{4.26}$$

and using the same argumentation as in the proof of (22, Proposition 9) gives the required formula from the second part. □

Interestingly, instead of using one CPA to recover the mask and one to recover the secret key (see Def. 40) we recover that the best methodology is to attack each share $\omega < 2^n$ with $\omega = 2^n$ (minus a regulation term) and then use a combination of all attacks.

In other words a better attack is to build an attack which recover the key in several manners based on different variables and then combined them. As a consequence we have build here an attack which is highly multi-target and as a consequence its parameter τ is also high.

Remark 16. *For low noise, we can straightforwardly use (22, Proposition 10), which is validated in our empirical results.*



(a) $\sigma = 0.5$.

(b) $\sigma = 1$.

(c) $\sigma = 3$.

(d) $\sigma = 5$.

Figure 4.1: Success Rate for masking table.

4.3.1 Experimental Validation

To empirically validate our theoretical results we use simulations of a first order masking scheme with precomputation tables. We target the xor operation in the precomputation phase and the AddRoundKey of the algorithm (see line 3 and line 8 of Alg. 1). For computationally reasons for all distinguishers we only target four bits ($n = 4$).

Remark 17. *Targeting the AddRoundKey phase has some advantages. First, it allows to perform the evaluation on only four bits without the loss of generality of using a four bits Sbox. Second, in the Sbox precomputation algorithm of Coron (39) the output masks are different for each entry of the Sbox and could therefore not be combined with the mask of the precomputation table. However, as in our analysis the attacker can still take advantage of the* 2^n *leakages of the*

masked inputs of the Sbox combined with the leakage of the AddRoundKey operation.

Similarly to the previous experiments, T is uniformly distributed over $\mathbb{F}_2^4$ and the noise is arising from a Gaussian distribution $N \sim \mathcal{N}(0, \sigma^2)$ for $\sigma = \sigma^{(0)} = ... = \sigma^{(16)} \in \{0.5, 1, 3, 5\}$. Again to compute the success rate we conducted 500 independent experiments with uniformly distributed $k^\star$ and shaded the success rate with error bars.

Figure 4.1 shows the success rate. For low noise ($\sigma = 0.5$ and $\sigma = 1$) the optimal distinguisher (HOOD) and its approximation for low noise (HOOD-low) perform similar and better than the 2*nd*-order CPA (2O-CPA) with normalized product combination function and the 2-stage CPA in Eq. 4.7 (2xCPA). Naturally, all distinguishers outperform 2nd-order CPA as it only utilizes two leakages $X^{(0)}$ and $X^{(256)}$. For higher noise ($\sigma = 3$ and $\sigma = 5$) the HOOD and its approximation for high noise (HOOD-high) perform better than the 2-stage CPA (2xCPA) and 2nd-order CPA. Moreover it can be noticed that the distinguisher based on combinations of CPA (Eq. (4.20)) (C-CPA) and the optimal ones are equally efficient. Accordingly, we have empirically validated that our new distinguisher approximated from the optimal distinguisher is valid for high noise and more efficient than the two-stage CPA. In particular, it requires around 1000 traces less to reach $\widehat{\mathbb{P}}_S = 90\%$ for $\sigma = 5$.

4.4 Classical countermeasure

The strategy to protect the table recomputation against HV attacks and the distinguisher presented in (23) is to shuffle the recomputation, i.e., do the recomputation in a random order, as illustrated in Alg. 2.

Different methods to randomize the order are presented in (174). One of the methods presented is based on a random permutation on a subset of $\mathbb{F}_2^n$.

Let S_{2^n} the symmetric group of 2^n elements, which represents all the ways to shuffle the set $\{0, \ldots, 2^n - 1\}$. If the random permutation over $\mathbb{F}_2^n$ is randomly drawn from a set of permutation $S \subset S_{2^n}$, where $card\,(S) \ll card\,(S_{2^n})$, it is still possible for an attacker to take advantage of the table recomputation. Indeed as it is shown in (174) attacks could be built by including all the possible permutations alongside with the key hypothesis. If the permutation is drawn uniformly over the S_{2^n} the number of added hypothesis is $2^n!$ which can be too much for attacks. For instance, for $n = 8$, we have $2^8! \approx 2^{1684}$.

By generating a highly entropic permutation, such as defined in (174) or any pseudo random permutation generator (RC4 key scheduler...), a designer could protect table recomputation

against HV attacks. Indeed using for example five or six bytes of entropy as seed for the permutation generator could be enough to prevent an attacker from guessing all the possible permutations.

The table is recomputed in a random order from line 3 to line 7.

Algorithm 2: Beginning of computation of a block cipher masked by table recomputation in a random order

input : t, one byte of plaintext, and k, one byte of key
output : The application of AddRoundKey and SubBytes on t, i.e., $S[t \oplus k]$

```
// Table precomputation protected by shuffling .............................
```

1 $m \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$, $m' \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$ // Draw of random input and output masks
2 $\varphi \leftarrow_{\mathcal{R}} \mathbb{F}_2^n \to \mathbb{F}_2^n$, bijective // Draw of random permutation of $\mathbb{F}_2^n$
3 **for** $\omega \in \{0, 1, \ldots, 2^n - 1\}$ **do** // S-box masking
4 $z \leftarrow \varphi(\omega) \oplus m$ // Masked input
5 $z' \leftarrow S[\varphi(\omega)] \oplus m'$ // Masked output
6 $S'[z] \leftarrow z'$ // Creating the masked S-box entry
7 **end**

```
// Masked computation ......................................................
```

8 $t \leftarrow t \oplus m$ // Plaintext masking
9 $t \leftarrow t \oplus k$ // Masked AddRoundKey
10 $t \leftarrow S'[t]$ // Masked SubBytes
11 $t \leftarrow t \oplus m'$ // Demasking
12 **return** t

4.5 Conclusions and Perspectives

In this chapter we used the optimal distinguisher to build the best possible attack.

We provide a new distinguisher based on correlation whose again is as efficient as the optimal distinguisher in case of high noise. Naturally, this new distinguisher outperforms all known (non-profiled) distinguisher for this application. Given all these results we theoretically and empirically show that for high noise the security analysis with non-profiled distinguisher is sufficient as it coincides with the optimal distinguisher. Interestingly this new distinguisher based on correlation as the property to be closed to a sum of high order attacks and as a consequence to be closed to multi-target attack. These results show that non-profiled distinguishers maybe enough to ensure the security of protected implementations at the condition to properly take into account all leakages and specifically the multi-target leakages.

These results raise various new perspectives. First of all, our methodology of starting from the optimal distinguisher and deriving approximated distinguisher could be applied to other scenarios. One application, for example, could be the scenario used in (145). Moreover, future work should deal with the exact analysis of the impact of noise on the masking efficiency in a theoretical manner. This comes along with an analysis of the impact of the number of shares, in particular, with an investigation of the arguments done in (132, 177) about exponential attack complexity.

CHAPTER 5

Multivariate High Order Attack against shuffled Masking Table

The results presented in this chapter have been published in collaboration with Sylvain Guilley, Zakaria Najm and Yannick Teglia in the international Workshop on Cryptographic Hardware and Embedded Systems (CHES 2015) (26).

Contents

Masking schemes based on tables recomputation are classical countermeasures against high-order side-channel attacks. Still, they are known to be attackable at order Ω in the case the masking involves Ω shares. In this work, we mathematically show that an attack of order strictly greater than Ω can be more successful than an attack at order Ω. To do so, we leverage the idea presented by Tunstall, Whitnall and Oswald at FSE 2013: we exhibit attacks which exploit the multiple leakages linked to one mask during the recomputation of tables. Specifically, regarding first-order table recomputation, improved by a shuffled execution, we show that there is a window of opportunity, in terms of noise variance, where a novel highly multivariate third-order attack is

more efficient than a classical bivariate second-order attack. Moreover, we show on the example of the high-order secure table computation presented by Coron at EUROCRYPT 2014 that the window of opportunity enlarges linearly with the security order Ω. Here, we also investigate the case of degree one leakage models, and formally show that the Hamming weight model is the less favorable to the attacker. Eventually, we validate our attack on a real ATMEL smartcard.

5.1 Introduction

Contributions.

Our first contribution is to describe a new HODPA tailored to target the table recomputation despite a highly entropic masking (unexploitable by exhaustive search). More precisely, we propose an innovative combination function, which has the specificity to be highly multivariate. We relate the combination function of state-of-the-art and our new HODPA attacks to their success rate, which allows for a straightforward comparison. In particular, we compare the success rates of our highly multivariate HODPA (exploiting leakages in the table recomputation as well as in the masked algorithm, where the secret key is used) and of a state-of-the-art HODPA (exploiting only the leakages within the masked algorithm).

We build a theoretical analysis of their success rate. Our analysis reveals that there is a window of opportunity, when the noise variance is smaller than a threshold, where our new HODPA is more successful than a straightforward HODPA, despite it is of higher-order. This analysis also allows to identify the relevant parameter which impact the success of the attacks. In particular the impact of the leakage functions is identified, and as a consequence the best and the worst cases for our new attack are found. Similarly using the *success exponent* our theoretical analysis gives the best/worst cases in terms of noise variance.

For instance in this chapter we attack a first-order masking scheme based on table recomputation with a $(2^{n+1}+1)$-variate third-order attack more efficiently than with a classical bivariate second-order attack. In this case HV attacks could not be applied. This is the first time that a non minimal order attack is proved better (in terms of success rate) than the attack of minimal order. Actually, this non intuitive result arises from a relevant selection of leaking samples — this question is seldom addressed in the side-channel literature. We generalize our attack to a higher-order masking scheme based on tables recomputation (Coron, EUROCRYPT 2014), and prove that it remains better than a classical attack, with a window of opportunity that actually grows linearly with the masking order Ω.

Finally we propose a new innovative countermeasure in order to protect masking scheme based on tables recomputation against our new attack.

Outline of the chapter.

The rest of the chapter is organized as follows. In Sect. 5.2 we propose a new attack against the "protected" implementation of the table recomputation, prove theoretically the soundness of the attack and validate these results by simulation. In Sect. 5.3 we apply this attack on a higher-order masking scheme. Sect. 5.4 extends our results to the case where the leakage function is affine in the bits of the targeted sensitive variable. In Sect. 5.5 we validate our results on real traces. Finally in Sect. 5.6 we present a countermeasure to mitigate the impact of our new attack.

5.1.1 Preliminary and notations

In order to conduct a Ωth-order attack an attacker should combine the leakages of Ω shares. To combine these leakages an attacker will use a *combination function* (29, 107, 121). The degree of this combination function must be at least Ω for the attack to succeed. The *combination function* will then be applied both on the measured leakages and on the model (this is the optimal HODPA). As a consequence, an HODPA is completely defined by the *combination function* used.

In the rest of the chapter the SNR is given by the following definition:

Definition 41 (Signal to noise ratio)**.** *The Signal to Noise Ratio of a leakage denoted by a random variable L depending on informative part denoted I is given by:*

$$\mathrm{SNR}\left[L, I\right] = \frac{Var\left[\mathbb{E}\left[L|I\right]\right]}{\mathbb{E}\left[Var\left[L|I\right]\right]} \ . \tag{5.1}$$

An attack is said *sound* when it allows to recover the key $k^\star$ with success probability which tends to one when the number of measurements tends to the infinity.

5.2 Totally random permutation and attack

In this section we present a new attack against shuffled table recomputation shown in Alg. 2 of Chapt. 4. The success of this attack will not be impacted by the entropy used to generate the shuffle. As a consequence this attack will succeed when the HV attacks will fail because the

quantity of entropy used to generate the shuffle is too large to be exhaustively enumerated. We then express the condition where this attack will outperform the state-of-the-art second order attack.

5.2.1 Defeating the countermeasure

As the permutation φ is completely random, the value of the current index in the **for** loop (line 3 to line 7 in Alg. 2) is unknown. But it can be noticed that this current index $\boldsymbol{\varphi(\omega)}$, printed in boldface for clarity, is manipulated twice at each step of the loop (line 4, line 5):

$$z \leftarrow \boldsymbol{\varphi(\omega)} \oplus m \ , \tag{5.2}$$

$$z' \leftarrow S[\boldsymbol{\varphi(\omega)}] \oplus m' \ . \tag{5.3}$$

Let U a random variable uniformly drawn over $\mathbb{F}_2^n$ and $m \in \mathbb{F}_2^n$ a constant. Then, it is shown in (133) that:

$$\mathbb{E}\left[(\mathrm{HW}[U] - \mathbb{E}\left[\mathrm{HW}[U]\right]) \times (\mathrm{HW}[U \oplus m] - \mathbb{E}\left[\mathrm{HW}[U \oplus m]\right])\right] = -\frac{\mathrm{HW}[m]}{2} + \frac{n}{4} \ . \tag{5.4}$$

As a consequence, it may be possible for an attacker to exploit the leakage depending on the two manipulations (Eq. (5.2) and (5.3)) of the current random index in the loop. Indeed, at each of the 2^n steps of the loop in the table recomputation, the leakage of the $\boldsymbol{\varphi(\omega)}$ in Eq. (5.2) and (5.3) which plays the role of U in Eq. (5.4) will be combined (by a centered product) to recover a variable depending on the mask. Afterwards, these 2^n variables will be combined together (by a sum) in order to increase the SNR as much as possible. Finally, this sum is combined (again by a centered product) with a leakage depending on the key. This rough idea of the attack is illustrated it on Fig. 5.1, which represents the "*trace*" corresponding to the *dynamic execution* of Alg. 2, followed by the masked AES AddRoundKey & SubBytes steps.

Remark 18 (Construction of the high-order attack)**.** *The construction of the attack depicted in Fig. 5.1 leverages on two building blocks:*

1. *the centered product, represented as $\boxtimes$, which allows to get rid of a mask (recall Eq. (5.4)), albeit at the expense of a smaller SNR (it is squared, as shown in (46) – see Sec. 5.2.3)*
2. *the sum of variables with the same leakage model, represented as $\boxplus$, which increases the SNR linearly with the number of variables summed together.*

An attacker could want to perform the attack on the output of the S-Box. But depending on the implementation of the masking scheme the output masks can be different for each address

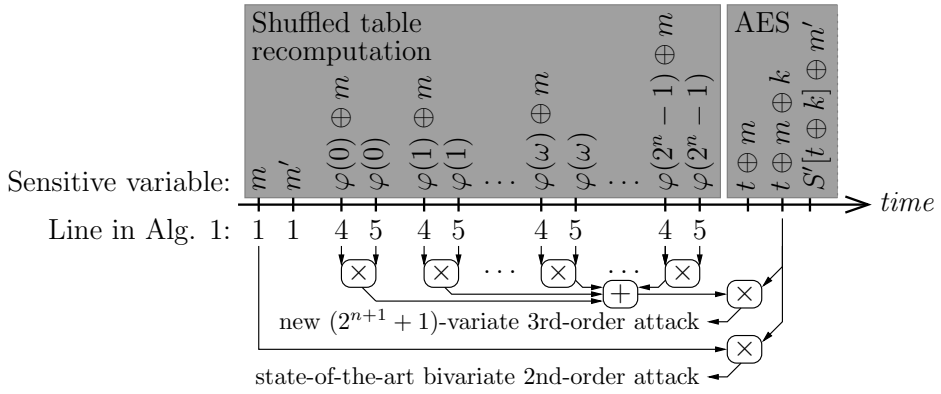


Figure 5.1: State-of-the-art attack and new attack investigated in this chapter.

of the S-Box (see for example the masking scheme of Coron (39)). To avoid loss of generality we focus our study on the S-Box input mask of the recomputation. Indeed by design of the table recomputation masking scheme, the input mask is the same for each address of the S-Box: the attacker can thus exploit it multiple times. Moreover an attacker can still take advantage of the confusion of the S-Box (53) to better discriminate the various key candidates. Indeed he can target the input the of SubBytes operation of the last round. Notice the use of capital M and capital Φ, which indicates that the leakage is modeled as a random variable.

5.2.2 Multivariate attacks against table recomputation

In the previous section, it has been shown that at each iteration of the loop of the table recomputation, it is possible to extract a value depending on the mask. As a consequence it is possible to use all of these values to perform a multivariate attack. In this subsection we give the formal formula of this new attack. Let us define the leakages of the table recomputation. The leakage of the masked random index in the loop is given by: $\mathrm{HW}[\Phi(\omega) \oplus M] + N_\omega^{(1)}$. The leakage of the random index is given by: $\mathrm{HW}[\Phi(\omega)] + N_\omega^{(2)}$. In this chapter as no matrix notation is used the bottom index is used to index the step in the loop.

Depending on the knowledge about the model, the leakage could be centered by the "true" expectation or by the estimation of this expectation. We assume this expectation is a known value given by: $\mathbb{E}\left[\mathrm{HW}[\Phi(\omega) \oplus M] + N_\omega^{(1)}\right] = \mathbb{E}\left[\mathrm{HW}[\Phi(\omega)] + N_\omega^{(2)}\right] = \frac{n}{2}$. Then let us denote

the central leakages as:

$$X_\omega^{(1)} = \mathrm{HW}[\Phi(\omega) \oplus M] + N_\omega^{(1)} - \frac{n}{2} \ , \tag{5.5}$$

$$X_\omega^{(2)} = \mathrm{HW}[\Phi(\omega)] + N_\omega^{(2)} - \frac{n}{2} \ . \tag{5.6}$$

Besides, the leakage of the masked AddRoundKey is:

$$X^\star = \mathrm{HW}[T \oplus M \oplus k^\star] + N - \frac{n}{2} \ . \tag{5.7}$$

In a view to use all the leakages of the table recomputation, an original combination function could be defined.

Definition 42. *The combination function C_{TR} exploiting the leakage of the table recomputation is given by:*

$$\begin{array}{rccc} \mathrm{C}_{TR}: & \mathbb{R}^{2^{n+1}} \times \mathbb{R} & \longrightarrow & \mathbb{R} \\ & \left(\left(X_\omega^{(1)}, X_\omega^{(2)}\right)_{0 \leqslant \omega \leqslant 2^n-1}, X^\star\right) & \longmapsto & \left(-2 \times \frac{1}{2^n} \sum_{\omega=0}^{2^n-1} X_\omega^{(1)} \times X_\omega^{(2)}\right) \times X^\star \ . \end{array}$$

Following the Fig. 5.1 it can be noticed that C_{TR} is in fact the combination of two sub-combination functions. Indeed, first of all, the leakages of the table recomputation are combined; the result of this combination is the following value:

$$X_{TR} = -2 \times \frac{1}{2^n} \sum_{\omega=0}^{2^n-1} X_\omega^{(1)} \times X_\omega^{(2)} \ . \tag{5.8}$$

Second, this value is multiplicatively combined with $X^\star$.

Remark 19. *It can be noticed that the random variable X_{TR} does not depend on Φ. Indeed in Eq. (5.8) the sum can be reordered by Φ. Moreover as this sum is computed over all the possible $\Phi(\omega)$ it implies that $\frac{1}{2^n} \sum_{\omega=0}^{2^n-1} X_\omega^{(1)} \times X_\omega^{(2)}$ is exactly the expectation over the $\Phi(\omega)$. As a consequence X_{TR} is random only through the mask and the noise.*

Based on the combination function C_{TR}, a multivariate attack can be built.

Definition 43. *The MultiVariate Attack (MVA) exploiting the leakage of the table recomputation (TR) is given by the function:*

$$\begin{array}{rccc} \mathrm{MVA}_{TR}: & \mathbb{R}^{2^{n+1}} \times \mathbb{R} \times \mathbb{R} & \longrightarrow & \mathbb{F}_2^n \\ & \left(\left(X_\omega^{(1)}, X_\omega^{(2)}\right)_\omega, X^\star, Y\right) & \longmapsto & \underset{k \in \mathbb{F}_2^n}{\mathrm{argmax}}\, \rho\left[\mathrm{C}_{TR}\left(\left(X_\omega^{(1)}, X_\omega^{(2)}\right)_\omega, X^\star\right), Y\right], \end{array}$$

where $Y = \mathbb{E}\left[\left(\mathrm{HW}[T \oplus M \oplus k] - \frac{n}{2}\right) \cdot \left(\mathrm{HW}[M] - \frac{n}{2}\right) | T\right]$ and ρ is the Pearson coefficient. According to Eq. (5.4), the model Y is equal to an affine transformation of $-\mathrm{HW}[T \oplus k]$ (note the negative sign for the correlation ρ extremal value when $k \in \mathbb{F}_2^n$ to be positive).

Proposition 17. MVA_{TR} *is sound.*

Proof. By the law of large numbers, correlation coefficient involved in the expression of MVA_{TR} tends to $\rho(-\mathrm{HW}[T \oplus k^\star], -\mathrm{HW}[T \oplus k])$ when the number of traces tends to infinity. This quantity is maximal when $k = k^\star$, by the Cauchy-Schwarz theorem. Then for enough traces the noise will impact all the key guesses similarly and as a consequence the result of MVA_{TR} is maximal when $k = k^\star$. □

Remark 20. *The attack presented in Def. 57 is a* $(2^{n+1}+1)$*-multivariate third order attack.*

Let us denote the leakage of the mask (which occurs at line 1 of Alg. 2) by:

$$X^{(3)} = \mathrm{HW}[M] + N^{(3)} - \frac{n}{2} \ . \tag{5.9}$$

Definition 44. *We denote by* 2O-CPA *the* CPA *using the centered product as combination function. Namely:*

$$\begin{array}{rccc} \text{2O-CPA}: & \mathbb{R}\times\mathbb{R}\times\mathbb{R} & \longrightarrow & \mathbb{F}_2^n \\ & \left(X^{(3)}, X^\star, Y\right) & \longmapsto & \underset{k\in\mathbb{F}_2^n}{\operatorname{argmax}}\, \rho\left[X^{(3)}\times X^\star, Y\right] \ . \end{array}$$

A careful look at Def. 42, Def. 57 and Eq. (5.8) reveals that the only difference between the MVA_{TR} and the 2O-CPA is the use of X_{TR} instead of $X^{(3)}$. Thus X_{TR} will act as the leakage of the mask. Let us call X_{TR} the *second order leakage.*

Lemma 4. *The informative part of the second order leakage is the same as the informative part of the leakage mask i.e.,*

$$\mathbb{E}\left[X_{TR}|M=m\right] = \mathbb{E}\left[X^{(3)}|M=m\right] \ .$$

Proof. It is a straightforward application of the results of (133): Use Eq. (5.4) and notice the intentional -2 factor in Eq. (5.8). Both expectations are thus equal to $\mathrm{HW}[m]$. □

5.2.3 Leakage analysis

By using the formula of the theoretical success rate (SR) we show that as the same operations are targeted by the MVA_{TR} and the 2O-CPA. Consequently, it is equivalent to compare the SNR or the SR of these attacks. Based on this fact we can theoretically establish the conditions in which the MVA_{TR} outperforms the 2O-CPA. These conditions are given in Theorem 5.2.1.

Recently A.A Ding et al. (46, §3.4) give the following formula to establish the Success Rate (SR) of second-order attacks:

$$\mathrm{SR} = \Phi_{N_k-1}\Big(\frac{\sqrt{b}\ \delta_0\delta_1}{4}K^{-1/2}\kappa\Big) \ . \tag{5.10}$$

In this formula:

- δ_0 denotes the SNR of the first share and δ_1 denotes the SNR of the second one;
- Φ_{N_k-1} denotes the cumulative distribution function of (N_k-1)-dimensional standard Gaussian distribution; as underlined by the authors in (46), if the noise distribution is not multi-variate Gaussian, then Φ_{N_k} is to be understood as its cumulative distribution function;
- N_k denotes the number of key candidates;
- K denotes the confusion matrix and κ the confusion coefficient;
- b denotes the number of traces.

Remark 21. *An update version of this formula for first order* CPA *has been presented in (52) which solves the issue of the non invertible matrix.*

This formula allows to establish the link between the SNR and SR of second order attacks against Boolean masking schemes.

Let us apply the A.A Ding et al. formula in the case of our two attacks:

$$\mathrm{SR}_{\text{2O-CPA}} = \Phi_{2^n-1}\Big(\sqrt{b}\,\frac{\mathrm{SNR}\left[X^{(3)},M\right]\mathrm{SNR}\left[X^\star,(T,M)\right]}{4}K^{-1/2}\kappa\Big)\ ,$$
$$\mathrm{SR}_{\mathrm{MVA}_{TR}} = \Phi_{2^n-1}\Big(\sqrt{b}\,\frac{\mathrm{SNR}\left[X_{TR},M\right]\mathrm{SNR}\left[X^\star,(T,M)\right]}{4}K^{-1/2}\kappa\Big)\ .$$

We target the same operation for the share that leaks the secret key ($X^\star$). Moreover by remark 4 the informative parts of the leakages depending on the mask (X_{TR} and $X^{(3)}$) is the same in the two leakages. As a consequence, K and κ are the same in the two attacks.

It can be noticed that the only difference in the success rate formula is the use of $\mathrm{SNR}\left[X_{TR},M\right]$ instead of $\mathrm{SNR}\left[X^{(3)},M\right]$. Therefore, it is equivalent to compare these values and compare the SR of these attacks.

Theorem 5.2.1. *The* SNR *of the "second-order leakage" is greater than the* SNR *of the leakage of the mask if and only if*

$$\sigma^2 \leqslant 2^{n-2} - \frac{n}{2}\ ,$$

where σ denotes the standard deviation of the Gaussian noise.

As a consequence MVA_{TR} *will be better than* 2O-CPA *in the interval $\sigma^2 \in [0, 2^{n-2} - n/2]$.*

Proof. See Appendix B.1. Interestingly, the same result is also a byproduct of the demonstration of Proposition 22 (see Appendix B.2.2). □

Theorem 5.2.1 gives us the cases where exploiting the second-order leakage will give better results than exploiting the classical leakage of the mask. For example if $n = 8$ (the case of AES) the second-order leakage is better until $\sigma^2 \leqslant 60$.

Figure 5.2 shows when the SNR of X_{TR} is greater than the SNR of $X^{(3)}$. In order to have a better representation of this interval $1/\text{SNR}$ is plotted.

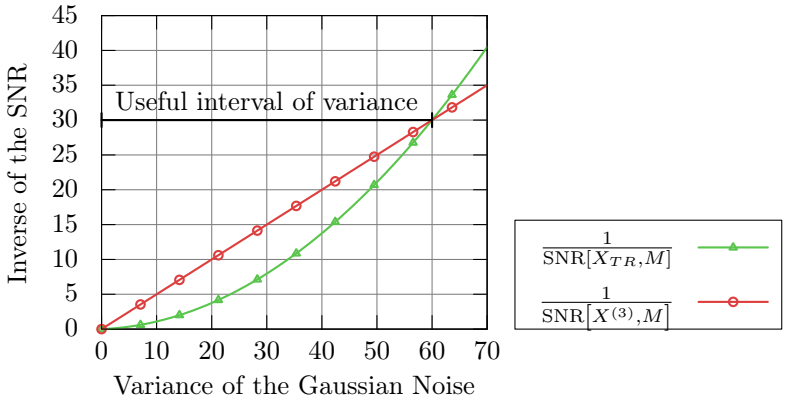


Figure 5.2: Comparison between the variance of the noise for the classical leakage and the second-order and the impact of these noises on the SNR

5.2.4 Simulation results

In order to validate empirically the results of Sect. 5.2, we test the method presented on simulated data. The target is a first order protected AES with table recomputation. To simulate the leakages we assume that each value leaks its Hamming weight with a Gaussian noise of standard deviation σ. The 512 leakages of the table recomputation are those given in Subsect. 5.2.2.

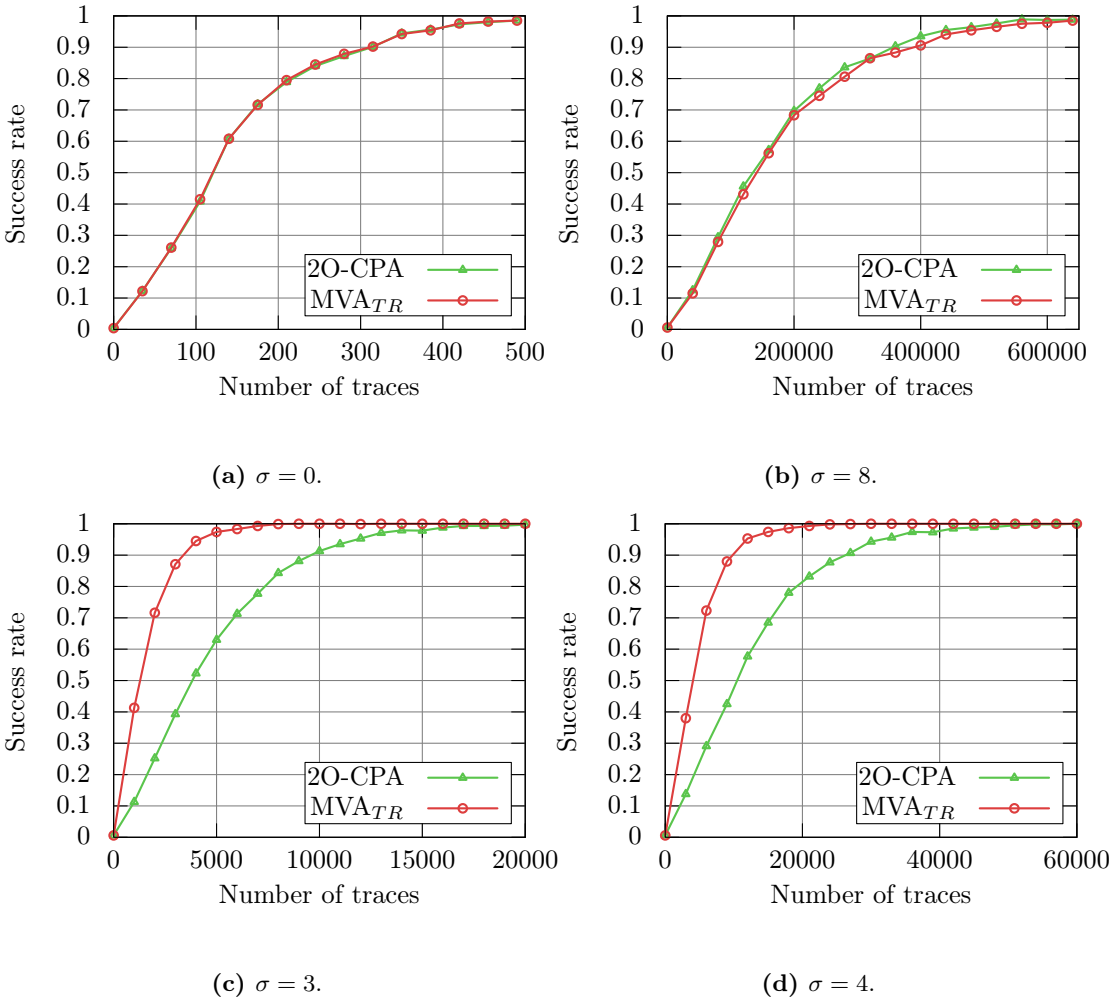
A total of 1000 attacks are realized to compute the success rate of each experiment. In this part, the comparisons are done on the number of traces needed to reach 80% of success.

It can be seen in Fig. 5.3a and in Fig. 5.3b that the difference between the two attacks is null for $\sigma = 0$ and $\sigma = 8$ (that is, $\sigma^2 = 64 \approx 60$). It confirms the bound of the interval shown in Fig. 5.2. This also confirms that comparing the SNR is equivalent to comparing the SR.

It can be seen in Fig. 5.3 that in presence of noise the MVA_{TR} outperforms the 2O-CPA. The highest difference between the MVA_{TR} and 2O-CPA is reached when $\sigma = 3$. In this case, the MVA_{TR} needs 2500 traces to mount the attack while the 2O-CPA needs 7500 traces. This represents a relative gain[1] of $\approx 200\%$. As shown in Fig. 5.3d, the relative gain decreases to 122%

[1]The formal definition of the relative gain is given in Def. 45.

when $\sigma = 4$.



(a) $\sigma = 0$.

(b) $\sigma = 8$.

(c) $\sigma = 3$.

(d) $\sigma = 4$.

Figure 5.3: Comparison between 2O-CPA and MVA$_{TR}$

5.2.5 Theoretical analysis of the SR

While the previous analysis of Subsect. 5.2.3 gives the bounds of effectiveness of the MVA$_{TR}$ it does not allow a quantitative comparison of the respective behaviors of the MVA$_{TR}$ and the 2O-CPA between these bounds. In this subsection we propose an approach which allows a deeper analysis of the relevant parameters of their SR. We exploit the results of (63) which presents a closed form formula which links the SR to the SNR for first order attacks. These results have recently been extended to high order attacks (64).

Proposition 18 ((63, Corollary 1))**.** *The* SR *of an* additive distinguisher *is satisfies:*

$$1 - \mathrm{SR} \approx \exp\left(-\mathrm{SE} \times q\right) \ , \tag{5.11}$$

where SE *is the success exponent and* q *the number of traces used for the attack.*

Proof. The proof is given in (63). □

Proposition 19. *The* SE *of the* 2O-CPA *is:*

$$\mathrm{SE}_{\text{2O-CPA}} = \min_{k \neq k^\star} \frac{\kappa\left(k^\star, k\right)}{2\left(\frac{\kappa'(k^\star,k)}{\kappa(k^\star,k)} - \kappa\left(k^\star, k\right)\right) + 2\left(\alpha_1^{-2}\sigma_1^2 + \alpha_2^{-2}\sigma_2^2 + \alpha_1^{-2}\sigma_1^2\alpha_2^{-2}\sigma_2^2\right)}, \tag{5.12}$$

where in our case (which complies to Eqn. (5.1) *of Definition 41):*

$$\alpha_1^2 = \alpha_2^2 = \mathit{Var}\left[\mathbb{E}\left[X^{(3)}|M\right]\right] = \mathit{Var}\left[\mathbb{E}\left[X^\star|M,T\right]\right] = \sqrt{\frac{n}{4}} \ ,$$
$$\sigma_1^2 = \sigma_2^2 = \mathbb{E}\left[\mathit{Var}\left[X^{(3)}|M\right]\right] = \mathbb{E}\left[\mathit{Var}\left[X^\star|M,T\right]\right] = \sigma^2 \ ,$$

$\kappa\left(k^\star, k\right)$ *and* $\kappa'\left(k^\star, k\right)$ *are general confusion coefficients defined in Definition 8 of (63). Notice that* $\kappa\left(k^\star, k\right)$ *is a natural extension of the seminal coefficient introduced by Fei et al. in (53).*

Proof. See Appendix B.2.1. □

We note that α_i^2 and σ_i^2 respectively represent the power of the signal and of the noise.

As Def. 42, Def. 57 and Eq. (5.8) reveals that the only difference between the MVA_{TR} and the 2O-CPA is the use of X_{TR} instead of $X^{(3)}$. Thus we can directly compute the success exponent of MVA_{TR}.

Proposition 20. *The* SE *of the* MVA_{TR} *is:*

$$\mathrm{SE}_{\mathrm{MVA}_{TR}} = \min_{k \neq k^\star} \frac{\kappa\left(k^\star, k\right)}{2\left(\frac{\kappa'(k^\star,k)}{\kappa(k^\star,k)} - \kappa\left(k^\star, k\right)\right) + 2\left(\alpha_1^{-2}\sigma_1^2 + \alpha_2^{-2}\sigma_2^2 + \alpha_1^{-2}\sigma_1^2\alpha_2^{-2}\sigma_2^2\right)}, \tag{5.13}$$

where in our case

$$\alpha_1^2 = \alpha_2^2 = \mathit{Var}\left[\mathbb{E}\left[X_{TR}|M\right]\right] = \mathit{Var}\left[\mathbb{E}\left[X^\star|M,T\right]\right] = \sqrt{\frac{n}{4}} \ ,$$
$$\sigma_1^2 = \mathbb{E}\left[\mathit{Var}\left[X_{TR}|M\right]\right] = 4 \times \left(\frac{\sigma^2}{2^n} \times \frac{n}{2} + \frac{\sigma^4}{2^n}\right) \ ,$$
$$\sigma_2^2 = \mathbb{E}\left[\mathit{Var}\left[X^\star|M,T\right]\right] = \sigma^2 \ .$$

Proof. The proof is similar as the proof of Prop. 19 using the values of noise computed in the Appendix B.1. □

Exploiting this values it is possible to extract the parameters which impact the respective behavior of the two attacks and especially the ones reaching to a higher difference between the two attacks. Similarly to Subsect. 5.2.4 we will compare the two attacks using the relative gain.

Definition 45 (rel-gain$^{(\mathrm{SR})}$)**.** *The relative gain between* 2O-CPA *and* MVA$_{TR}$ *is given by:*

$$\text{rel-gain}^{(\mathrm{SR})} = \frac{m^{(\mathrm{SR})}_{\text{2O-CPA}} - m^{(\mathrm{SR})}_{\mathrm{MVA}_{TR}}}{m^{(\mathrm{SR})}_{\mathrm{MVA}_{TR}}} \ ,$$

where $m^{(\mathrm{SR})}_{\text{2O-CPA}}$ *and* $m^{(\mathrm{SR})}_{\mathrm{MVA}_{TR}}$ *are respectively the number of traces needed by* 2O-CPA *and* MVA$_{TR}$ *to reach success rate value* SR*.*

And we will also use the difference in number of traces needed to reach SR.

Definition 46 (gain$^{(\mathrm{SR})}$)**.** *The difference in number of traces needed to reach* SR *of success is given by the gain:*

$$\text{gain}^{(\mathrm{SR})} = m^{(\mathrm{SR})}_{\text{2O-CPA}} - m^{(\mathrm{SR})}_{\mathrm{MVA}_{TR}} \ ,$$

where $m^{(\mathrm{SR})}_{\text{2O-CPA}}$ *and* $m^{(\mathrm{SR})}_{\mathrm{MVA}_{TR}}$ *are respectively the number of traces needed by* 2O-CPA *and* MVA$_{TR}$ *to reach* SR *of success rate.*

Notice that rel-gain$^{(\mathrm{SR})}$ and gain$^{(\mathrm{SR})}$ are tools to compare attacks after having computed their SR. They differ from *relative distinguishing margins* metrics (181) which analyses the value of the distinguisher (and not their SR).

Proposition 21. rel-gain$^{(\mathrm{SR})}$ *does not depend on the value of* SR*.*

Proof. See Appendix B.2.2. □

This means that, in Fig. 5.3, the SR curves for 2O-CPA and MVA$_{TR}$ are the same, modulo a scaling in the X axis. For instance, in Fig. 5.3 (a) and (b), the scaling factor is 1, i.e., the two curves superimpose perfectly. As a result, one can compare these two attacks in terms of traces number to extract the key, irrespective of the SR value chosen for the threshold.

Proposition 22. gain$^{(\mathrm{SR})}$ *depends on the value of* SR*, but the value of the noise variance where* gain$^{(\mathrm{SR})}$ *is maximum not depends on* SR*.*

Proof. See Appendix B.2.3. □

Remark 22. *While the bounds of Theorem 5.2.1 depend only on the* SNR *the maximum effectiveness (the maximum of* gain$^{(\mathrm{SR})}$ *or* rel-gain$^{(\mathrm{SR})}$*) of the* MVA$_{TR}$ *compare to the* 2O-CPA *also depends on the operation targets (e.g.* AddRoundKey *or* SubBytes*) by the confusion coefficients* κ *and* κ'*.*

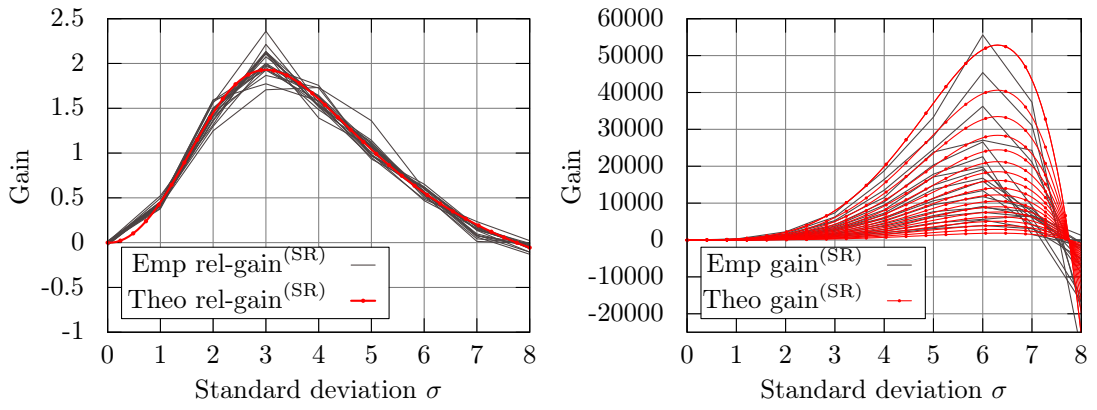
5.2.5.1 Numerical Results.

In order to validate our theoretical analysis we build empirical validation based on simulations. We reuse the curves generated for Sect. 5.2.4. In Fig. 5.4 the empirical results based simulation are plotted in gray and the Theoretical ones in red pointed lines. The first observation is that the theoretical analysis match well the simulations which validates our model choices.

In Fig 5.4a it can be noticed that for several SR (different gray lines) the empirical rel-gain$^{(\mathrm{SR})}$ are closed which confirmed the Prop. 21. Exploiting the formula of Def. 45 we can find the noise variance σ^2 where rel-gain$^{(\mathrm{SR})}$ is maximum. Indeed it occurs in a root of the derivative of rel-gain$^{(\mathrm{SR})}$. In our scenario it occurs for $\sigma^2 = 9.11$ (that is $\sigma \approx 3.02$).

The behavior of gain$^{(\mathrm{SR})}$ is different indeed the SR has an impact on it, the gray lines are not superimposed (see Fig. 5.4b). But similarly to rel-gain$^{(\mathrm{SR})}$ the SR does not impact the value of noise where the maximum gain$^{(\mathrm{SR})}$ is reached. This confirms the Prop. 22. In our scenario it is reached for $\sigma^2 = 39.67$ (that is $\sigma \approx 6.30$).

In order to compute this maximum we have computed the roots of the derivatives (of rel-gain$^{(\mathrm{SR})}$ and gain$^{(\mathrm{SR})}$ w.r.t. σ^2) using the MAXIMA software.



(a) rel-gain$^{(\mathrm{SR})}$ **(b)** gain$^{(\mathrm{SR})}$

Figure 5.4: Comparison between the 2O-CPA and the MVA$_{TR}$

5.3 An example on a high-order countermeasure

The result of the previous section can be extended to any masking scheme based on table recomputation. In particular the MVA$_{TR}$ can apply to High-Order masking schemes.

5.3.1 Coron masking scheme attack and countermeasure

The table recomputation countermeasure can be made secure against High-Order attacks. An approach has been proposed by Schramm and Paar (150). However, it happened that this masking scheme can be defeated by a third order attack (40). To avoid this vulnerability Coron recently presented (39) a new method based on table recomputation, which guarantees a truly high-order masking. The core idea of this method is to mask each output of the S-Box with a different mask and refresh the set of masks between each shift of the table (masking the inputs by one mask). HV attacks are still a threat against such schemes. Indeed an attacker will recover iteratively each input mask. Afterwards he will be able to perform a first order attack on the AddRoundKey to recover the key. To prevent attacks based on the exploitation of the leakages of the input masks an approach based on a random shuffling of the loop index is possible (see Alg. 3). Algorithm 3 is a $(\Omega-1)$-th order countermeasure, meaning that attacks of order strictly less than Ω fail. In this algorithm, the x_i for $i < \Omega$ can be seen indifferently as *shares* or as *masks*. The original masked S-Box algorithm from Coron (39) is the same as Alg. 3, with φ chosen as the identity. It can be noticed that the entropy needed to build the permutation could be low compared to the entropy needed for the masking scheme (especially because of the numerous costly RefreshMasks operations).

5.3.2 Attack on the countermeasure

We apply Alg. 3 on X which is equal to $T \oplus k^\star$, i.e., $\bigoplus_{i=1}^{\Omega} X_i = T \oplus k^\star$. Similarly to the definitions in Subsect. 5.2.2, let us define the leakages of the table recomputation of the masking scheme of Coron where the order of the masking is $\Omega-1$: $X^{(1)}_{(\omega,i,j)} = \mathrm{HW}[\Phi(\omega) \oplus X_i] + N^{(1)}_{(\omega,i,j)} - \frac{n}{2}$ and $X^{(2)}_{(\omega,i,j)} = \mathrm{HW}[\Phi(\omega)] + N^{(2)}_{(\omega,i,j)} - \frac{n}{2}$, where $i \in [\![1, \Omega-1]\!]$ will index the $\Omega-1$ masks. The Ω-th share is the masked sensitive value. Besides $j \in [\![1, \Omega]\!]$ denotes the index of the loop from lines 7 to lines 9 of the Alg. 3. The leakage of the masks is given by $X^{(3)}_i = \mathrm{HW}[X_i] + N^{(3)}_i - \frac{n}{2}$. Finally, we denote by: $X^\star = \mathrm{HW}[\bigoplus_{i=1}^{\Omega-1} X_i \oplus k^\star \oplus T] + N - \frac{n}{2}$ the leakage of the masked value.

Definition 47. *The combination function C^{Ω}_{CS} exploiting the leakage of the table recomputation (Coron Scheme, abridged CS) is given by:*

$$\begin{array}{rccc} \mathrm{C}^{\Omega}_{CS}: & \mathbb{R}^{\Omega\times(\Omega-1)\times 2^{n+1}} \times \mathbb{R} & \to & \mathbb{R} \\ & \left(\left(X^{(1)}_{(\omega,i,j)}, X^{(2)}_{(\omega,i,j)}\right)_{\substack{\omega\in\mathbb{F}_{2^n} \\ i\in[\![1,\Omega-1]\!] \\ j\in[\![1,\Omega]\!]}}, X^\star\right) & \mapsto & \displaystyle\prod_{i=1}^{\Omega-1}\left(\frac{-2}{\Omega 2^n}\sum_{\substack{\omega\in\mathbb{F}_{2^n} \\ j\in[\![1,\Omega]\!]}} X^{(1)}_{(\omega,i,j)} \times X^{(2)}_{(\omega,i,j)}\right) \times X^\star. \end{array}$$

Algorithm 3: Masked and shuffled computation of $y = S(x)$

input : $x_1, \ldots, x_\Omega$, such that $x = x_1 \oplus \ldots \oplus x_\Omega$
output: $y_1, \ldots, y_\Omega$, such that $y = y_1 \oplus \ldots \oplus y_\Omega = S(x)$

$\varphi \leftarrow_{\mathcal{R}} \mathbb{F}_2^n \to \mathbb{F}_2^n$ // Draw of random permutation of $\mathbb{F}_2^n$
for $\omega \in \mathbb{F}_2^n$ **do**
  $\mathcal{T}(\omega) \leftarrow (S(\omega), 0, \ldots, 0) \in (\mathbb{F}_2^n)^d$ // $\oplus(\mathcal{T}(\omega)) = S(\omega)$
end
for $i = 1$ **to** $\Omega - 1$ **do**
  for $\omega \in \mathbb{F}_2^n$ **do**
    for $j = 1$ **to** Ω **do**
      $\mathcal{T}'(\varphi(\omega))[j] \leftarrow \mathcal{T}(\varphi(\omega) \oplus x_i)[j]$ // $\mathcal{T}'(\varphi(\omega)) \leftarrow \mathcal{T}(\varphi(\omega) \oplus x_i)$
    end
  end
  for $\omega \in \mathbb{F}_2^n$ **do**
    $\mathcal{T}(\varphi(\omega)) \leftarrow \mathsf{RefreshMasks}(\mathcal{T}'(\varphi(\omega)))$ // See in Alg. 2 of (39)
  end
end
// Invariant: $\oplus(\mathcal{T}(\varphi(\omega))) = S(\varphi(\omega) \oplus x_1 \oplus \ldots \oplus x_{\Omega-1}), \ \forall \omega \in \mathbb{F}_2^n$
$(y_1, \ldots, y_\Omega) \leftarrow \mathsf{RefreshMasks}(\mathcal{T}(x_\Omega))$ // $\oplus(\mathcal{T}(x_\Omega)) = S(x)$
return $y_1, \ldots, y_\Omega$

Similarly to Subsect. 5.2.3, we define for all $1 \leqslant i \leqslant \Omega - 1$:

$$X_{CS_i^\Omega} = \frac{-2}{\Omega 2^n} \sum_{\substack{\omega \in \mathbb{F}_{2^n} \\ j \in [\![1,\Omega]\!]}} X^{(1)}_{(\omega,i,j)} \times X^{(2)}_{(\omega,i,j)} \ .$$

This value is the combination of all the leaking values of the table recomputation depending of one share.

Remark 23. *The scaling by factor $-2/\Omega$ allows to have, for all $i \in [\![1, \Omega - 1]\!]$:*

$$\mathbb{E}\left[X_{CS_i^\Omega} | X_i = x_i\right] = \mathbb{E}\left[X_i^{(3)} | X_i = x_i\right] \ .$$

Additionally we define for, $i = \Omega$, $X_{CS_i^\Omega} = X^\star$. Based on the combination function C^Ω_{CS} a multivariate attack can be built.

Definition 48. *The MultiVariate Attack exploiting the leakage of the table recomputation of the $\Omega - 1$ order Coron masking Scheme is given by:*

$$\begin{array}{rccc} \mathrm{MVA}^\Omega_{CS}: & \mathbb{R}^{\Omega\times(\Omega-1)\times 2^{n+1}} \times \mathbb{R} \times \mathbb{R} & \to & \mathbb{F}_2^n \\ & \left(\left(X^{(1)}_{(\omega,i,j)}, X^{(2)}_{(\omega,i,j)}\right)_{\substack{\omega\in\mathbb{F}_{2^n} \\ i\in[\![1,\Omega-1]\!] \\ j\in[\![1,\Omega]\!]}}, X^\star, Y\right) & \mapsto & \underset{k\in\mathbb{F}_2^n}{\operatorname{argmax}}\, \rho\left[\prod_{i=1}^{\Omega}\left(X_{CS_i^\Omega}\right), Y\right] \ , \end{array}$$

where $Y = (-1)^{\Omega-1} \times \left(\mathrm{HW}[T \oplus k] - \frac{n}{2}\right)$.

Proposition 23. MVA^Ω_{CS} *is sound.*

Proof. The demonstration follows the same lines as that of Proposition 17. In the case of Proposition 23, the expectation of $\prod_{i=1}^{\Omega}\left(X_{CS_i^\Omega}\right)$ knowing the plaintext $T = t$ is proportional to $\mathrm{HW}[t \oplus k]$. Indeed by (142) $\mathbb{E}\left[\prod_{i=1}^{\Omega}\left(X_{CS_i^\Omega}\right) | T = t\right] = \left(\frac{-1}{2}\right)^{\Omega-1} \times \left(\mathrm{HW}[t \oplus k] - \frac{n}{2}\right)$ □

Remark 24. *The attack presented in Def. 48 is a $(\Omega\times(\Omega-1)\times 2^{n+1}+1)$-variate $(2\times(\Omega-1)+1)$-order attack.*

Definition 49. *The "classical"* ΩO-CPA *is the* HOCPA *build by combining the Ω shares using the centered product combination function.*

$$\begin{array}{rccc} \Omega\text{O-CPA}: & \mathbb{R}^{\Omega-1} \times \mathbb{R} \times \mathbb{R} & \longrightarrow & \mathbb{F}_2^n \\ & \left(\left(X_i^{(3)}\right)_{i\in[\![1,\Omega-1]\!]}, X^\star, Y\right) & \longmapsto & \underset{k\in\mathbb{F}_2^n}{\operatorname{argmax}}\, \rho\left[\prod_{i=1}^{\Omega-1} X_i^{(3)} \times X^\star, Y\right] \ . \end{array}$$

5.3.3 Leakage analysis

The difference between the two attacks is the use of $X_{CS_i^\Omega}$ instead of $X_i^{(3)}$ as the leakage of the $\Omega - 1$ shares which do not leak the secret key. A.A Ding et al. also provides a formula to compute the SR of HOCPA (46, §3.4).

Similarly to Sect. 5.2, the only differences in the formula are the SNR of the shares which do not leak the key. Then by comparing the SNR $\left[X_{CS_i^\Omega}, X_i\right]$ and SNR $\left[X_i^{(3)}, X_i\right]$ we compare the success rate of the attacks. It can be noticed that in our model the SNR does not depend on i.

Theorem 5.3.1. *The* SNR *of the "second-order leakage" is greater than the* SNR *of the leakage of the mask if and only if*

$$\sigma^2 \leqslant \Omega \times 2^{n-2} - \frac{n}{2} \ , \tag{5.14}$$

where σ denotes the standard deviation of the Gaussian noise.

As a consequence $\mathrm{MVA}_{CS}^{\Omega}$ *will be better than* ΩO-CPA *when the noise variance lays in the interval* $[0, \Omega \times 2^{n-2} - n/2]$. *We can immediately deduce that the size of the Useful Interval of Variance increases linearly with the order of the masking scheme.*

Proof. See Appendix B.3. □

Figure 5.5 shows the impact of the attack order Ω on the interval of noise where the $\mathrm{MVA}_{CS}^{\Omega}$ outperforms ΩO-CPA (let us called this interval the Useful Interval of Variance denoted by UIoV). We can see that the size of these intervals increases with the order. For example for $\Omega = 3$ the useful interval of variance is $[0, 188]$. In practice, it is very difficult to perform a third order attack with a noise variance of 188. Indeed, recall that the number of traces to succeed an attack with probability 80% is proportional to the inverse of the SNR (63).

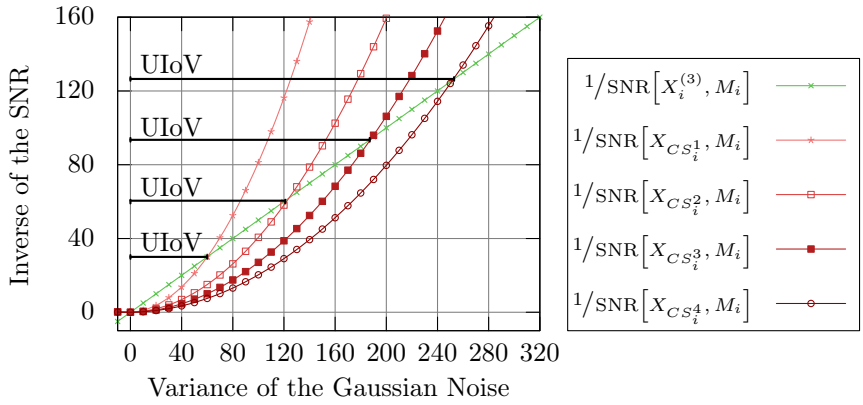


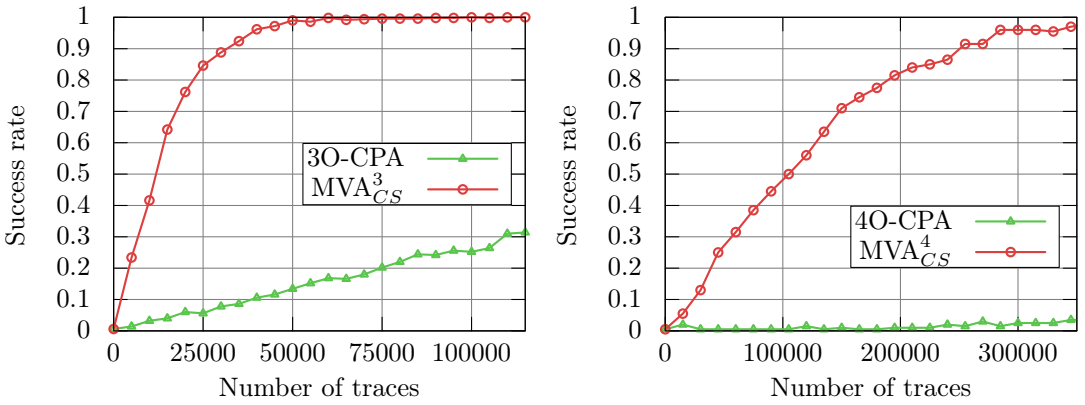
Figure 5.5: Comparison between the signal to noise ratio of $X_i^{(3)}$ and signal to noise ratio of $X_{CS_i^\Omega}$ (where Ω is the attack order).

5.3.4 Simulation results on Coron masking Scheme

In order to validate the theoretical results of Subsect. 5.3.3, the MVA_{CS}^{Ω} has been tested on simulated data and compared to ΩO-CPA. The simulations have been done with the Hamming weight model and Gaussian noise such as the leakages defined in Subsect. 5.3.2. We test these attacks against a second and a third order masking scheme.

To compute the success rate, attacks are redone 500 times for the second order masking and 100 times for the third order masking (because this attack requires an intensive computational power).

In Fig. 5.6a it can be seen that $\text{MVA}_{CS}^{(3)}$ reaches 80% of success rate for less than 20000 traces while the 3O-CPA does not reach 30% for 100000. In Fig. 5.6b it can be seen that $\text{MVA}_{CS}^{(4)}$ reaches 80% of success rate for less than 200000 traces while the 4O-CPA does not reach 5%.



(a) $\Omega = 3, \sigma = 3$. **(b)** $\Omega = 4, \sigma = 3$.

Figure 5.6: Comparison between ΩO-CPA and MVA_{CS}^{Ω}

5.4 A note on affine model

In Sect. 5.2 and 5.3, the leakage function was expected to be the Hamming weight. Let us now study the impact of the leakage function on the MVA_{TR} attack. We suppose that the leakage function is affine i.e. we expecte that the leakage function is the Weighted Hamming Weight.

5.4.1 Properties of the affine model

Definition 50 (Affine leakage function)**.** *Let V the leaking value, α the weight of the leakage of each bit, and $\cdot$ the inner product in $\mathbb{R}^n$, that is $\alpha \cdot V = \sum_{i=1}^{n} \alpha_i V_i$. A leakage function Ψ_α is said affine if this function is a weighted sum of the bits of the leaking value, i.e., $\Psi_\alpha(V) = \alpha \cdot V$.*

In the sequel, we assume sensitive variables are balanced and have each bit independent of the other, as is customary in cryptographic applications.

Proposition 24. *Let $\mathbf{1} = (1, \ldots, 1) \in \mathbb{F}_2^n$.*

$$\mathbb{E}[\Psi_\alpha[V]] = \frac{1}{2}(\alpha \cdot \mathbf{1}) \qquad \textit{and} \qquad \mathit{Var}[\Psi_\alpha[V]] = \frac{1}{4}\|\alpha\|_2^2 \ .$$

Proof. We have $\mathbb{E}[\Psi_\alpha[V]] = \alpha \cdot \mathbb{E}[V] = \alpha \cdot \left(\frac{1}{2}\mathbf{1}\right)$ and $\mathrm{Var}[\Psi_\alpha[V]] = \alpha^{\mathsf{t}}\mathsf{Cov}[V]\,\alpha = \frac{1}{4}\|\alpha\|_2^2$. □

Then it is possible to compute the results of the centered product.

Lemma 5. *Let U be a random variable following a uniform law over $\mathbb{F}_2^n$, and $z \in \mathbb{F}_2^n$. We have:*

$$\mathbb{E}[(\Psi_\alpha[U] - \mathbb{E}[\Psi_\alpha[U]]) \times (\Psi_\beta[U \oplus z] - \mathbb{E}[\Psi_\beta[U \oplus z]])] = -\frac{1}{2}(\alpha \odot \beta) \cdot z + \frac{1}{4}\alpha \cdot \beta \ ,$$

where $\odot$ denotes the element-wise multiplication, that is $(\alpha \odot \beta)_i = \alpha_i \beta_i$.

Proof. See in Appendix B.4.1. □

Assumption 1. *In order to compare the results in case of an affine model and the Hamming weight model ($\mathsf{HW} = \Psi_{\mathbf{1}}$) let us assume that the model variance is the same in the two cases i.e., $\mathit{Var}[\Psi_\alpha(V)] = \mathit{Var}[\mathrm{HW}[V]]$; this is equivalent to $\|\alpha\|_2^2 = n$.*

Let us also assume that all the values manipulated during the algorithm leak in the same way i.e., the weight vector α of the sum is the same for all the variables V of the algorithm. This is realistic because it is likely that sensitive variables transit through a given resource, e.g., the accumulator register.

In the rest of this section, we will denote by α the vector of weight of the leakage model.

Let us redefine the leakage of the table recomputation the (centered) leakage of the random index: $X_\omega^{(1)} = \alpha \cdot (\Phi(\omega) \oplus M) + N_\omega^{(1)} - \frac{1}{2}(\alpha \cdot \mathbf{1})$, the (centered) leakage of the mask random index: $X_\omega^{(2)} = \alpha \cdot (\Phi(\omega)) + N_\omega^{(2)} - \frac{1}{2}(\alpha \cdot \mathbf{1})$, the (centered) leakage of the mask: $X^{(3)} = \alpha \cdot M - \frac{1}{2}(\alpha \cdot \mathbf{1})$, Besides, let $X^\star$ be the leakage of a sensitive value depending on the key. We have either:

- $X^\star = \alpha \cdot (T \oplus k^\star \oplus M) + N - \frac{1}{2}(\alpha \cdot \mathbf{1})$, which is similar to Eq. (5.7), or
- $X^\star = \alpha \cdot (S(T \oplus k^\star) \oplus M) + N - \frac{1}{2}(\alpha \cdot \mathbf{1})$, if there is an S-Box S.

In a view to unite both expressions, we denote by Z the sensitive variable, that is either $Z = T \oplus k^\star$, or $Z = S(T \oplus k^\star)$. Consequently, we have $X^\star = \alpha \cdot (Z \oplus M) + N - \frac{1}{2}(\alpha \cdot \mathbf{1})$.

Lemma 6. *In case of affine leakage model the second order leakage X_{TR} is given by:*

$$\mathbb{E}\left[X_{TR}|M=m\right] = \mathbb{E}\left[\frac{-2}{2^n}\sum_{\omega=0}^{2^n-1} X_\omega^{(1)} \times X_\omega^{(2)} \mid M=m\right] = \left(\alpha^2\right)\cdot m - \frac{1}{2}\|\alpha\|_2^2 \ ,$$

where $\alpha^2 = \alpha \odot \alpha$.

Proof. Direct application of Lemma 5. □

Proposition 25. *In case of affine model, the leakages of the* MVA_{TR} *(recall Def. 42) and the 2O-CPA are different. Indeed, let us denote $\alpha^n = \underbrace{\alpha \odot \alpha \odot \cdots \odot \alpha}_{n \text{ times}}$. We have:*

$$\mathbb{E}\left[\mathrm{C}_{TR}\left(\left(X_\omega^{(1)}, X_\omega^{(2)}\right)_\omega, X^\star\right) \mid T\right] = -\frac{1}{2}\alpha^3\cdot z + \frac{1}{4}\sum_{i=1}^{n}\alpha_i^3 \ ,$$

and

$$\mathbb{E}\left[X^{(3)} \times X^\star \mid T\right] = -\frac{1}{2}\alpha^2\cdot z + \frac{1}{4}\|\alpha\|_2^2 \ .$$

Proof. Direct application of Lemma 6 and Lemma 5. □

5.4.2 Impact of the model on the confusion coefficient

As the models in the two different attacks are different, the parameters K and κ (recall Eq. (5.10)) also differ. In order to compare the two attacks we first establish the impact of the model on the value of the minimum confusion coefficient $\min_{k\neq 0}\kappa_k$. Then we show that the impact is not important in case of the targeted sensitive value is proceed in a nonlinear part of the algorithm (an S-Box).

In practice the confusion coefficients are very close. We study the impact of the disparity of α using several distributions (see Fig. 5.7):

- $\alpha_i = \sqrt{1+\varepsilon}$ for i even and $\alpha_i = \sqrt{1-\varepsilon}$ otherwise (abridged $\alpha = \sqrt{1\pm\varepsilon}$),
- and the other sign convention (abridged $\alpha = \sqrt{1\mp\varepsilon}$).

We also randomly generate 1000 α. All those distributions satisfy the assumption 1, namely $\sum_{i=1}^{n}\alpha_i^2 = n$.

The confusion coefficient for α^2 and α^3 are very close (see Fig. 5.7).

Moreover we find that the maximum difference in all the simulations with random weight is $\max\left(\min_{k\neq 0\ \alpha^2}\kappa_k - \min_{k\neq 0\ \alpha^3}\kappa_k\right) = 0.019$. In terms of number of traces needed to reach 80% of success this represents a small difference of 5%.

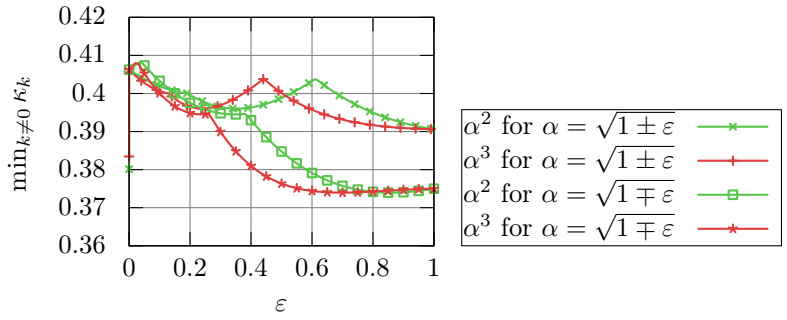


Figure 5.7: Comparison of $\min_{k\neq 0} \kappa_k$ for the MVA_{TR} and the 2O-CPA

5.4.3 Theoretical analysis

Similarly to the Subsect. 5.2.3 let us study the impact of the affine model on the success of the MVA_{TR} compared to the 2O-CPA.

As motivated in Sect. 5.2.1, we can modify the MVA_{TR} in order to target the last round S-Box input: $X^\star = \alpha \cdot \left(\texttt{Sbox}^{-1}[T \oplus k^\star] \oplus M\right) + N - \frac{1}{2}(\alpha \cdot \mathbf{1})$.

Theorem 5.4.1. *The* SNR *of the "second-order leakage" is greater than the* SNR *of the leakage of the mask if and only if*

$$\sigma^2 \leqslant \|\alpha\|_4^4 \times \frac{2^{n-2}}{n} - \frac{n}{2} \ ,$$

where $\|\alpha\|_p = (\sum_{i=1}^n |\alpha_i|^p)^{1/p}$ *is the p-norm* ($p \geqslant 1$) *of vector* α*, and where* σ *denotes the standard deviation of the Gaussian noise.*

As a consequence MVA_{TR} *is better than* 2O-CPA *when the noise variance is in the interval* $[0, \|\alpha\|_4^4\, 2^{n-2}/n - n/2]$.

Proof. See Appendix B.4.2. □

Corollary 6. *The minimal value of* $\|\alpha\|_4^4$ *subject to* $\|\alpha\|_2^2 = n$ *is reached when all the component of* α *are equal. This means that the worst case for the* MVA_{TR} *compared to the* 2O-CPA *is when the leakage is in Hamming Weight.*

Proof. See Appendix B.4.3. □

5.4.4 Simulation results

Some simulations have been done in order to validate the results of the theoretical study of the previous sections. The results, presented in this section, confirm that:

- attacks are not impacted by the small differences of the confusion coefficient (κ, recall Sec. 5.4.2).

- attacks depend on the SNR as predicted by Theorem 5.4.1.

For the purpose of the simulations, the target considered is the input of the S-Box of the last round; as a consequence we consider

$$X^\star = \alpha \cdot \left(\texttt{Sbox}^{-1}[T \oplus k^\star] \oplus M\right) + N - \frac{1}{2}\left(\alpha \cdot \mathbf{1}\right) \quad .$$

The mask M and the plain text T are randomly drawn from $\mathbb{F}_2^8$. The noises are drawn from a Gaussian distribution with different variances σ^2. The results of the attacks are expressed using the success rate. To compute the success rates the experiments have been redone 1000 times. For each experiment the secret key $k^\star$ are randomly drawn over $\mathbb{F}_2^8$. To compare the efficiency of the two attacks we compare the number of traces needed to reach 80% of success.

For the first experiment we choose $\alpha = \sqrt{1 \pm \varepsilon}$ (i.e., $\forall i$, $\alpha_i = \sqrt{1 + (-1)^i \varepsilon}$).

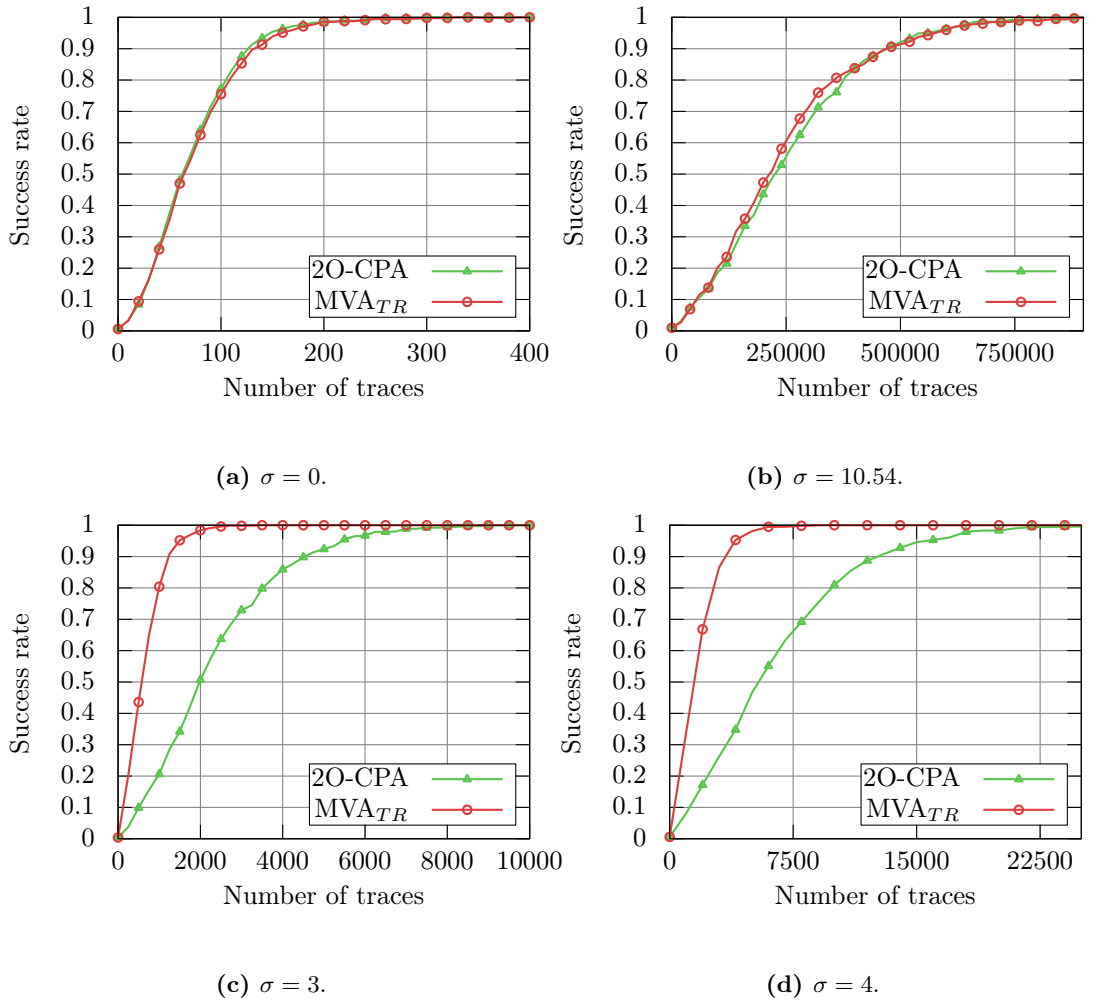
5.4.4.1 Case $\varepsilon = 0.9$

In this case $\|\alpha\|_4^4 = 14.480$ and according to Theorem 5.4.1, the MVA_{TR} should outperform the classical success rate in the interval $[0, 111]$. It can be seen in Fig. 5.8a and 5.8b that in such case when $\sigma^2 = 0$ or when $\sigma^2 = 111$ the MVA_{TR} and the 2O-CPA need the same number of traces to reach 80% of success. First of all, this confirms the soundness of our model. Second, it validates that, in case of affine model when the target is proceeded in a non linear part of the cryptographic algorithm, the main factor which makes attacks different is the SNR. When $\sigma = 3$ the 2O-CPA needs around 3800 traces to reach 80% of success whereas the MVA_{TR} needs around 1000 traces (see Fig. 5.8c). This represents a relative gain of 280%. Compared to the relative gain observed in case of the Hamming weight model (recall Fig. 5.3c), this confirms that the MVA_{TR} performs better compare to the 2O-CPA in case of an affine model. It can be seen in Fig. 5.8d, when the $\sigma = 4$, the number of traces needed to reach 80% of success is around 2500 for the MVA_{TR} and around 10000 for the 2O-CPA; this represents a relative gain of 300%.

5.4.4.2 Case $\varepsilon = 0.5$

When $\varepsilon = 0.5$, $\|\alpha\|_4^4 = 10$; consequently, Theorem 5.4.1 predicts that the MVA_{TR} should outperform 2O-CPA in the interval $[0, 76]$. It can be seen in Figure 5.9a and 5.9b that in such case when $\sigma^2 = 0$ or when $\sigma^2 = 76$ the MVA_{TR} and the 2O-CPA need the same number of traces to reach 80% of is success. This confirms the results of Theorem 5.4.1.

It can be seen in Fig. 5.9c that when $\sigma = 3$ the MVA_{TR} needs around 1000 traces to reach 80% of success whereas the 2O-CPA needs 3500 traces. The relative gain of use the MVA_{TR} is



(a) $\sigma = 0$.

(b) $\sigma = 10.54$.

(c) $\sigma = 3$.

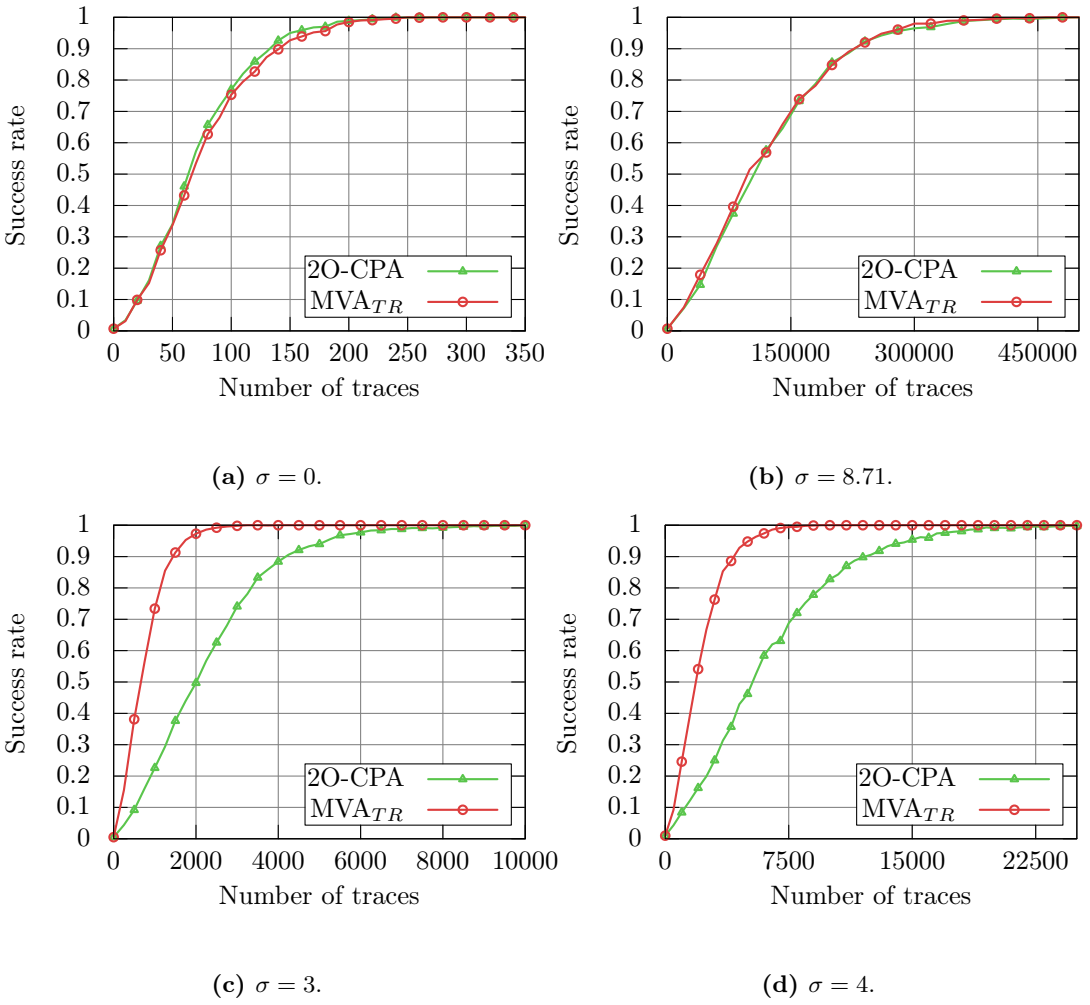
(d) $\sigma = 4$.

Figure 5.8: Comparison between 2O-CPA and MVA$_{TR}$ for $\varepsilon = 0.9$

250%. When $\sigma = 4$ then the number traces needed by the MVA$_{TR}$ to reach 80% of success is around 3000. The number of traces needed by the 2O-CPA is around 9000. The relative gain of the MVA$_{TR}$ with respect to the 2O-CPA is 200%.

5.4.4.3 For one bit attacks

The best case for MVA$_{TR}$ compared to the 2O-CPA is when all the bits are zero except one (see Appendix B.4.3). Let us compare the two attacks in a such case. We assume that all the coordinates of α are equal to zero except the most significant bit. As $\|\alpha\|_4^4 = 64$ the Useful Interval of Variance is $[0, 508]$. It can bee see in Fig.5.10a that when the noise is null both



(a) $\sigma = 0$.

(b) $\sigma = 8.71$.

(c) $\sigma = 3$.

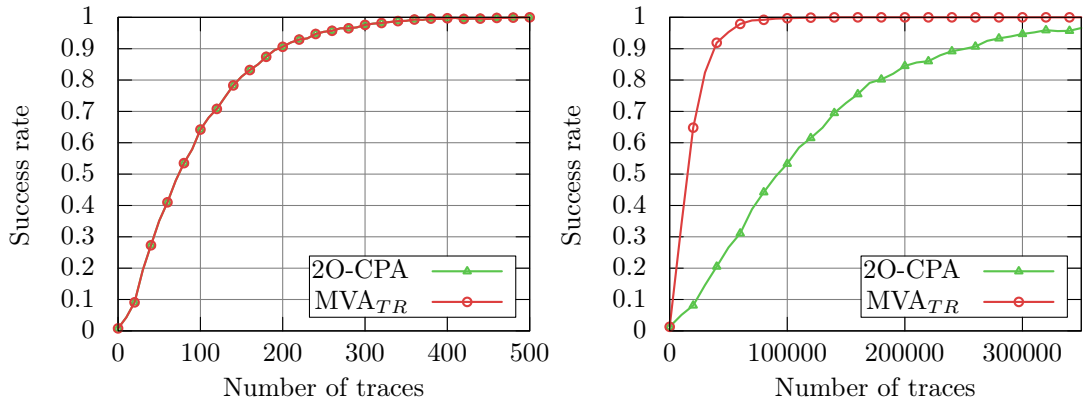
(d) $\sigma = 4$.

Figure 5.9: Comparison between 2O-CPA and MVA$_{TR}$ for $\varepsilon = 0.5$

attacks perform in the same way. It confirms that also in this case the difference resides in the SNR. When $\sigma = 8$ the MVA$_{TR}$ reach 80% of success with 25000 traces whereas the 2O-CPA needs 175000; this represents a relative gain of 600% (see Fig. 5.10b).

5.5 Practical validation

This section presents the results of the multivariate attack exploiting the table recomputation stage on true traces.



(a) $\sigma = 0$. **(b)** $\sigma = 8$.

Figure 5.10: Comparison between the 2O-CPA and the MVA$_{TR}$ in case of one bit model in presence of High Gaussian noise

5.5.1 Experimental Setup

The traces are electromagnetic leakages of the execution of an AES-128 assembly implementation with table recomputation. Our implementation has been loaded on ATMEL ATMega163 8-bit to be analyzed. This smartcard is known to be leaky. It contains 16Kb of in-system programmable flash, 512 bytes of EEPROM, 1Kb of internal SRAM and 32 general purpose working registers. The smartcard is controlled by a computer through the Xilinx Spartan-VI FPGA embedded in a SASEBO-W platform. The ATMega is powered at 2.5 V and clocked at 3.57 MHz.

The measurements were taken using a LeCroy wave-runner 6100A oscilloscope by means of a Langer EMV 0–3 GHz EM probe and PA-303 30 dB Langer amplifier. The acquisition have been acquired with full bandwidth and with a sampling rate of $F_S = 500$ MS/s.

To build our experiments 13000 traces have been acquired. Each trace contains 12 million leakages samples in order to simplify our analysis we only acquired the table recomputation step and the first round of the AES.

5.5.2 Experimental results

Let us first study the results of the attack in terms of success rate. The leakage function as been recovered using a linear regression. For example the normalized vector of weight for the leakage of the first share is

$$\alpha = (0.95, 1.22, 0.98, 1.13, 0.59, 1.01, 1.04, 0.95) \ .$$

Both the MVA_{TR} and the 2O-CPA target $\texttt{Sbox}[T \oplus k^\star] \oplus M$ as in our implementation the input and output masks are the same.

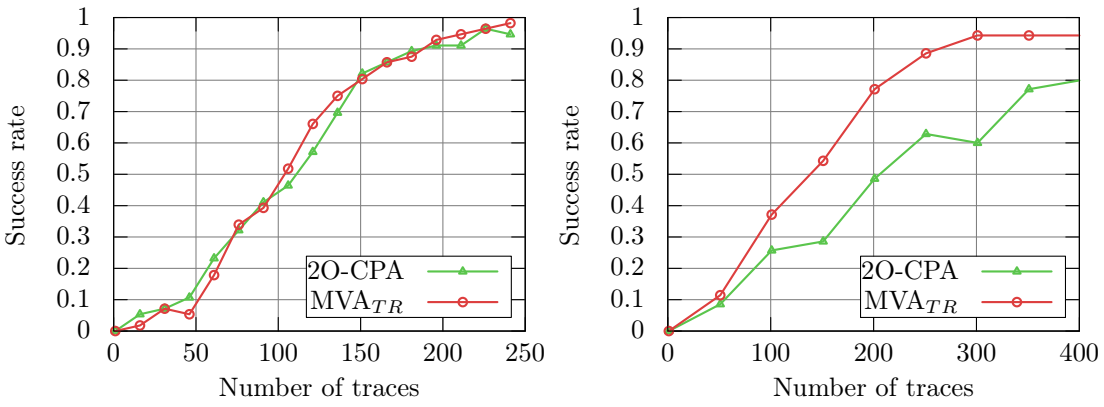
It can be seen in Fig. 5.11a that the results of the two attacks are similar. Both attacks perform similarly because the curves are not noisy.

Indeed the average values of the SNR of the 256 leakages of the masked random index $(\Phi(\omega) \oplus M)$ and the SNR of the 256 leakages of the random index $(\Phi(\omega))$ is 5.

If we assume that the variance of the signal is equal to two (such as HW on 8-bit CPUs) then the variance of the noise is less than 0.5. The mask (M) and the key-dependent share $\texttt{Sbox}[T \oplus k^\star] \oplus M)$ leak with a SNR of 14 which corresponds to a noise variance of 0.1, which is very low (compared to the upper bound of the useful interval of variance given in Theorem 5.2.1, namely 60).

This two results are specific to the implementation and a clear disadvantage for the MVA_{TR}. But even in this case the MVA_{TR} works as well as the 2O-CPA, this shows that there is (generally) a gain to use the MVA_{TR}.

In order to confirm these results let us verify that when the noise increases the MVA_{TR} outperforms the 2O-CPA. Let us add an artificial Gaussian noise with a standard deviation of 0.0040. This models the addition of a countermeasure on top of the table recomputation. Then it can be seen in Fig. 5.11b that in this case the MVA_{TR} outperforms the 2O-CPA. This confirms the practicality of our attack, and also that the gain is in the SNR.



(a) Comparison on raw traces

(b) Comparison with noise addition

Figure 5.11: Comparison of the SR of the MVA_{TR} and the 2O-CPA

5.6 Countermeasure.

The MVA_{TR} represents a threat against block ciphers with table recomputation step. In order to mitigate this new vulnerability we present in this section a countermeasure. This countermeasure will ensure the security against the new proposed attack. We present it in the context of a first order masking scheme but this countermeasure is generic and as a consequence can be applied in a higher order masking scheme such as the masking scheme of Coron.

Remark 25. *The proposed countermeasure tackles the input masks vulnerability. The protection of the output mask is easier as all the output masks can be different for all the table entries.*

5.6.1 Countermeasure Principle

The core idea of this countermeasure is to randomly drawn permutations not all over the possible permutations but only over a particular kind of permutations: the ones which are commutative with S (the SubBytes function).

Definition 51. *A permutation $f : \mathbb{F}_2^n \to \mathbb{F}_2^n$ is said commutative with respect to the function $g : \mathbb{F}_2^n \to \mathbb{F}_2^n$ and the composition law if and only if $f(g(x)) = g(f(x)), \forall x \in \mathbb{F}_2^n$.*

Exploiting this kind of function the countermeasure principle is as follow: as random permutation a commutative permutation with respect to S is drawn. Let us call the permutation γ. Exploiting the commutative property of the random permutation $\gamma(S[\omega])$ is computed instead of $S[\gamma(\omega)]$ (line 5 of Alg. 4). Contrast this line with line 5 of Alg. 2. As a consequence if an attacker combines the leakages of the random mask index (line 4) and the random index (line 5) the obtained value depends very little in the masks m and m' (see analysis in Sec. 5.6.3).

5.6.2 Implementations

The major issue of the countermeasure in an implementation perspective is to randomly generate a commutative permutation.

A first approach could be to generate off line a set of permutations and store them into the device. At each execution using a random number, a permutation will be selected. Of course such approach can be prohibitive in terms of memory.

A probably better approach is to generate on the fly a commutative permutation. In this subsection we give an example of a such algorithm. The idea is to randomly generate a power (with respect to the combination law) of the SubBytes : S function.

Algorithm 4: Shuffled masked table recomputation, with additional countermeasure

input : Genuine SubBytes $S : \mathbb{F}_2^n \to \mathbb{F}_2^n$ bijection

output : Masked SubBytes $S' : \mathbb{F}_2^n \to \mathbb{F}_2^n$ bijection

1 $m \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$, $m' \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$ // Draw of random input and output masks

2 $\varphi \leftarrow_{\mathcal{R}} \mathbb{F}_2^n \to \mathbb{F}_2^n$, $\gamma \leftarrow_{\mathcal{R}} \mathbb{F}_2^n \to \mathbb{F}_2^n$ such that $\gamma \circ S = S \circ \gamma$ // Draw of random permutations φ, γ of $\mathbb{F}_2^n$, γ permuting with S (see Definition 51)

3 **for** $\omega \in \{\varphi(0), \varphi(1), \ldots, \varphi(2^n - 1)\}$ **do** // S-Box recomputation loop

4 $\quad z \leftarrow \gamma(\omega) \oplus m$ // Masked input

5 $\quad z' \leftarrow \gamma(S[\omega]) \oplus m'$ // Masked output

6 $\quad S'[z] = z'$ // Creating the masked S-Box entry

7 **end**

8 **return** S'

Definition 52. *The power $p \in \mathbb{N}$ of the function S is given by:*

$$\begin{array}{rccc} S^p : & \mathbb{F}_2^n & \longrightarrow & \mathbb{F}_2^n \\ & x & \longmapsto & \underbrace{S \circ S \circ \ldots \circ S}_{p}(x)\,, \end{array}$$

where $\circ$ denotes the composition law.

Proposition 26. *The functions $S^p : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^n$ and $S : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^n$ are commutative $\forall p \in \mathbb{N}$.*

In order to generate a random power of S it is possible to directly compute S^r by applying r times the permutation S where r is a random number. Notice that r can be larger than the number of possible power S by the group law property of the combination. But this approach can be time consumptive.

To faster it, the use of the cycle decomposition of S may be an interesting approach. Let us recall this well known theorem:

Proposition 27 (Theorem 5.19 (43))**.** *Let S_n be the symmetric group of n elements then each element of S_n can be expressed as a product of disjoint cycles.*

Proposition 28. *The maximum number of exponentiations needed to compute S^p could be reduced from p to $p \pmod{l_1} + p \pmod{l_2} + ... + p \pmod{l_m}$ where the l_i denote the respective length of the cycles in the cycles decomposition of S. Notice that $l_1 + l_2 + \ldots + l_m = 2^n$*

Proof. We can express S as $S = c_1 \circ c_2 \circ \ldots \circ c_m$ by Prop. 27. As the order of a cycle is equal to its length l we have that:

$$S^p = c_1^{p \pmod{l_1}} \circ c_2^{p \pmod{l_2}} ... \circ c_m^{p \pmod{l_m}}$$

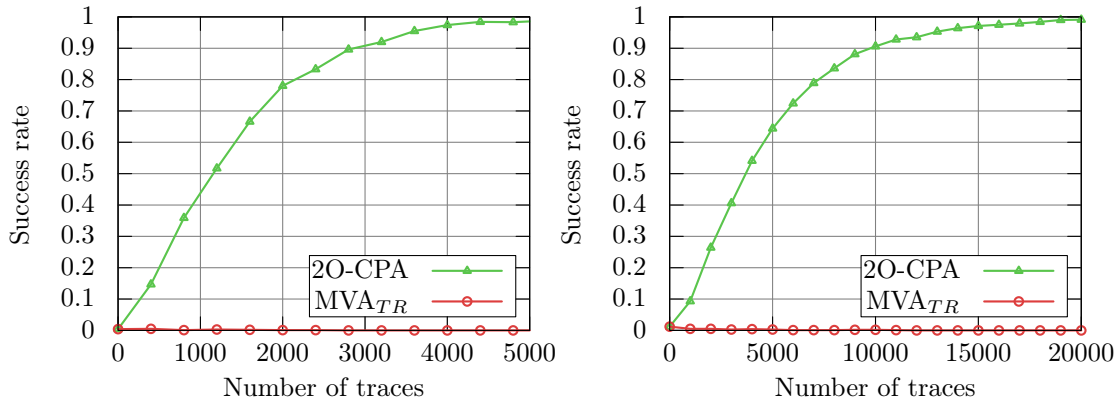
□

Example 4. *Let us take as example of S the* SubBytes *function of AES. This permutation can be decomposed of on five disjoint cycles of respectively length $l_1 = 59$, $l_2 = 81$, $l_3 = 87$, $l_4 = 27$, $l_5 = 2$. The order of S is in this case is* $\mathrm{lcm}(59, 87, 81, 27, 2) = 277182$. *As a consequence the computation of S^{277182} requires a maximum of* 256 *table evaluations.*

5.6.3 Security Analysis

The security provided by this countermeasure comes from different parameters. Of course the first one is to ensure that the MVA_{TR} is not still available or at least less effective than the 2O-CPA which remains available. We validated this security using simulation with the same set up as in Subsect. 5.2.4. Namely we assume that each value leaks its Hamming weight with a Gaussian noise of standard deviation σ. A total of 1000 attacks has been realized to compute the success rate of each experiment.

The attacker can combine $\gamma(S[\omega])$ with $\gamma(\omega)$. The results of this combination can be found in Fig. 5.12 for two different noise standard deviations. We can immediately see that in this case the MVA_{TR} does not allow to recover the key.



(a) $\sigma = 2$. **(b)** $\sigma = 3$.

Figure 5.12: MVA_{TR} with commutative function as countermeasure.

The other parameter is the number of possible commutative permutations. Indeed if this number is too low an attacker can test all the permutations and build attacks such as in (174). For example using the possible power of S in AES we reach 277182 which is hard to exhaustively test but remains possible. For some specific cases, such as involutional block ciphers (e.g., ICEBERG (163)), the countermeasure cannot apply because the order of the substitution box S is equal to two.

Of course another parameter is the security of the permutation generation itself against possible Side Channel Analysis. If an attacker is able for example to recover: $p \pmod{l_1}$, $p \pmod{l_2}$, ..., $p \pmod{l_m}$, he will be able to recover the random permutation. This means that at least the exponentiation of S should be in constant time.

5.7 Conclusions and Perspectives

The table recomputation is a known weakness of masking schemes. We have recalled that practical countermeasures (e.g., shuffling with a high entropy) could be built to protect the table recomputation. In this chapter, we have presented a new multivariate attack exploiting the leakage of the protected table that outperformed classical HODPA even if a large amount of entropy is used to generate the countermeasure. This multivariate attack gives an example of a HODPA of non-minimal order which is more efficient than the corresponding minimal order HODPA. We have theoretically expressed the bound of noise in which this attack outperforms HOCPA using the SNR. Then we have empirically validated this bound. Interestingly, we show that if the leakage model consists in a linear combination of bits, then our attack becomes all the better as the model gets further away from uniform weights (so-called Hamming weight model). Moreover, we have shown that the relative gain to use the multivariate attack grows linearly with the order of the masking schemes. This result highlights the fact that the study of masking scheme should take into account as second parameter the number of variables exploitable by these attacks. Indeed we have shown in this chapter that when the number of variables used to perform the attacks increases, the *order* does not alone provide a criterion to evaluate the security of the countermeasure, and that the SNR is a better security metric to consider.

In future works we will investigate how to protect table recomputation against such attacks and investigate the cost of such countermeasures, evaluate the threat of such attacks on high-order masking schemes implemented on real components. We will also investigate how multivariate attacks could be applied on other masking schemes and protection techniques. And then, we will quantify the impact of these attacks.

CHAPTER 6

Truncation of Optimal Distinguisher against shuffled Masking Table

The results presented in this chapter have been published in collaboration with Sylvain Guilley, Annelie Heuser and Olivier Rioul in the international Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2016) (25).

Contents

The maximum likelihood side-channel distinguisher of a template attack scenario is expanded into lower degree attacks according to the increasing powers of the signal-to-noise ratio (SNR). By exploiting this decomposition we show that it is possible to build highly multivariate attacks which remain efficient when the likelihood cannot be computed in practice due to its computational complexity. The shuffled table recomputation is used as an illustration to derive a new attack which outperforms the ones presented by Bruneau et al. at CHES 2015, and so across the full range of SNRs. This attack combines two attack degrees and is able to exploit high dimensional leakage which explains its efficiency.

6.1 Introduction

In order to protect embedded systems against side-channel attacks, countermeasures need to be implemented. Masking and shuffling are the most investigated solutions for this purpose (99). Intuitively, masking aims at increasing the order of the statistical moments (in the leakage distributions) that reveal sensitive information (30, 78), while shuffling aims at increasing the noise in the adversary's measurements (73). As a result, an important challenge is to develop sound tools to understand the security of these countermeasures and their combination (142). For this purpose, the usual strategy is to consider template attacks for which one can split the evaluation goals into two parts: offline profiling (building an accurate leakage model) and online attack (recovering the key using the leakage model). As far as profiling is concerned, standard methods range from non-parametric ones (e.g., based on histograms or kernels) of which the cost quite highly suffers from the curse of dimensionality (see e.g., (4) for an application of these methods in the context of non-profiled attacks) to parametric methods, typically exploiting the mixture nature of shuffled and masked leakage distributions (91, 92, 121, 125, 164), which is significantly easier if the masks (and permutations) are known during the profiling phase. Our premise in this chapter is that an adversary is able to obtain such a mixture model via one of these means, and therefore we question its efficient exploitation during the online attack phase.

In this context, a starting observation is that the time complexity of template attacks exploiting mixture models increases exponentially with the number of masks (when masking) and permutation length (when shuffling (176)). So typically, the time complexity of an optimal template attack exploiting Q traces against an implementation where each n-bit sensitive value is split into Ω shares and shuffled over Π different positions is in $\mathcal{O}\left(Q \cdot (2^n)^{\Omega-1} \cdot \Pi!\right)$, which rapidly turns out to be intractable. In order to mitigate the impact of this high complexity, we propose a small, well-controlled and principled relaxation of the optimal distinguisher, based on its Taylor expansion (already mentioned in the field of side-channel analysis in (23, 46)) of degree L. Such a simplification leads to various concrete advantages. First, when applied to masked implementations, it allows us to perform the (mixture) computations corresponding to the $(2^n)^\Omega$ factor in the complexity formula only once (thanks to precomputation) rather than Q times. Second, when applied to shuffled implementations, it allows us to replace the $\Pi!$ factor in this formula by $\binom{\Pi}{\min\left(\left\lceil \frac{\Pi}{2} \right\rceil, L\right)} = \binom{\Pi}{L}$, thanks to the bounded degree L.

Additionally it can be noticed that an attacker will only build, during the offline profiling, the leakage models needed for the attack. By applying the Taylor expansion of the optimal

distinguisher the complexity of the offline profiling is significantly reduced. In general the complexity of the offline profiling becomes equivalent to the complexity of the online attack.

The resulting "rounded template attacks" additionally carry simple intuitions regarding the minimum degree of the Taylor expansion needed for the attacks to succeed. Namely, this degree L needs to be at least equal to the security order O of the target implementation, defined as the smallest statistical moment in the leakage distributions that are key-dependent.

We then show that these attacks only marginally increase the data complexity (for a given success rate) when applied against a masked (only) implementation. More importantly, we finally exhibit that rounded template attacks are especially interesting in the context of high-dimensional higher-order side-channel attacks, and put forward the significant improvement of the attacks against the masked implementations with shuffled table recomputations from CHES 2015 (27).

Contributions. We show that the expansion of the likelihood allows attacks with a very high computational *efficiency*, while remaining very *effective* from a key recovery standpoint. This means that the expanded distinguisher requires only little more traces to reach a given success rate, while being much faster to compute.

We also show how to grasp in a multivariate setting several leakages of different orders. In particular, we present an attack on shuffled table recomputation which succeeds with less traces than (27). Notice that the likelihood attack cannot be evaluated in this setting because it is computationally impossible to average over both the mask and the shuffle (the sole number of shuffles is $2^n! \approx 2^{1684}$ with $n = 8$).

Finally, we show that are our rounded version of the maximum likelihood allows better attacks than the state-of-the-art. Namely, our attack is better than the classical 2O-CPA and the recent attack of CHES'15 (27) in all noise variance settings.

Outline. The remainder of the chapter is organized as follows. Sec. 6.2 provides the necessary notations and mathematical definitions. The theoretical foundation of our method is presented in Sec. 6.3. The case study (shuffled table recomputation) is shown in Sec. 6.4. Sec. 6.5 evaluates the complexity of our method. The performance results are presented in Sec. 6.6. Conclusions and perspectives are presented in Sec. 6.7. Some technical results are deferred to the appendices.

6.2 Notations

Randomization countermeasures consist in *masking* and *shuffling* protections. When evaluating randomized implementations, there are a number of important parameters to consider. First, the number of shares and the shuffle length in the scheme, next denoted as Ω and Π, are algorithmic properties of the countermeasure. These numbers generally influence the tradeoff between the implementation overheads and the security of the countermeasures. Second, the order of the implementation protected by a randomization countermeasure, next denoted as O, which is a statistical property of the implementation. It corresponds to the smallest key-dependent statistical moment in the leakage distributions. When only masking is applied and the masked implementation is "perfect" (meaning that the leakage of each share is independent of each other), the order O equals to Ω at best. Finally, the number of dimensions (or dimensionality) used in the traces, next denoted as D, is a property of the adversary. In this respect, adversaries may sometimes be interested by using the lowest possible D (since it makes the detection of POIs in the traces easier). But from the measurement complexity point of view, they have a natural incentive to use D as large as possible. A larger dimension D allows to increase the signal to noise ratio (21).

In summary, our notations are:

- Ω: number of shares in the masking countermeasure,
- Π: length of the shuffling countermeasure,
- O: order of the implementation,
- D: dimensionality of the leakages.

Examples. Existing masking schemes combine these four values in a variety of manners. For example, in a perfect hardware masked implementation case with three shares, we may have $\Omega = 3$, $O = 3$ and $D = 1$ (since the three shares are manipulated in parallel). If this implementation is not perfect, we may observe lower order leakages (e.g. $\Omega = 3$, $O = 1$ and $D = 1$, that is a first-order leakage). And in order to prevent such imperfections, one may use a Threshold Implementation (117), in which case one share will be used to prevent glitches (so $\Omega = 3$, $O = 2$ and $D = 1$). If we move to the software case, we may then have more informative dimensions, e.g. $\Omega = 3$, $O = 3$, $D = 3$ if the adversary looks for a single triple of informative POIs. But we can also have a number of dimensions significantly higher than the order (which

usually corresponds to stronger attacks). Let us also give an example of S-boxes masking with one mask, where the masking process of the S-box (often called recomputation) is shuffled. A permutation Φ of $\Pi = 2^n$ values is applied while computing the masked table. If the attacker ignores the recomputation step, he can carry out an attack on the already computed table. Hence parameters $\Omega = 2$, $O = 2$, $D = 2$ (also known as "second-order bivariate CPA"). But the attacker can also exploit the shuffled recomputation of the S-box in addition to a table look-up, as presented in (27); the setting is thus highly multivariate: $\Omega = 2$, $\Pi = 2^n$, $O = 2$, $D = 2 \cdot 2^n + 1$. Interestingly, the Chapt. 5 shows an attack at degree $L = 3$ which succeeds in less traces than attacks at minimal degree $L = O = 2$.

In general, a template attack based on mixture distributions (often used in parametric estimation) would require a summation over all random values of the countermeasure, that is $\mathcal{R}$, which consists in the set of masks and permutations. One can represent $\mathcal{R}$ as the Cartesian product of the set of mask and the set of permutations. Let us denote by $\mathcal{M}$ the set of mask and $\mathcal{S}$ the set of permutations. Then $\mathcal{R} = \mathcal{M} \times \mathcal{S}$. Therefore, the cardinality of $\mathcal{R}$ is $2^{n(\Omega-1)}\Pi!$.

Eventually, the security of a masked implementation depends on its order and noise level. More precisely, the security increases exponentially with the order (with the noise as basis) (47). So for the designer, there is always an incentive to increase the noise and order. And for adversary, there is generally an incentive to use the largest possible D (given the time constraints of his attack), so that he decreases the noise.

6.2.1 Model

We characterize the protection level in terms of the most powerful attacker, namely an attacker who knows everything about the design, except the masks and the noise. This means that we consider the case where the templates are known. How the attacker got the templates is related with *security by obscurity*, somehow he will know the model. Of course depending on the learning phase these estimations can be more or less accurate. For the sake of simplicity we assume in this chapter the better scenario where all the estimations are exact[1].

Besides, we assume that the noise is independently distributed over each dimension. This is the least favorable situation for the attacker (as there is in this case the most noise entropy). For the sake of simplicity, we assume that the noise variance is equal to σ^2 at each point

[1] We recall that, even if the templates are perfectly known, the online attack phase still requires $\mathcal{O}(Q \cdot 2^{n(\Omega-1)} \cdot \Pi!)$ computations.

$d = 1, 2, \ldots, D$. This allows for a simple theoretical analysis. Let us give an index $q = 1, 2, \ldots, Q$ to each trace. For one trace q, the model is written as:

$$X = y(t, k^\star, R) + N, \tag{6.1}$$

where for notational convenience the dependency in q and d has been dropped. Here X is a leakage measurement; $y = y(t, k^\star, R)$ is the deterministic part of the model that depends on the correct key $k^\star$, some known text (plaintext or ciphertext) t, and the unknown random values (masks and permutations) R. Each sample (of index d) of N is a random noise, which follows a Gaussian distribution $p_N(z) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{z^2}{2\sigma^2}\right)$.

6.3 A Generic Log-Likelihood for Masked Implementations

In this section we derive a rounded version of Template Attack. Namely we expand a particular instantiation of the template attack the so-called optimal distinguisher using its Taylor Expansion. By rounding this expansion at the Lth degree we are able to build a rounded version of the optimal distinguisher (later defined as ROPT_L). This attack features two advantages: it allows to combine different statistical moments and its complexity becomes manageable.

6.3.1 Maximum Likelihood (ML) Attack

The most powerful adversary knows exactly the leakage model (but the actual key, the masks, and the noise are unknown during the online step) and computes a likelihood. In the case of masking the optimal distinguisher which maximize the success rate is given by (23):

Theorem 6.3.1 (Maximum Likelihood)**.** *When the $y(t, k, R)$ are known and the Gaussian noise N is i.i.d. across the queries (measurements) and independent across the dimension, then the optimal distinguisher is:*

$$\begin{aligned} \mathrm{OPT}\colon \quad \mathbb{R}^{DQ} \times \mathbb{R}^{DQ} &\longrightarrow \mathbb{F}_2^n \\ (\mathbf{x}, y(\mathbf{t}, k, R)) &\longmapsto \operatorname*{argmax}_{k \in \mathbb{F}_2^n} \sum_{q=1}^{Q} \log \mathbb{E} \exp \frac{-\|x_q - y(t_q, k, R)\|^2}{2\sigma^2} \end{aligned} \tag{6.2}$$

where the expectation operator $\mathbb{E}$ is applied with respect to the random variable $R \in \mathcal{R}$, and the norm is the Euclidean norm $\|x_q - y(t_q, k, R)\|^2 = \sum_{d=1}^{D} (x_q^{(d)} - y^{(d)}(t_q, k, R))^2$.

Proof. It is proven in (23) that the Maximum Likelihood distinguisher is:

$$\operatorname*{argmax}_{k \in \mathbb{F}_2^n} \prod_{q=1}^{Q} \sum_{r \in \mathcal{R}} \mathbb{P}(r)\, p\left(x_q | y(t_q, k, r)\right).$$

Applying (6.1) for Gaussian noise and taking the logarithm yields (6.2). □

Remark 26. *Notice that for each trace q, the Maximum Likelihood distinguisher involves a summation over $\#\mathcal{R}$ values, which correspond to $\#\mathcal{R}$ accesses to precharacterized templates.*

If $D = 1$, then the signal-to-noise ratio (SNR) is defined in a natural way as the ratio between the variance of the model Y and the variance of the noise N. But when the setup is multivariate, it is more difficult to quantify a notion of SNR. For this reason, we use the following quantity

$$\gamma = \frac{1}{2\sigma^2}, \tag{6.3}$$

which is actually proportional to an SNR, in lieu of SNR. In practice, we assume that γ is small. It is indeed a condition for masking schemes to be efficient (see for instance (47)).

Proposition 29 (Taylor Expansion of Optimal Attacks in Gaussian Noise)**.** *The attack consists in maximizing the sum over all traces $q = 1, \ldots, Q$ of*

$$\sum_{\ell=1}^{+\infty} \frac{\kappa_\ell}{\ell!}(-\gamma)^\ell, \tag{6.4}$$

where κ_ℓ is the ℓth-order cumulant of the random variable $\|x - y(t,k,R)\|^2$, which can be found inductively from ℓth-order moments:

$$\mu_\ell = \mathbb{E}_R\big(\|x - y(t,k,R)\|^{2\ell}\big), \tag{6.5}$$

using the relation:

$$\kappa_\ell = \mu_\ell - \sum_{\ell'=1}^{\ell-1} \binom{\ell-1}{\ell'-1} \kappa_{\ell'} \mu_{\ell-\ell'} \qquad (\ell \geq 1). \tag{6.6}$$

Proof. The log-likelihood can be expanded according to the increasing powers of the SNR as:

$$\log \mathbb{E} \exp\big(-\gamma \|x - y(t,k,R)\|^2\big) = \sum_{\ell=1}^{+\infty} \frac{\kappa_\ell}{\ell!}(-\gamma)^\ell, \tag{6.7}$$

where we have recognized the cumulant generating function (166). The above relation (6.6) between cumulants and moments is well known (180). □

Definition 53. *The Taylor expansion of the log-likelihood truncated to the Lth degree LL_L in SNR is*

$$\mathrm{LL}_L = \sum_{\ell=1}^{L} (-1)^\ell \kappa_\ell \frac{\gamma^\ell}{\ell!}. \tag{6.8}$$

Put differently, we have $\mathrm{LL} = \mathrm{LL}_L + o(\gamma^L)$ (using the Landau notation). The optimal attack can now be "rounded" in the following way:

Definition 54 (Rounded OPTimal Attack of Degree L in γ)**.** *The rounded optimal* Lth-degree attack *consists in maximizing over the key hypothesis the sum over all traces of the* L*th order Taylor expansion* LL_L *in the SNR of the log-likelihood:*

$$\begin{array}{rccc} \mathrm{ROPT}_L\colon & \mathbb{R}^{DQ}\times\mathbb{R}^{DQ} & \longrightarrow & \mathbb{F}_2^n \\ & (\mathbf{x}, y(\mathbf{t}, k, R)) & \longmapsto & \underset{k\in\mathbb{F}_2^n}{\operatorname{argmax}}\,\mathrm{LL}_L. \end{array} \tag{6.9}$$

Proposition 30. *If the degree* L *is smaller than the order* O *of the countermeasure then the attack fails to distinguish the correct key.*

Proof. One can notice that μ_ℓ combines (by a product) a most ℓ terms following the formula:

$$\mu_\ell = \sum_{k_1+\ldots+k_D=\ell} \binom{\ell}{k_1,\ldots,k_D} \mathbb{E} \prod_{0<i<D+1} (x^{(i)} - y^{(i)})^{2\cdot k_i},$$

with $k_1 + \ldots + k_d = \ell$. It implies that it exits at most ℓ different $k_i > 0$ and as a consequence there are at most ℓ different variables in the expectation. Therefore by definition of a perfect masking scheme μ_L does not depend on the key. As a consequence LL_L with $L < O$ neither depends on the key.

□

Theorem 6.3.2. *Let an implementation be secure at order* $O-1$*. The lowest-degree successful attack is the one at degree* $L = O$ *which maximizes* LL_L*. This is equivalent to summing*

$$\mu_L = \mathbb{E}_R\big(\|x - y(t,k,R)\|^{2L}\big),$$

over all traces and

- *maximize the result over the key hypotheses, if* L *is even;*
- *minimize the result over the key hypotheses, if* L *is odd.*

Proof. Since κ_ℓ is independent of k for all $\ell \leq L$, the first sensitive contribution to the log-likelihood is

$$(-1)^L \kappa_L \frac{\gamma^L}{L!}.$$

Now, $\kappa_L = \mu_L +$ lower order terms (which do not depend on the key as the implementation is secure at order $O-1$), and removing constants independent of k the contribution to the log-likelihood reduces to $(-1)^L \mu_L$. □

Theorem 6.3.3 (Mixed Degree Attack)**.** *Assuming an implementation secure at order* O*, the next degree successful attack is the one at degree* $L+1 = O+1$ *which maximizes* LL_{L+1}*. This is equivalent to summing*

$$\mu_L(1+\gamma\mu_1) - \gamma\frac{\mu_{L+1}}{L+1},$$

over all traces and

- *maximize the result over the key hypotheses, if L is even;*
- *minimize the result over the key hypotheses, if L is odd.*

Proof. The $(L+1)$th-order term in the log-likelihood becomes

$$(-1)^L \kappa_L \frac{\gamma^L}{L!} + (-1)^{L+1} \frac{\kappa_{L+1}}{(L+1)!} \gamma^{L+1}.$$

Now from (6.6) we have, for $L > 0$

$$\kappa_{L+1} = \mu_{L+1} - (L+1)\mu_L \mu_1 + \text{ lower-order terms.}$$

Removing terms that do not depend on k, we obtain:

$$(-1)^L \gamma^L \Big(\mu_L - \gamma (\frac{\mu_{L+1}}{L+1} - \mu_L \mu_1) \Big).$$

Compared to a Lth-degree attack, we see that μ_L is replaced by a corrected version:

$$\mu_L (1 + \gamma \mu_1) - \gamma \frac{\mu_{L+1}}{L+1},$$

where μ_1 is independent of k. However, μ_1 cannot be removed as it scales the relative contribution of μ_L and μ_{L+1} in the distinguisher. □

Remark 27. *In contrast to* LL_L*, implementing* LL_{L+1} *requires knowledge of the SNR parameter* $\gamma = 1/2\sigma^2$*.*

Remark 28. *In general, when $L \geq O$ the rounded optimal attack* ROPT_L *exploits all key dependent terms of degree ℓ, where $O \leq \ell \leq L$, whereas an LO-CPA (30) or MCP-DPA (113) only exploits the term of degree L.*

6.4 Case Study: Shuffled Table Recomputation

In this section we apply the ROPT_L formula of Eq. (6.9) in Def. 54 to the particular case of a block cipher with a shuffled table recomputation stage. We show that in this scenario our new method allows to build a better attack than that from the state-of-the-art. By combining the second and the third cumulants we construct an attack which is better than:

- any second-order attack;
- the attack presented at CHES 2015. Following the notations of (27) we denote this attack by MVA_{TR} (which stands for Multi-Variate Attack on Table Recomputation) in the rest of this chapter. This is a third-order attack that achieves better results than 2O-CPA when the noise level σ is below a given threshold (namely $\sigma^2 \leq 2^{n-2} - n/2$).

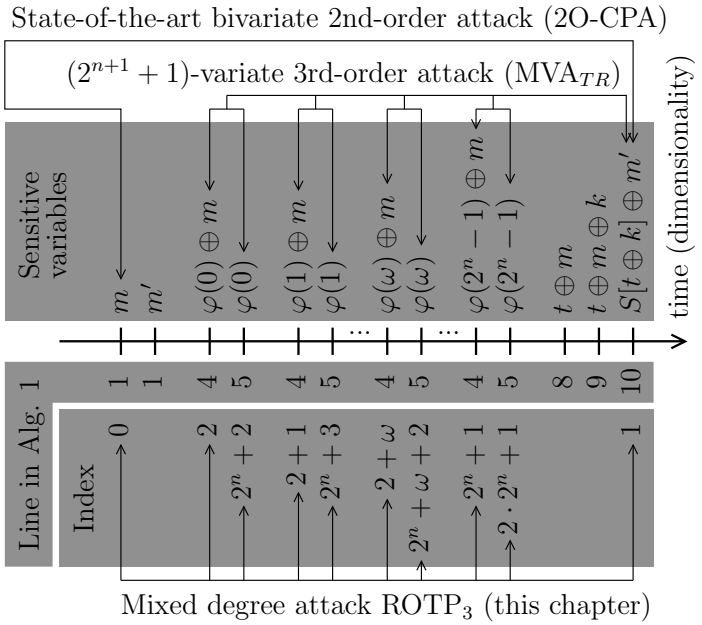


Figure 6.1: Leakages of the shuffled table recomputation scheme

6.4.1 Parameters of the Randomization Countermeasure

In order to validate our results we take as example a first order ($O = 2$), masking scheme where the sensitive variables are split into two shares ($\Omega = 2$). The nonlinear part of this scheme is computed using a table recomputation stage. This step is shuffled ($\Pi = 2^n$) for protection against some known attacks (124, 174). Then an attacker builds highly multivariate attacks with a dimensionality of $D = 2 \cdot 2^n + 1$. The beginning of this combined countermeasure is given in Algorithm 2 of Chapt. 4.

We used lower case letter (e.g., m, φ) for the realizations of random variables, written upper-case (e.g., M, Φ). For the sake of simplicity in the rest of this case study, we assume that $m = m'$.

An overview of the leakages over time is given in Fig. 6.1.

We detail below the mathematical expression of these leakages. The randomization consists in one mask M chosen randomly in $\{0,1\}^n$, and one shuffle (random permutation of $\{0,1\}^n$) denoted by Φ. Thus, we denote $R = (M, \Phi)$, which is uniformly distributed over the Cartesian product $\{0,1\}^n \times S_{2^n}$ (i.e. $\mathcal{M} = \{0,1\}^n$ and $\mathcal{S} = S_{2^n}$), where S_m is the symmetric group of m elements. We have $D = 2^{n+1} + 2$ leakage models, namely:

- $X^{(0)} = y^{(0)}(t,k,R) + N^{(0)}$ with $y_0(t,k,R) = \mathrm{HW}[M]$,
- $X^{(1)} = y^{(1)}(t,k,R) + N^{(1)}$ with $y_1(t,k,R) = \mathrm{HW}[S[T \oplus k] \oplus M]$,
- $X^{(i)} = y^{(i)}(t,k,R) + N^{(i)}$, for $i = 2, \ldots, 2^n + 1$ with $y^{(i)}(t,k,R) = \mathrm{HW}[\Phi(i-2) \oplus M]$,
- $X^{(j)} = y^{(j)}(t,k,R) + N^{(j)}$, for $j = 2^n + 2, \ldots, 2^{n+1} + 1$ with $y^{(j)}(t,k,R) = \mathrm{HW}[\Phi(j - 2^n - 2)]$.

We recall that we assume the noises N are i.i.d. Clearly, there is a second-order leakage, as the pair $(X^{(0)}, X^{(1)})$ does depend on the key. But there is also a large multiplicity of third-order leakages, such that $(X^{(1)}, X^{(i)}, X^{(j=i+2^n)})$, as will be analyzed in this case study.

The following side-channel attacks are applied on a set of Q realizations. Let us define I and J as $I = [\![2, 2^n + 1]\!]$ and $J = [\![2^n + 2, 2 \times 2^n + 1]\!]$. Then the maximal dimensionality is $D = 2 + 2 \times 2^n$, and we denote a sample d as $d \in \{0,1\} \cup I \cup J$. The Q leaks (resp. models) at sample d are denoted as $\mathbf{x}^{(d)}$ and $\mathbf{y}^{(d)} = y^{(d)}(\mathbf{t}, k, R)$. For any measurement indexed by $q \in Q$:

- We denote by $x_q^{(0)}$ the leakage of the mask (M).
- We denote by $x_q^{(1)}$ the leakage of the masked SubBytes ($S[t_q \oplus k] \oplus M$).
- The 2^n leakages of the random masked index ($\Phi(\omega) \oplus M$ where ω denotes the index of the loop of the recomputation stage) are denoted by $x_q^{(i)}$ with $i \in [\![2, 2^n + 1]\!] = I$. Then $\omega = i - 2$.
- We denote by $x_q^{(j)}$ the 2^n leakages of the random index ($\Phi(\omega)$) with $j \in [\![2^n+2, 2\times 2^n+1]\!] = J$. Then $\omega = j - 2^n + 2$.

Let us denote in bold letter the set of leakages of all the queries i.e. $\mathbf{x}^{(d)} = \left(x_0^{(d)}, ..., x_{Q-1}^{(d)}\right)$, $d \in \{0,1\} \cup I \cup J$.

In this section $y^{(d)}(t_q, k, R)$, $d \in \{0,1\} \cup I \cup J$ denotes the expected leakage model of $x_q^{(d)}$. The expected leakage model for all queries is given by $y^{(d)}(\mathbf{t}, k, R) = \left(y^{(d)}(t_0, k, R), ..., y^{(d)}(t_{Q-1}, k, R)\right)$, $d \in \{0,1\} \cup I \cup J$.

In order to simplify the notations we introduce

$$f_q^{(d)} = \left(x_q^{(d)} - y^{(d)}(t_q, k, R)\right)^2, \tag{6.10}$$

with $d \in \{0,1\} \cup I \cup J$. The $_q$ can be omitted where there is no ambiguity.

6.4.2 Second-Order Attacks

As any other high order masking scheme, our example can be defeated by High Order Attacks (30, 106, 133, 177). As our scheme is a first order masking scheme with two shares it can be defeated using a second order attack (30, 106) which combines the leakages of the two shares using a *combination function* (30, 106, 121) such as the second order CPA (2O-CPA) with the centered product as combination function.

Using our notation it implies $D = 2$.

Definition 55 (2O-CPA (133))**.** *We denote by* 2O-CPA *the* CPA *using the centered product as combination function. Namely:*

$$\begin{array}{rccl} \text{2O-CPA}: & \mathbb{R}^Q \times \mathbb{R}^Q \times \mathbb{R}^Q & \longrightarrow & \mathbb{F}_2^n \\ & \left(\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \mathbf{y}\right) & \longmapsto & \underset{k \in \mathbb{F}_2^n}{\operatorname{argmax}} \widehat{\rho}\left[\mathbf{x}^{(0)} \odot \mathbf{x}^{(1)}, \mathbf{y}\right], \end{array} \tag{6.11}$$

where $\mathbf{y} = \mathbb{E}_M\left(y_0\left(\mathbf{t}, k, R\right) \odot y_1\left(\mathbf{t}, k, R\right)\right)$*,* $\odot$ *is the element wise product and* $\widehat{\rho}$ *is an estimator of the Pearson correlation coefficient. It can be noticed that as the terms* $y^{(0)}\left(\mathbf{t}, k, R\right)$ *and* $y^{(1)}\left(\mathbf{t}, k, R\right)$ *only depend on* M *the expectation is only computed over* $\mathcal{M}$*.*

Remark 29. *Here we have assumed without loss of generality that the leakages and the model are centered.*

An attacker can restrict himself in order to ignore the recomputation stage. Since such attacker ignores the table recomputation no random shuffle is involved. As a consequence the optimal distinguisher restricted to these leakages becomes computable. Nevertheless as we will see in Sec. 6.6 this approach is not the best. Indeed a lot of exploitable information is lost by not taking into account the table recomputation.

Definition 56 ($\mathrm{OPT}_{2\mathrm{O}}$ Distinguisher — Eq. (6.2) for $D = 2$)**.** *We define by* $\mathrm{OPT}_{2\mathrm{O}}$ *the optimal attack which targets the mask and the masked sensitive value.*

$$\begin{array}{rccl} \mathrm{OPT}_{2\mathrm{O}}: & \mathbb{R}^{2Q} \times \mathbb{R}^{2Q} & \to & \mathbb{F}_2^n \\ & \left(\mathbf{x}^{(d)}, y^{(d)}\left(\mathbf{t}, k, R\right)\right)_{d \in \{0,1\}} & \mapsto & \underset{k \in \mathbb{F}_2^n}{\operatorname{argmax}} \sum_{q=1}^{Q} \log \mathbb{E} \exp\left(-\gamma \sum_{d \in \{0,1\}} f_q^{(d)}\right), \end{array} \tag{6.12}$$

with $f_q^{(d)}$ *as defined in Eq.* (6.10)*.*

6.4.3 Exploiting the Shuffled Table Recomputation Stage

It is known that the table recomputation step can be exploited to build better attacks than second order attacks (23, 174). Recently a new attack has been presented which remains better

than the 2O-CPA even when the recomputation step is protected (27). Let us recall the definition of this attack:

Definition 57 (MVA_{TR} (27))**.** *The MultiVariate Attack (MVA) exploiting the leakage of the table recomputation (TR) is given by the function:*

$$\begin{array}{rccl} \mathrm{MVA}_{TR}\colon & \mathbb{R}^{Q\left(2^{n+1}+1\right)} \times \mathbb{R}^{Q} & \longrightarrow & \mathbb{F}_2^n \\ & \left(\mathbf{x}^{(d)}, \mathbf{y}\right)_{d\in\{1\}\cup I\cup J} & \longmapsto & \underset{k\in\mathbb{F}_2^n}{\operatorname{argmax}}\, \widehat{\rho}\left[\left(-\frac{1}{2}\sum_{\substack{i\in I,\\ j=i+2^n}} \mathbf{x}^{(i)} \odot \mathbf{x}^{(j)}\right) \odot \mathbf{x}^{(1)}, \mathbf{y}\right], \end{array} \tag{6.13}$$

where, like for Def. 55, $\mathbf{y} = \mathbb{E}_M\left(y^{(0)}\left(\mathbf{t}, k, R\right) \odot y^{(1)}\left(\mathbf{t}, k, R\right)\right)$*,* $\odot$ *is the element wise product and* $\widehat{\rho}$ *is an estimator of the Pearson coefficient.*

Let us now apply our new ROPT_L on a block cipher protected with a shuffled table recomputation. In this case the lower moments are given by:

$$\mu_\ell = \mathbb{E}\left[\left(\sum_d f^{(d)}\right)^\ell\right] = \mathbb{E}\left[\Big(\underbrace{f^{(0)}}_{S[t\oplus k]\oplus M} + \underbrace{f^{(1)}}_{M} + \sum_{i\in I}\underbrace{f^{(i)}}_{\Phi(\omega)\oplus M} + \sum_{j\in J}\underbrace{f^{(j)}}_{\Phi(\omega)}\Big)^\ell\right].$$

Proposition 31. *The second degree rounded optimal attack (Def. 54 for* $L = 2$*) on the table recomputation is:*

$$\begin{array}{rccl} \mathrm{ROPT}_2\colon & \mathbb{R}^{2Q} \times \mathbb{R}^{2Q} & \longrightarrow & \mathbb{F}_2^n \\ & \left(\mathbf{x}^{(d)}, y^{(d)}\left(\mathbf{t}, k, R\right)\right)_{d\in\{0,1\}} & \longmapsto & \underset{k\in\mathbb{F}_2^n}{\operatorname{argmax}} \sum_{q=1}^{Q} \mathbb{E}(f_q^{(0)} \times f_q^{(1)}). \end{array} \tag{6.14}$$

Proof. Combine Theorem 6.3.2 and Eq. (C.9) of Appendix C.1.2. □

Remark 30. *The* ROPT_2 *which targets the second order moment happens not to take into account the terms of the recomputation stage. Naturally the only second order leakages are also the ones used by* 2O-CPA *and* $\mathrm{OPT}_{2\mathrm{O}}$ *distinguishers.*

Proposition 32. *The third degree rounded optimal attack (Def. 54 for* $L = 3$*) on the table recomputation is:*

$$\begin{array}{rccl} \mathrm{ROPT}_3\colon & \mathbb{R}^{(2^{n+1}+2)Q} \times \mathbb{R}^{(2^{n+1}+2)Q} & \longrightarrow & \mathbb{F}_2^n \\ & \left(\mathbf{x}^{(d)}, y^{(d)}\left(\mathbf{t}, k, R\right)\right)_{d\in\{0,1\}\cup I\cup J} & \longmapsto & \underset{k\in\mathbb{F}_2^n}{\operatorname{argmax}} \sum_{q=1}^{Q} \mu_{2_q}(1+\gamma\mu_{1_q}) - \gamma\frac{\mu_{3_q}}{3}, \end{array} \tag{6.15}$$

where the values of μ_{1_q}*,* μ_{2_q} *and,* μ_{3_q} *are respectively provided in Eq.* (C.1) *of Appendix C.1.1, Eq.* (C.9) *of Appendix C.1.2 and Eq.* (C.12) *of Appendix C.1.3.*

Proof. Combining Theorem 6.3.2 and Appendix C.1. □

Proposition 33. *To compute* μ_1*,* μ_2 *and* μ_3 *an attacker does not need to compute the prohibitive expectation over* S_{2^n}*.*

Proof. Proof given in Appendix C.1. □

6.5 Complexity

In this section we give the *time* complexity needed to *compute* OPT and ROPT_L. We also show that when $L \ll D$ the complexity of ROPT_L remains manageable whereas the complexity of OPT is prohibitive. In this section all the complexities are computed for one key guess.

6.5.1 Complexity in the General Case

Let us first introduce an intermediate lemma.

Lemma 7. *The complexity of computing μ_ℓ (for one trace) is lower than:*

$$\mathcal{O}\left(\binom{D+\ell-1}{\ell} \cdot 2^{(\Omega-1)n} \cdot \binom{\Pi}{\min\left(\left\lceil \frac{\Pi}{2} \right\rceil, \ell\right)}\right). \tag{6.16}$$

Proof. See Appendix C.2.1. □

Proposition 34. *The complexity of* OPT *is:*

$$\mathcal{O}\left(Q \cdot (2^n)^{\Omega-1} \cdot \Pi! \cdot D\right). \tag{6.17}$$

The complexity of ROPT_L *is lower than:*

$$\mathcal{O}\left(Q \cdot L \cdot \binom{D+L-1}{L} \cdot 2^{(\Omega-1)n} \cdot \binom{\Pi}{\min\left(\left\lceil \frac{\Pi}{2} \right\rceil, L\right)}\right). \tag{6.18}$$

Proof. The proof is given in Appendix C.2.2. □

Prop. 34 allows to compare the complexity of the two attacks. One can notice that there are still terms with $\Pi!$ or $D!$ in ROPT_L such as $\binom{D+L-1}{L}$ or $\binom{\Pi}{\min\left(\left\lceil \frac{\Pi}{2} \right\rceil, L\right)}$. Nevertheless these two terms can be seen as constants when $L \ll D$. As a consequence we have the following remark.

Important Remark. When the degree L of the attack ROPT_L is such that $L \ll D$ the complexity of OPT is much higher than the complexity of ROPT_L. Indeed the main term for OPT is $\Pi!$ whereas the one for ROPT_L is $2^{(\Omega-1)n}$.

Proposition 35. *The complexity of* ROPT_L *can be reduced to* $\mathcal{O}\left(Q \cdot L \cdot \binom{D+L-1}{L}\right)$ *with a pre-computation in* $\mathcal{O}\left(L \cdot \binom{D+L-1}{L} \cdot 2^{(\Omega-1)n} \cdot \binom{\Pi}{\min\left(\left\lceil \frac{\Pi}{2} \right\rceil, L\right)}\right)$.

Proof. See Appendix C.2.3. □

This means that for Q large enough i.e. when γ is low enough this computational "trick" allows a speed-up factor of $2^{(\Omega-1)n}\binom{\Pi}{\min\left(\left\lceil \frac{\Pi}{2} \right\rceil, L\right)}$. The idea is to output the values depending on the queries from the computation of the expectations. These expectations only depend on the model which can be computed only once.

6.5.2 Complexity of our Case Study

Let us now compute the complexity of these two distinguishers applied to our case study. Of course an approach could be to use the formula of the previous section 6.5.1. But one can notice that a lot of terms could be independent of the key and as consequence not needed in an attack. Another approach is to use the formula of the distinguisher.

Proposition 36. *The complexity of* OPT *is:*

$$\mathcal{O}\left(Q \cdot (2^n) \cdot 2^n! \cdot \left(2^{n+1} + 2\right)\right). \tag{6.19}$$

The complexity of ROPT_2 *is:*

$$\mathcal{O}\left(Q \cdot 2^n\right). \tag{6.20}$$

The complexity of ROPT_3 *is lower than:*

$$\mathcal{O}\left(Q \cdot 2^{4n}\right). \tag{6.21}$$

Proof. See Appendix C.2.4. □

Remark 31. *As already mentioned an attacker can ignore the leakages of the table recomputation and only target the two shares. In such case the complexity of* $\mathrm{OPT}_{2\mathrm{O}}$ *(Def. 56) is* $\mathcal{O}\left(Q \cdot (2^n)\right)$*. With the result of Prop. 35 the complexity of* ROPT_2 *reduces to* $\mathcal{O}\left(Q\right)$*.*

Remark 32. *Using the result of Prop. 35 the complexity of* ROPT_3 *can be reduced to* $\mathcal{O}\left(Q \cdot 2^{2n}\right)$ *with a precomputation step of* $\mathcal{O}\left(2^{2n}\right)$*.*

Remark 33. *A summary of the complexity, and the computation time of the distinguishers are provided in Appendix C.2.5 in Table C.1.*

6.6 Simulation Results

In this section we validate in simulation the soundness of our approach for the case study described in Sec. 6.4.1. The results of these simulations are expressed in success rate (defined in (160) and denoted by SR). All simulations are computed using the Hamming weight model as a leakage model. As we assume an attacker with a perfect knowledge, the leakages are the model (denoted by y) plus some noise. The noise is Gaussian with a standard deviation of σ.

In Subsec. 6.6.1 we assume that the attacker does not take into account the table recomputation stage. He only targets the leakages of the mask and the masked share (the leakage of masked S-Box). Namely the leakages which occurs in lines 1 and 10 of Algorithm 2. This approach allows to compute the restricted version of the maximum likelihood. We compare the results of the maximum likelihood, our rounded version and the high order attacks.

In Subsec. 6.6.2 we present our main results. In this subsection the attacker can exploit the leakage of the mask, the masked share and all the leakages of the table recomputation. In this scenario we show that our rounded version of the optimal distinguisher outperforms all the attacks of the state-of-the-art.

6.6.1 Exploiting only Leakage of the Mask and the Masked Share

In this subsection all the attacks are computed using only the leakages of the line 1 and the line 10 of Algorithm 2 of Chapt. 4.

In this case study we assume a perfect masking scheme with: $Y^{(0)} = \mathrm{HW}[M]$ and $Y^{(1)} = \mathrm{HW}[S[T \oplus k] \oplus M]$.

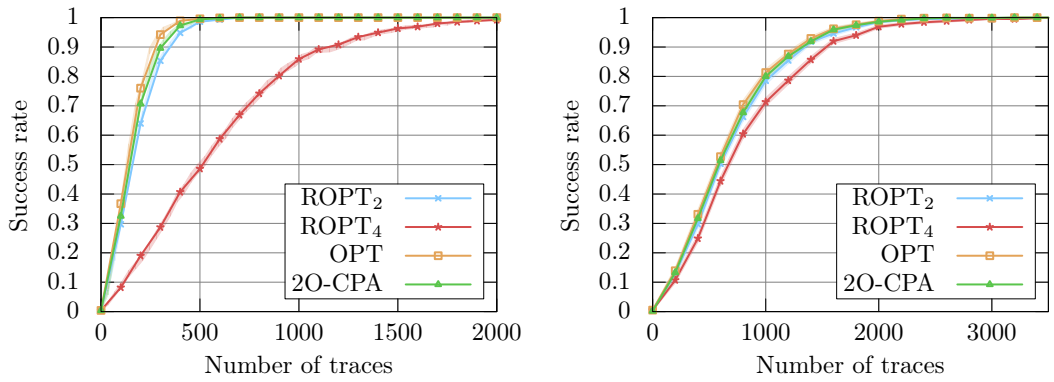
It can be seen in Fig. 6.2 that even for small noise ($\sigma = 1$, Fig. 6.2a) the 2O-CPA and ROPT_2 are equivalent. Indeed the two curves superimpose almost perfectly (in order to better highlight a difference, as many as 1000 attacks have been carried out for the estimation of the success rate). Moreover these two attacks are nearly equivalent to the optimal distinguisher (we recover here the results of (23)). We can notice that for both $\sigma = 1$ and $\sigma = 2$, ROPT_4 is not as good as ROPT_2. This means that the noise standard deviation is not large enough for approximations of higher degrees to be accurate. Indeed when the noise is not low enough the weight of each term of the decomposition can be such that some useful terms vanish due to the alternation of positive and negative terms in the Taylor expansion.

Let us recall that the decomposition of Eq. (6.8) is valid only for low $\gamma = 1/(2\sigma^2)$ i.e. high noise. The error term ($o(\gamma^L)$) in by the Taylor expansion gives the asymptotic evolution of this error when the noise increases but does not provide information about the error for a fixed value of noise variance. This means that the noise is too small for ROPT_4 to be a good approximation of OPT although ROPT_2 is nearly equivalent to OPT.

For $\sigma = 2$ the noise is high enough to have a good approximation of OPT by ROPT_4. For this noise all the attacks are close to OPT (Fig. 6.2b).

In the context where only the mask and the masked share are used it is equivalent to compute the 2O-CPA, ROPT_2 and OPT. As a consequence in the rest of this chapter only the 2O-CPA will be displayed.

To conclude our ROPT_L is in this scenario at least as good as the HO-CPA of order L, which validates the optimality of state-of-the-art attacks against perfect masking schemes of order $O = L$.



(a) $\sigma = 1$ **(b)** $\sigma = 2$

Figure 6.2: Bivariate attacks

6.6.2 Exploiting the Shuffled Table Recomputation

In this subsection the attacker can target the leakage of the mask, the masked share and all the leakages occurring during the table recomputation. As a consequence the attacks of Subsec. 6.6.1 remain possible. It has been shown in (23, 164) that the 2O-CPA with the centered product becomes close to the $\text{OPT}_{2\text{O}}$ (the Maximum Likelihood) when the noise becomes high. It is moreover confirmed by our simulation results as it can be seen in Fig. 6.2. We choose as attack reference for the Fig. 6.3 and Fig. 6.4 the 2O-CPA and not the $\text{OPT}_{2\text{O}}$ because it performs similarly 6.2 and it is much faster to compute (see Table C.1) which is mandatory for attacks with high noise (e.g. for $\sigma = 12$) which involve many traces.

Following the formulas provided previously empirical validations have been done. For $\sigma \leq 8$ the attacks have been redone 1000 times to compute the SR. For $\sigma > 8$ the attacks have been done 250 times. Results are plotted in Fig. 6.3 and Fig. 6.4. In these figures the results of the 2O-CPA, the MVA_{TR} and ROPT_3 are plotted. Noticed that the likelihood is not represented because we cannot average over R.

Recall that the cardinality of the support of R is $2^n \times 2^n!$. It can be first noticed that for all the noises ROPT_3 is the best attack.

Let us analyze how much better ROPT_3 is than 2O-CPA and MVA_{TR}. The comparison with our new attack can be divided in three different categories. For low noise $\sigma = 3$ (see Fig. 6.3b) the results of ROPT_3 are similar to the results of MVA_{TR}. This means that the leakage of the shuffled table recomputation is the most leaking term in this case. At the opposite when the noise is high (for $\sigma = 12$ see Fig. 6.4c) ROPT_3 becomes close to 2O-CPA which means that

as expected the most informative part is the second order term. For medium noise $7 \leq \sigma \leq 9$ (see Fig. 6.3d, Fig. 6.4a and Fig. 6.4b) the results of ROPT_3 are much better than the result of 2O-CPA and MVA_{TR}. Moreover, the gain compared to the second best attack is maximum when the results of 2O-CPA and MVA_{TR} are the same. Indeed for $\sigma = 7$ (see Fig. 6.3d), ROPT_3 needs 35000 traces to reach 80% of success whereas MVA_{TR} (the second best attack) needs 60000 traces. This represents a gain of 71%. For $\sigma = 8$ (see Fig. 6.4a), ROPT_3 needs 65000 traces to reach 80% of success whereas the MVA_{TR} and the 2O-CPA needs 120000 traces. This represents a gain of 85%. And when the noise increases to $\sigma = 9$ (see Fig. 6.4b), ROPT_3 needs 120000 traces to reach 80% of success whereas 2O-CPA (the second best attack) needs 200000 traces, which is a gain of 66%.

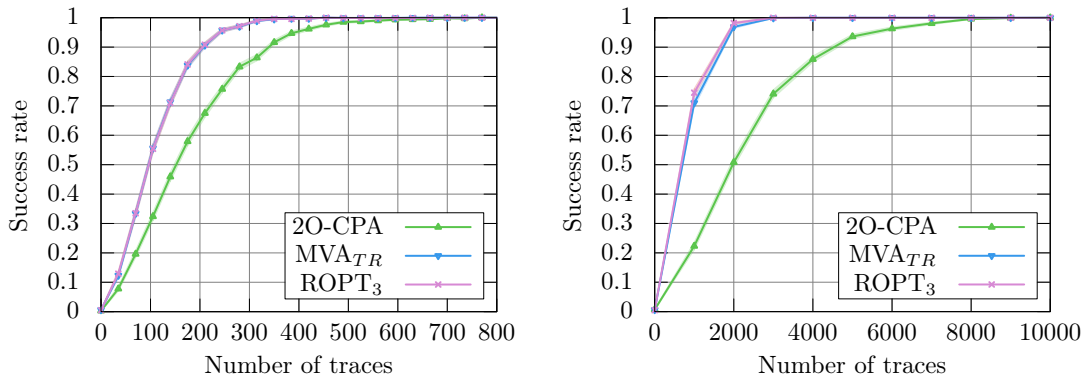
These results can be interpreted as follows: The MVA_{TR} is a third order attack which depends on the third order moment. The 2O-CPA is a second order attack which depends on the second order moment. The new ROPT_3 attack combines these two moments. When the noise is low the MVA_{TR} and the ROPT_3 performs similarly; this shows that the dominant term in the Taylor expansion is the third order one. At the opposite when the noise increases the ROPT_3 becomes close to the 2O-CPA which indicates that the important term in the Taylor expansion is the second order one. As ROPT_3 combines the second and the third order moment weighted by the SNR it is always better than any attack exploiting only one moment.

6.7 Conclusions and Perspectives

In this chapter, we derived new attacks based on the Lth degree Taylor expansion in the SNR of the optimal Maximum Likelihood distinguisher. We have shown that this Lth degree truncation allows to target a moment of order L. The new attack outperforms the optimal distinguisher with respect to time complexity. In fact as we have theoretically shown, the Taylor approximation can be effectively computed whereas the fully optimal maximum likelihood distinguisher, was not computationally tractable.

We have illustrated this property by applying our new method in a complex scenario of "shuffled table recomputation" and have compared the time complexity of the new attack and the optimal distinguisher. In addition, we have shown that in this context our attack has a higher success rate than all the attacks of the state-of-art over all possible noise variances.

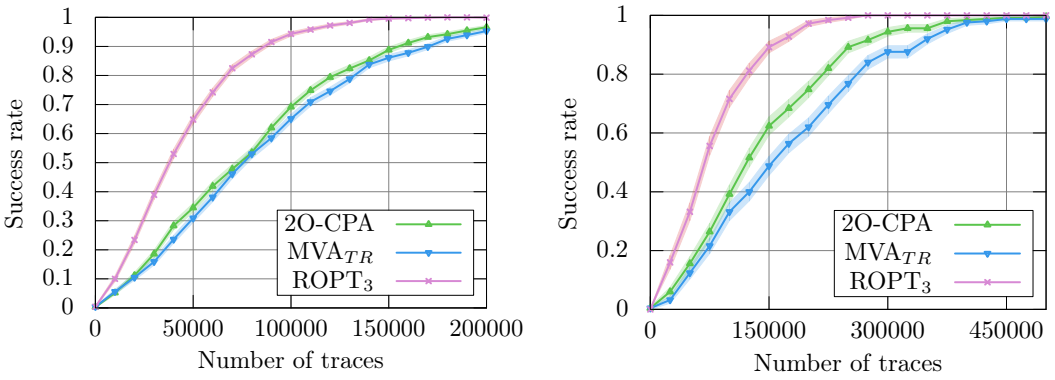
An open question is how to quantify the accuracy of the approximation $\text{LL} \longrightarrow \text{LL}_\ell$ as a function of the noise. In other words, what is the optimal degree of the Taylor expansion



(a) $\sigma = 1$ **(b)** $\sigma = 3$

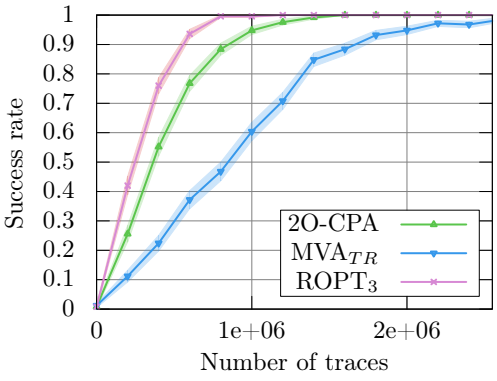
(c) $\sigma = 6$ **(d)** $\sigma = 7$

Figure 6.3: Attack on shuffled table recomputation (Low Noises)

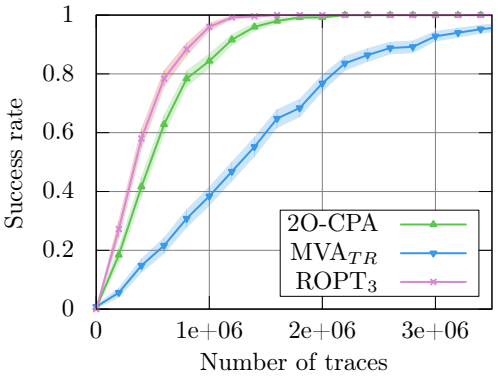


(a) $\sigma = 8$

(b) $\sigma = 9$



(c) $\sigma = 12$



(d) $\sigma = 13$

Figure 6.4: Attack on shuffled table recomputation (High Noises)

of the likelihood for a given SNR? Another interesting extension of this framework would be on hardware devices which are known to leak at various orders (see the real-world examples in (111, 113, 114)).

CHAPTER 7

Conclusion

Contents

7.1 Conclusion

Side-Channel Analysis are still a very dynamic research area in which new results are continuously published. In this manuscript we proposed different ways to improve the results of SCA. We have both explored them in empirical and theoretical manners. We were especially interested in the exploitation of multiple leakages to improve the success of the SCA. We have shown that this behavior of Side channel measurements is a meaningful way to improve the attacks.

In particular in the first part of this manuscript (Part I) we have shown the possible exploitations of multiple leakages of a unique variable. In Chap. 2 we studied the optimal way to exploit this kind of leakages in order to maximize the Success Rate in a profiling scenario. We have shown that in the context of a powerful attacker with a full knowledge of the leakage function it coincides with a dimensionality reduction. In this context we gave the exact formula of this reduction and linked it with two classical dimensionality reduction tools the PCA and the LRA. Specifically we have shown that our new dimensionality reduction is asymptotically

equivalent to the LRA. In Chap. 3 we analyzed the optimal dimensionality reduction in the case of a less powerful attacker. In this scenario the attacker has no a priori knowledge on the leakage function. This generalization of the attacker model comes with a less powerful result. Indeed in this chapter the optimization problem is the maximization of the distinguisher. Moreover we took as case-study the High-Order attacks against masking scheme while the state-of-the-art attack mainly deals with univariate non protected implementations. As a consequence we investigated in this chapter the dimensionality reduction in the case of 2-variate attacks. In this context the rational of the dimension reduction may differ to the univariate one. Especially we showed that the gain of the uses of dimensionality reduction increases with the order. As a consequence such methods are a powerful tool against protected implementations. In these two scenarios the methods presented increase significantly the results of the attack. This means that both in non protected and protected implementations, an increase of the dimension of the attacks improves their results. We showed these results in 1-variate and 2-variate scenario and theoretically extended these results at any orders.

In the second part of this manuscript (Part II) we investigated the possible exploitation of the leakages of multiple variables. In the Chap. 4 we extended the results of the state-of-the-art by deriving the optimal attack against a masking scheme with a table recomputation step. In particular we showed that in this case the exploitation of the multiple leakages of the table recomputation can greatly improve the results of the attacks. Indeed the new attacks offer better results in success probability than the different attacks of the state-of-the-art. This theoretical and empirical analysis provides a better understanding on the construction of highly multivariate attacks. Indeed we showed that the optimal attack takes into account all the leakages in once. Moreover we showed that for high noises this new attack is closed to the "mean" of different attacks. As a consequence the optimal attack is closed to a multi-target attack. The multi-target attacks represent an interesting research path. The implementations of the masking schemes with a table recomputation step can be protected, as a consequence, we presented in Chap. 5 a new multivariate attack tailored to defeat such countermeasures. This new attack is the first example of a non minimal order attack more efficient that the minimum one. In other words we showed that by combining more variables than the minimum number needed to recover a key depend variable, we could build better attacks. This means that in presence of masking scheme it may have an interest to build highly multivariate attacks. As a consequence in order to properly evaluate the security of the masked implementation we showed that the order is not a sufficient metric. In this context we extended this attack for different leakage

functions and different masking schemes at different orders. Finally we additionally gave some countermeasures. In Chap. 6 we theoretically investigated how far an attacker can exploit these leakages. As a consequence we put themselves in the case of an attacker with full knowledge of the models. In this context we showed that the main issue is the complexity of the optimal attack. In order to avoid this, we presented a new attack based on a truncated version of the Taylor Expansion of the Maximum Likelihood. This new attack retains the efficiency property of the Maximum Likelihood by completely out performing the results of all known attacks. As a truncated version this new attack has a better effectiveness and as a consequence is computable. Moreover it gives interesting insight on the behavior High Order attacks. Indeed contrary to the state-of-the-art attack a better attack will take into account the leakages at different orders and combined them using the SNR as weight. These features highlight the fact these attacks are close to the combinations of attacks at different orders. They exploit different combination of variables. As a consequence these attacks are multi-target.

In this study we showed that the number of exploitable leakage samples is a fundamental parameter in the evaluation of the security of the implementation of cryptographic algorithms. Especially we showed that the order of the implementation of the masking scheme is not a sufficient criterion to ensure the security of protected implementations.

7.2 Perspectives.

7.2.1 Under Review.

During this thesis we investigated some other research paths which are not part of this manuscript and are now under review.

Success Exponent. The first one is to identify the relevant parameters of the success of the SCA in cases of protected implementations. In order to investigate these parameters an interesting way is to extent the results of the Success Exponent in the cases of protected implementations. We already extended these results in the case of High Order masking schemes, shuffling implementations and noise addition. In this context we showed that the SNR of each shares play a fundamental role. As in the unprotected cases the confusion coefficient is another important parameter. As already mentioned in the state-of-the-art the impact of the order is in the number of products needed to recover a value depending on the key. As a consequence the maximum degree of the polynomial in the noise variance will depend on the order of the

implementation of the masking schemes. The overall gain of the exploitation of the multiple leakages will be then in the coefficient of the polynomial.

Horizontal Attacks. It is well-known that the leakages of multiple variables are an effective way to attack protected implementations of asymmetric cryptography algorithms. Indeed the so called "horizontal" attacks are ones of the most effective ways to defeat such implementations. During this thesis we investigated how these attacks can be improved using machine learning methods. We showed that the uses of particular distances in cluster algorithms may improve the results of attacks based on clustering.

Stochastic Collision Attacks. We showed that in order to attack protected implementations multi-target attacks are an efficient tool. These attacks combined the leakages of the key dependent variables but also the variables involved in the countermeasures. A classical kind of attacks which exploit several variables in order to break protected implementation are the so called *collision* attacks. One our current work introduces the so called *stochastic collision* which combined the behaviors of collision attacks and stochastic attacks. Interestingly this new attack can be applied on masking schemes. The simulation results show that this new attack gives better results than the state-of-the-art collision attacks.

7.2.2 Research Perspectives.

In this thesis we showed that the exploitation of multiple leakages improves the result of SCA. Interestingly we showed that the parameters of the multiplicity of leakages have a greater impact than the order of the implementation. As a consequence the order has to be be seen more as a design parameter than a security parameter even if of course it has an impact of the security of the implementation. As the order is not a sufficient parameter to evaluate the security of protected implementations, the security should be established using other parameters. Specifically it should be based on the number of leakages, the number of variables and their possible combinations. Any security evaluation should be based on a detailed analysis of the masking schemes where all combinations of variables should be considered.

Additionally an interesting perspective is to combined, in the context of protection implementation, the different methods presented in this manuscript with the recent results of multi-target attacks. In this context exploiting the multiple, possibly masked, leakages of the different parts of the cipher algorithm should lead to significant improvements.

In this manuscript we showed that a possible approach is to perform unified attacks using the Maximum Likelihood. An interesting research path would be to generalize this approach for any protected implementations noticing that in some cases it may be not sufficient or not computable. In these cases truncated versions may be an interesting approach.

The theoretical approach based on the Success Exponent can be an interesting first step for further researches. Indeed the expression of the theoretical Success Rate will allow to highlight the different behaviors of different attacks against different implementations. Such approaches will lead to fairer comparisons as it will take into account more parameters than straightforward comparisons based for example only on the order or the SNR.

Another question is how this result can be extended on other secure implementations? Especially are the others masking schemes vulnerable against attacks exploiting multiple leakages? If it is the case, how their security is impacted compared to their traditional security parameters?

Generally we showed in this manuscript that the use of the multiple leakages improve the results of the SCA. We introduce different parameters to classify these leakages depending on the multiplicity of different variables or the way there are exploited to recover the key. We saw that the way multiple leakages are exploited depends on these parameters. A possible research path is to optimally derived the way to take into account this multiplicity of type of leakages. By doing so it may possible to identify a generic framework to exploit the multiple leakages and as a consequence establish a parameter to evaluate the security of protected implementations.

7.3 List of publications.

During this thesis I contributed to different publications which are not part of this manuscript. In the following we list the contributions.

The following contributions are part of this manuscript:

Multi-Variate High-Order Attacks of Shuffled Tables Recomputation, *Nicolas Bruneau, Sylvain Guilley, Zakaria Najm and Yannick Teglia*, to appear in Journal of Cryptology.

Abstract

Masking schemes based on tables recomputation are classical countermeasures against high-order side-channel attacks. Still, they are known to be attackable at order d in the case the masking involves d shares. In this work, we mathematically show that an attack of order strictly greater than d can be more successful than an

attack at order d. To do so, we leverage the idea presented by Tunstall, Whitnall and Oswald at FSE 2013: we exhibit attacks which exploit the multiple leakages linked to one mask during the recomputation of tables. Specifically, regarding first-order table recomputation, improved by a shuffled execution, we show that there is a window of opportunity, in terms of noise variance, where a novel highly multivariate third-order attack is more efficient than a classical bivariate second-order attack. Moreover, we show on the example of the high-order secure table computation presented by Coron at EUROCRYPT 2014 that the window of opportunity enlarges linearly with the security order d. These results extend that of the CHES '15 eponymous paper. Here, we also investigate the case of degree one leakage models, and formally show that the Hamming weight model is the less favorable to the attacker. Eventually, we validate our attack on a real ATMEL smartcard.

Keywords: Shuffled table recomputation, highly multivariate high-order attacks, signal-to-noise ratio.

Taylor Expansion of Maximum Likelihood Attacks for Masked and Shuffled Implementations, *Nicolas Bruneau, Sylvain Guilley, Annelie Heuser, Olivier Rioul, François-Xavier Standaert and Yannick Teglia*, appeared in Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security.

Abstract

The maximum likelihood side-channel distinguisher of a template attack scenario is expanded into lower degree attacks according to the increasing powers of the signal-to-noise ratio (SNR). By exploiting this decomposition we show that it is possible to build highly multivariate attacks which remain efficient when the likelihood cannot be computed in practice due to its computational complexity. The shuffled table recomputation is used as an illustration to derive a new attack which outperforms the ones presented by Bruneau et al. at CHES 2015, and so across the full range of SNRs. This attack combines two attack degrees and is able to exploit high dimensional leakage which explains its efficiency.

Keywords: Template Attacks, Taylor expansion, Shuffled table recomputation.

Less is More - Dimensionality Reduction from a Theoretical Perspective, *Nicolas*

Bruneau, Sylvain Guilley, Annelie Heuser, Damien Marion and Olivier Rioul, appeared in Cryptographic Hardware and Embedded Systems - CHES 2015 - 17th International Workshop.

Abstract

Reducing the dimensionality of the measurements is an important problem in side-channel analysis. It allows to capture multi-dimensional leakage as one single compressed sample, and therefore also helps to reduce the computational complexity. The other side of the coin with dimensionality reduction is that it may at the same time reduce the efficiency of the attack, in terms of success probability.

In this paper, we carry out a mathematical analysis of dimensionality reduction. We show that optimal attacks remain optimal after a first pass of preprocessing, which takes the form of a linear projection of the samples. We then investigate the state-of-the-art dimensionality reduction techniques, and find that asymptotically, the optimal strategy coincides with the linear discriminant analysis.

Keywords: Dimentionality Reduction, Side Channel Analysis.

Multi-Variate High-Order Attacks of Shuffled Tables Recomputation, *Nicolas Bruneau, Sylvain Guilley, Zakaria Najm and Yannick Teglia*, appeared in Cryptographic Hardware and Embedded Systems - CHES 2015 - 17th International Workshop.

Abstract

Masking schemes based on tables recomputation are classical countermeasures against high-order side-channel attacks. Still, they are known to be attackable at order d in the case the masking involves d shares. In this work, we mathematically show that an attack of order strictly greater than d can be more successful than an attack at order d. To do so, we leverage the idea presented by Tunstall, Whitnall and Oswald at FSE 2013: we exhibit attacks which exploit the multiple leakages linked to one mask during the recomputation of tables. Specifically, regarding first-order table recomputation, improved by a shuffled execution, we show that there is a window of opportunity, in terms of noise variance, where a novel highly multivariate third-order attack is more efficient than a classical bivariate second-order attack. Moreover, we show on the example of the high-order secure table computation presented by Coron at EUROCRYPT 2014 that the window of opportunity enlarges linearly with the security order d.

Keywords: Shuffled table recomputation, highly multivariate high-order attacks, signal-to-noise ratio.

Masks will Fall Off, *Nicolas Bruneau, Sylvain Guilley, Annelie Heuser and Olivier Rioul*, appeared in Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security.

Abstract

Higher-order side-channel attacks are able to break the security of cryptographic implementations even if they are protected with masking countermeasures. In this paper, we derive the best possible distinguishers (High-Order Optimal Distinguishers or HOOD) against masking schemes under the assumption that the attacker can profile. Our exact derivation admits simple approximate expressions for high and low noise and shows to which extent the optimal distinguishers reduce to known attacks in the case where no profiling is possible. From these results, we can explain theoretically the empirical outcome of recent works on second-order distinguishers. In addition, we extend our analysis to any order and to the application to masked tables precomputation. Our results give some insight on which distinguishers have to be considered in the security analysis of cryptographic devices.

Keywords: Side-channel analysis, higher-order masking, masking tables, higher-order optimal distinguisher (HOOD), template attack.

Boosting Higher-Order Correlation Attacks by Dimensionality Reduction, *Nicolas Bruneau, Jean-Luc Danger, Annelie Heuser and Yannick Teglia*, appeared in Security, Privacy, and Applied Cryptography Engineering - 4th International Conference, SPACE 2014.

Abstract

Multi-variate side-channel attacks allow to break higher-order masking protections by combining several leakage samples. But how to optimally extract all the information contained in all possible d-tuples of points? In this article, we introduce preprocessing tools that answer this question. We first show that maximizing the higher-order CPA coefficient is equivalent to finding the maximum of the covariance. We apply this equivalence to the problem of trace dimensionality reduction by linear combination of its samples. Then we establish the link between this problem and

the Principal Component Analysis. In a second step we present the optimal solution for the problem of maximizing the covariance. We also theoretically and empirically compare these methods. We finally apply them on real measurements, publicly available under the DPA Contest v4, to evaluate how the proposed techniques improve the second-order CPA (2O-CPA).

Keywords: Bi-variate attacks, second-order correlation power analysis (2O-CPA), principal component analysis, interclass variance, covariance vector.

The following contributions are not part of this manuscript:

Analysis and Improvements of the DPA Contest v4 Implementation, *Shivam Bhasin,Nicolas Bruneau, Jean-Luc Danger, Sylvain Guilley and Zakaria Najm*, appeared in Security, Privacy, and Applied Cryptography Engineering - 4th International Conference, SPACE 2014.

Abstract

DPA Contest is an international framework which allows researchers to compare their attacks under a common setting. The latest version of DPA Contest proposes a software implementation of AES-256 protected with a low-entropy masking scheme. The masking scheme is called Rotating Sbox Masking (RSM) which claims first-degree security. In this paper, we review the attacks submitted against DPA Contest v4 implementation to identify the common loop holes in the proposed implementation. Next we propose some ideas to improve the existing implementation to resist most of the proposed attacks at affordable performance overhead. Finally we compare our implementation with the original proposal in terms of complexity and side-channel leakage.

Keywords: Side Channel Attacks, DPA Contest, Low Entropy Masking Schemes Shuffling.

Time-Frequency Analysis for Second-Order Attacks, *Pierre Belgarric, Shivam Bhasin, Nicolas Bruneau, Jean-Luc Danger, Nicolas Debande, Sylvain Guilley, Annelie Heuser, Zakaria Najm and Olivier Rioul*, appeared in mart Card Research and Advanced Applications - 12th International Conference, CARDIS 2013.

Abstract

Second-order side-channel attacks are used to break first-order masking protections. A practical reason which often limits the efficiency of second-order attacks is the temporal localisation of the leaking samples. Several leakage samples must be combined which means high computational power. For second-order attacks, the computational complexity is quadratic. At CHES '04, Waddle and Wagner introduced attacks with complexity $\mathcal{O}(n \log_2 n)$ on hardware traces, where n is the window size, by working on traces auto-correlation. Nonetheless, the two samples must belong to the same window which is (normally) not the case for software implementations. In this article, we introduce preprocessing tools that improve the efficiency of bi-variate attacks (while keeping a complexity of $\mathcal{O}(n \log_2 n)$), even if the two samples that leak are far away one from the other (as in software). We put forward two main improvements. Firstly, we introduce a method to avoid losing the phase information. Next, we empirically notice that keeping the analysis in the frequency domain can be beneficial for the attack. We apply these attacks in practice on real measurements, publicly available under the DPA Contest v4, to evaluate the proposed techniques. An attack using a window as large as 4000 points is able to reveal the key in only 3000 traces.

Keywords: Bi-variate attacks, zero-offset 2O-CPA, discrete Hartley transform, leakage in phase.

Part III

Appendix.

APPENDIX A

Appendix of Dimensionality Reduction

A.1 Proof of Theorem 3.2.1

Proof. On the one side we have

$$\begin{aligned}\mathsf{Cov}\left[\alpha\cdot X,\widehat{\Psi}(Z)\right] &= \left(\mathsf{Cov}\left[S^{(d)}+N^{(d)},\widehat{\Psi}(Z)\right]\right)_{d\in D}\cdot\alpha\\ &= \alpha\cdot\left(\mathsf{Cov}\left[S^{(d)},\widehat{\Psi}(Z)\right]\right)_{d\in D}\\ &= \alpha\cdot\left(\mathbb{E}\left[S^{(d)}\widehat{\Psi}(Z)\right]\right)_{d\in D}\ .\end{aligned}$$

The other side yields $\mathrm{Var}\left[\alpha\cdot\mathbb{E}\left[X|\widehat{\Psi}(Z)\right]\right] = \mathrm{Var}\left[\alpha\cdot\left(S^{(d)}\right)_{d\in D}\right]$. Now if $S^{(d)} = \beta^{(d)}\widehat{\Psi}(Z)$, then we have for both sides

$$\begin{cases}\mathsf{Cov}\left[\alpha\cdot X;\widehat{\Psi}(Z)\right]^2 &= (\alpha\cdot\beta)^2\,\mathbb{E}\left[\widehat{\Psi}(Z)^2\right]^2,\\ \mathrm{Var}\left[\alpha\cdot\mathbb{E}\left[X|\widehat{\Psi}(Z)\right]\right] &= \mathrm{Var}\left[(\alpha\cdot\beta)\,\widehat{\Psi}(Z)\right] = (\alpha\cdot\beta)^2\,\mathbb{E}\left[\widehat{\Psi}(Z)^2\right],\end{cases}$$

which proves equivalence. □

A.2 Proof of Lemma 2

Proof.

$$\begin{aligned}\underset{\|\alpha\|=1}{\mathrm{argmax}}\,\mathsf{Cov}\left[\alpha\cdot X,\widehat{\Psi}(Z)\right]^2 &= \underset{\|\alpha\|=1}{\mathrm{argmax}}\,(\alpha\cdot\beta)^2\,\mathbb{E}\left[\widehat{\Psi}(Z)^2\right]^2\\ &= \underset{\|\alpha\|=1}{\mathrm{argmax}}\,(\alpha\cdot\beta)^2\text{, because }\mathbb{E}\left[\widehat{\Psi}(Z)^2\right]^2>0.\end{aligned}$$

By the Cauchy-Schwarz theorem, we have: $(\alpha \cdot \beta)^2 \leqslant \|\alpha\|^2 \times \|\beta\|^2$,where equality holds if and only if α and β are linearly dependent, i.e., $\alpha = \lambda\beta$. Accordingly, if $\|\alpha\| = 1$ we have $\lambda = \frac{1}{\|\beta\|}$, which gives us the required solution. □

A.3 Proof of Proposition 14

Proof. We have

$$\mathsf{Cov}\left[\alpha \cdot X, \widehat{\Psi}(Z)\right] = \alpha \cdot \left(\mathsf{Cov}\left[X^{(d)}; \widehat{\Psi}(Z)\right]\right)_{d \in D} \quad .$$

Similar to the proof of Lemma 2, we use the Cauchy-Schwarz inequality. In particular,

$$\left(\alpha \cdot \left(\mathsf{Cov}\left[X^{(d)}; \widehat{\Psi}(Z)\right]\right)_{d \in D}\right)^2 \leqslant \|\alpha\|^2 \times \|\left(\mathsf{Cov}\left[X^{(d)}; \widehat{\Psi}(Z)\right]\right)_{d \in D}\|^2.$$

We have the equality,

$$\left(\alpha \cdot \left(\mathsf{Cov}\left[X^{(d)}; \widehat{\Psi}(Z)\right]\right)_{d \in D}\right)^2 = \|\alpha\|^2 \times \|\left(\mathsf{Cov}\left[X^{(d)}; \widehat{\Psi}(Z)\right]\right)_{d \in D}\|^2,$$

if and only if $\alpha = \lambda \left(\mathsf{Cov}\left[X^{(d)}; \widehat{\Psi}(Z)\right]\right)_{d \in D}$.

So, if $\|\alpha\| = 1$ we have $\lambda = \frac{1}{\|\left(\mathsf{Cov}\left[\widehat{\Psi}(Z); \widehat{\Psi}(Z)\right]\right)_{d \in D}\|}$.

□

APPENDIX B

Appendix of Multivariate Attack.

B.1 Proof of Theorem 5.2.1

In order to prove the Theorem 5.2.1 let us first introduce some lemmas. By Remark 19 the only random parts of X_{TR} are the noise and the mask. As a consequence the random variable $(X_{TR}|M=m)$ depends only on the noise, and is equal to:

$$\begin{aligned}(X_{TR}|M=m) = -2 \times \frac{1}{2^n} \sum_{\omega=0}^{2^n-1} \Bigg[\Big(\mathrm{HW}[\Phi(\omega) \oplus m] + N_\omega^{(1)} - \frac{n}{2} \Big) \\ \times \Big(\mathrm{HW}[\Phi(\omega)] + N_\omega^{(2)} - \frac{n}{2} \Big) \Bigg] \ .\end{aligned} \tag{B.1}$$

Lemma 8.

$$\begin{aligned}(X_{TR}|M=m) = \ & \mathrm{HW}[m] - \frac{n}{2} \\ & - 2 \times \frac{1}{2^n} \sum_{\omega=0}^{2^n-1} \Big[N_\omega^{(1)} \times \Big(\mathrm{HW}[\Phi(\omega)] - \frac{n}{2} \Big) \Big] \\ & - 2 \times \frac{1}{2^n} \sum_{\omega=0}^{2^n-1} \Big[N_\omega^{(2)} \times \Big(\mathrm{HW}[\Phi(\omega) \oplus m] - \frac{n}{2} \Big) \Big] \\ & - 2 \times \frac{1}{2^n} \sum_{\omega=0}^{2^n-1} \Big[N_\omega^{(1)} \times N_\omega^{(2)} \Big] \ .\end{aligned} \tag{B.2}$$

Proof. $(X_{TR}|M=m)$ can be split into a deterministic part and a random part:

$$(X_{TR}|M=m) = -2 \times (S_d + S_r) \ ,$$

where

$$S_d = \frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[\left(\mathrm{HW}[\Phi(\omega)\oplus m]-\frac{n}{2}\right)\times\left(\mathrm{HW}[\Phi(\omega)]-\frac{n}{2}\right)\right] ,$$

$$\begin{aligned} S_r = &\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times\left(\mathrm{HW}[\Phi(\omega)]-\frac{n}{2}\right)\right] \\ &+\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(2)}\times\left(\mathrm{HW}[\Phi(\omega)\oplus m]-\frac{n}{2}\right)\right] \\ &+\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times N_\omega^{(2)}\right] . \end{aligned}$$

$$\begin{aligned} S_d &= \mathbb{E}\left[(\mathrm{HW}[U\oplus M]-\mathbb{E}\left[\mathrm{HW}[U\oplus M]\right])\times(\mathrm{HW}[U]-\mathbb{E}\left[\mathrm{HW}[U\oplus M]\right])\mid M=m\right] \\ &= -\frac{1}{2}\mathrm{HW}[m]+\frac{n}{4} \qquad \text{by (133)} , \end{aligned}$$

where U denotes a random variable drawn uniformly over $\mathbb{F}_2^n$. □

Lemma 9.

$$Var[(X_{TR}|M=m)] = 4\times\left(\frac{\sigma^2}{2^n}\times\frac{n}{2}+\frac{\sigma^4}{2^n}\right) . \tag{B.3}$$

Proof. Recall that the random variable $(X_{TR}|M=m)$ can be write as in Lemma 8; thus $\mathrm{Var}\left[(X_{TR}|M=m)\right] = 4\times(V_1+V_2+V_3+C_1+C_2+C_3)$, where

$$V_1 = \mathrm{Var}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times\left(\mathrm{HW}[\Phi(\omega)]-\frac{n}{2}\right)\right]\right] ,$$

$$V_2 = \mathrm{Var}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(2)}\times\left(\mathrm{HW}[\Phi(\omega)\oplus m]-\frac{n}{2}\right)\right]\right] ,$$

$$V_3 = \mathrm{Var}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times N_\omega^{(2)}\right]\right] ,$$

$$C_1 = 2\times\mathsf{Cov}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times\left(\mathrm{HW}[\Phi(\omega)]-\frac{n}{2}\right)\right],\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times N_\omega^{(2)}\right]\right] ,$$

$$C_2 = 2\times\mathsf{Cov}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(2)}\times\left(\mathrm{HW}[\Phi(\omega)\oplus m]-\frac{n}{2}\right)\right],\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times N_\omega^{(2)}\right]\right] ,$$

$$\begin{aligned} C_3 = 2\times\mathsf{Cov}\Bigg[&\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times\left(\mathrm{HW}[\Phi(\omega)]-\frac{n}{2}\right)\right] , \\ &\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(2)}\times\left(\mathrm{HW}[\Phi(\omega)\oplus m]-\frac{n}{2}\right)\right]\Bigg] . \end{aligned}$$

Let us now prove that $C_1 = C_2 = 0$. First we have:

$$\mathsf{Cov}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times\left(\mathrm{HW}[\Phi(\omega)]-\frac{n}{2}\right)\right],\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times N_\omega^{(2)}\right]\right] = C_1^{(1)}-C_1^{(2)},$$

with

$$C_1^{(1)} = \mathsf{Cov}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[\mathrm{HW}[\Phi(\omega)]\times N_\omega^{(1)}\right], \frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times N_\omega^{(2)}\right]\right]$$
$$= \frac{1}{2^n}\sum_{\omega'=0}^{2^n-1}\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[\mathsf{Cov}\left[\mathrm{HW}[\Phi(\omega)]\times N_\omega^{(1)}, N_{\omega'}^{(1)}\times N_{\omega'}^{(2)}\right]\right] \ .$$

The random variables $N_\omega^{(i)}$, where $i\in\{1,2\}$ and $\omega\in\mathbb{F}_2^n$ are mutually independent and independent with all the $\mathrm{HW}[\Phi(\omega)]$. Thus we have:

$$\forall\omega,\omega', \mathsf{Cov}\left[\mathrm{HW}[\Phi(\omega)]\times N_\omega^{(1)}, N_{\omega'}^{(1)}\times N_{\omega'}^{(2)}\right] = 0$$
$$\iff \frac{1}{2^n}\sum_{\omega'=0}^{2^n-1}\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[\mathsf{Cov}\left[\mathrm{HW}[\Phi(\omega)]\times N_\omega^{(1)}, N_{\omega'}^{(1)}\times N_{\omega'}^{(2)}\right]\right] = 0$$
$$\iff C_1^{(1)} = 0 \ .$$

Besides

$$C_1^{(2)} = \mathsf{Cov}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[\frac{n}{2}\times N_\omega^{(1)}\right], \frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times N_\omega^{(2)}\right]\right]$$
$$= \frac{n}{2}\times\frac{1}{2^n}\sum_{\omega'=0}^{2^n-1}\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[\mathsf{Cov}\left[N_\omega^{(1)}, N_{\omega'}^{(1)}\times N_{\omega'}^{(2)}\right]\right] \ .$$

As $N_\omega^{(i)}$, where $i\in\{1,2\}$ and $\omega\in\mathbb{F}_2^n$, are mutually independent, we have:

$$\mathsf{Cov}\left[N_\omega^{(1)}, N_{\omega'}^{(1)}\times N_{\omega'}^{(2)}\right] = 0, \ \forall(\omega,\omega')\in\mathbb{F}_2^n\times\mathbb{F}_2^n$$
$$\iff C_1^{(2)} = \mathsf{Cov}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[\frac{n}{2}\times N_\omega^{(1)}\right], \frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times N_\omega^{(2)}\right]\right] = 0$$
$$\iff \mathsf{Cov}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}N_\omega^{(1)}\times\left(\mathrm{HW}[\Phi(\omega)]-\frac{n}{2}\right), \frac{1}{2^n}\sum_{\omega=0}^{n-1}N_\omega^{(1)}\times N_\omega^{(2)}\right] = 0 \ .$$

Identically we prove that:

$$\mathsf{Cov}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(2)}\times\left(\mathrm{HW}[\Phi(\omega)\oplus m]-\frac{n}{2}\right)\right], \frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)}\times N_\omega^{(2)}\right]\right] = 0 \ .$$

As a consequence $C_1 = C_2 = 0$. Let us now study C_3. By the bi-linearity of the covariance C_3 can be rewritten such that:

$$C_3 = \frac{2}{2^{2n}}\sum_{\omega=0}^{2^n-1}\sum_{\omega'=0}^{2^n-1}\mathsf{Cov}\left[N_\omega^{(1)}\times\left(\mathrm{HW}[\Phi(\omega)]-\frac{n}{2}\right), N_{\omega'}^{(2)}\times\left(\mathrm{HW}[\Phi(\omega)\oplus m]-\frac{n}{2}\right)\right].$$

But

$$\begin{aligned}
&\mathsf{Cov}\left[N_\omega^{(1)} \times \left(\mathrm{HW}[\Phi(\omega)] - \frac{n}{2}\right), N_{\omega'}^{(2)} \times \left(\mathrm{HW}[\Phi(\omega) \oplus m] - \frac{n}{2}\right)\right] \\
&= \mathbb{E}\left[N_\omega^{(1)} \times \left(\mathrm{HW}[\Phi(\omega)] - \frac{n}{2}\right) \times N_{\omega'}^{(2)} \times \left(\mathrm{HW}[\Phi(\omega) \oplus m] - \frac{n}{2}\right)\right] \\
&- \mathbb{E}\left[N_\omega^{(1)} \times \left(\mathrm{HW}[\Phi(\omega)] - \frac{n}{2}\right)\right] \times \mathbb{E}\left[N_{\omega'}^{(2)} \times \left(\mathrm{HW}[\Phi(\omega) \oplus m] - \frac{n}{2}\right)\right] \ .
\end{aligned}$$

By definition, $N_\omega^{(1)}$ is independent from $\mathrm{HW}[\Phi(\omega)]$. Thus:

$$\begin{aligned}
&\mathbb{E}\left[N_\omega^{(1)} \times \left(\mathrm{HW}[\Phi(\omega)] - \frac{n}{2}\right)\right] = \mathbb{E}\left[N_\omega^{(1)}\right] \times \mathbb{E}\left[\left(\mathrm{HW}[\Phi(\omega)] - \frac{n}{2}\right)\right] = 0 \text{ and} \\
&\mathbb{E}\left[N_\omega^{(1)} \times \left(\mathrm{HW}[\Phi(\omega)] - \frac{n}{2}\right)\right] \times \mathbb{E}\left[N_{\omega'}^{(2)} \times \left(\mathrm{HW}[\Phi(\omega) \oplus m] - \frac{n}{2}\right)\right] = 0 \ .
\end{aligned}$$

$N_\omega^{(1)}$ is independent from $\left(\mathrm{HW}[\Phi(\omega)] - \frac{n}{2}\right) \times N_{\omega'}^{(2)} \times \left(\mathrm{HW}[\Phi(\omega) \oplus m] - \frac{n}{2}\right)$. Thus $\mathbb{E}\left[N_\omega^{(1)} \times \left(\mathrm{HW}[\Phi(\omega)] - \frac{n}{2}\right) \times N_{\omega'}^{(2)} \times (\mathrm{HW}\right.$ 0, which implies that $C_3 = 0$. As a consequence $\mathrm{Var}\left[(X_{TR}|M = m)\right] = 4 \times (V_1 + V_2 + V_3)$.

$$\begin{aligned}
V_1 &= \mathrm{Var}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(1)} \times \left(\mathrm{HW}[\Phi(\omega)] - \frac{n}{2}\right)\right]\right] \\
&= \frac{1}{2^{2n}}\sum_{\omega=0}^{2^n-1}\mathrm{Var}\left[N_\omega^{(1)} \times \left(\mathrm{HW}[\Phi(\omega)] - \frac{n}{2}\right)\right] \\
&\quad + \frac{2}{2^{2n}}\sum_{0 \leqslant \omega < \omega' \leqslant 2^n-1} \mathsf{Cov}\left[N_\omega^{(1)} \times (\mathrm{HW}[\Phi(\omega)]), N_{\omega'}^{(1)} \times (\mathrm{HW}[\Phi(\omega')])\right] \ . \qquad \text{(B.4)}
\end{aligned}$$

As $\mathsf{Cov}\left[N_\omega^{(1)} \times (\mathrm{HW}[\Phi(\omega)]), N_{\omega'}^{(1)} \times (\mathrm{HW}[\Phi(\omega')])\right] = 0$, the terms in Eq. (B.4) are all null. It can be noticed that $\mathbb{E}\left[\mathrm{HW}[\Phi(\omega)] - \frac{n}{2}\right] = 0$ as $\Phi(\omega)$ is uniformly distributed over S_{2^n} and $\mathbb{E}\left[N_\omega^{(1)}\right] = 0$. As a consequence:

$$\begin{aligned}
&\mathrm{Var}\left[N_\omega^{(1)} \times \left(\mathrm{HW}[\Phi(\omega)] - \frac{n}{2}\right)\right] = \mathrm{Var}\left[\mathrm{HW}[\Phi(\omega)] - \frac{n}{2}\right] \times \mathrm{Var}\left[N_\omega^{(1)}\right] \text{ , hence} \\
&V_1 = \frac{1}{2^{2n}}\sum_{\omega=0}^{2^n-1}\sigma^2 \times \frac{n}{4} = \frac{1}{2^n} \times \sigma^2 \times \frac{n}{4} \ .
\end{aligned}$$

Identically, we have

$$\begin{aligned}
V_2 &= \mathrm{Var}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\left[N_\omega^{(2)} \times \left(\mathrm{HW}[\Phi(\omega) \oplus m] - \frac{n}{2}\right)\right]\right] = \frac{\sigma^2}{2^n} \times \frac{n}{4} = V_1 \text{ , and} \\
V_3 &= \mathrm{Var}\left[\frac{1}{2^n}\sum_{\omega=0}^{2^n-1} N_\omega^{(1)} \times N_\omega^{(2)}\right] = \frac{\sigma^4}{2^n} \ .
\end{aligned}$$

Finally

$$\mathrm{Var}\left[(X_{TR}|M = m)\right] = 4 \times \left(\frac{\sigma^2}{2^n} \times \frac{n}{2} + \frac{\sigma^4}{2^n}\right) \ .$$

□

Then let us prove the Theorem 5.2.1.

Proof. Lemma 9 gives us the value of the variance of the noise. Then by the definition of the SNR, we have:

$$\begin{aligned}
\mathrm{SNR}\left[X_{TR}, M\right] \geqslant \mathrm{SNR}\left[X^{(3)}, M\right] &\iff \frac{\mathrm{Var}\left[\mathrm{HW}[M]\right]}{\mathrm{Var}\left[(X_{TR}|M=m)\right]} \geqslant \frac{\mathrm{Var}\left[\mathrm{HW}[M]\right]}{\mathrm{Var}\left[N^3\right]} \\
&\iff 4 \times \left(\frac{\sigma^2}{2^n} \times \frac{n}{2} + \frac{\sigma^4}{2^n}\right) \leqslant \sigma^2 \\
&\iff \frac{2^{n-1}-n}{2} \geqslant \sigma^2
\end{aligned}$$

□

B.2 Proof of the propositions of Sect. 5.2.5

B.2.1 Proof of Prop. 19

Proof. By Theorem 2 in (64, Appendix A.2) (extended version of (63)) we have that the SE of is given by

$$\begin{aligned}
\mathrm{SE}_{\text{2O-CPA}} &= \min_{k \neq k^\star} \frac{\kappa\left(k^\star, k\right)}{2\left(\frac{\kappa'(k^\star,k)}{\kappa(k^\star,k)} - \kappa\left(k^\star, k\right)\right) + 2\sum_{\substack{i \in \{0,2\}^d \\ i \neq (0,\ldots,0)}} \prod_{1 \leq \delta \leq d} \left(\alpha_\delta^{-i_\delta} \cdot \sigma_\delta^{i_\delta}\right)} \\
&= \min_{k \neq k^\star} \frac{\kappa\left(k^\star, k\right)}{2\left(\frac{\kappa'(k^\star,k)}{\kappa(k^\star,k)} - \kappa\left(k^\star, k\right)\right) + 2\left(\alpha_1^{-2}\sigma_1^2 + \alpha_2^{-2}\sigma_2^2 + \alpha_1^{-2}\sigma_1^2\alpha_2^{-2}\sigma_2^2\right)}.
\end{aligned}$$

□

B.2.2 Proof of Prop. 21

Proof.

$$\begin{aligned}
\frac{m^{(\mathrm{SR})}_{\text{2O-CPA}} - m^{(\mathrm{SR})}_{\mathrm{MVA}_{TR}}}{m^{(\mathrm{SR})}_{\mathrm{MVA}_{TR}}} &= \left(\frac{\log\left(1-\mathrm{SR}\right)}{\mathrm{SE}_{\text{2O-CPA}}} - \frac{\log\left(1-\mathrm{SR}\right)}{\mathrm{SE}_{\mathrm{MVA}_{TR}}}\right) \times \frac{\mathrm{SE}_{\mathrm{MVA}_{TR}}}{\log\left(1-\mathrm{SR}\right)} \\
&= \frac{\mathrm{SE}_{\mathrm{MVA}_{TR}}}{\mathrm{SE}_{\text{2O-CPA}}} - 1 \ ,
\end{aligned}$$

which indeed does not depend on SR. □

B.2.3 Proof of Prop. 22

Let us now compute the difference of traces needed to reach any SR.

$$m^{(\mathrm{SR})}_{\text{2O-CPA}} - m^{(\mathrm{SR})}_{\mathrm{MVA}_{TR}} = \frac{\log\left(1-\mathrm{SR}\right)}{\mathrm{SE}_{\text{2O-CPA}}} - \frac{\log\left(1-\mathrm{SR}\right)}{\mathrm{SE}_{\mathrm{MVA}_{TR}}}$$

Let us rewrite using $\alpha = \alpha_1 = \alpha_2$. In such case:

$$\begin{aligned} m^{(\mathrm{SR})}_{\text{2O-CPA}} - m^{(\mathrm{SR})}_{\mathrm{MVA}_{TR}} &= \frac{\log(1-\mathrm{SR})}{\mathrm{SE}_{\text{2O-CPA}}} - \frac{\log(1-\mathrm{SR})}{\mathrm{SE}_{\mathrm{MVA}_{TR}}} \\ &= \frac{\log(1-\mathrm{SR})}{\mathrm{SE}_{\text{2O-CPA}}} - \frac{\log(1-\mathrm{SR})}{\mathrm{SE}_{\mathrm{MVA}_{TR}}} \\ &= \left(2\alpha^{-2}\frac{\log(1-\mathrm{SR})}{\kappa(k^\star,k)}\right)\left(1+\alpha^{-2}\sigma^2\right)\left(\sigma^2 - 4\left(\frac{\sigma^2}{2^n}\frac{n}{2} + \frac{\sigma^4}{2^n}\right)\right) \end{aligned}$$

The attacks perform similarly when $m_{\text{2O-CPA}} - m_{\mathrm{MVA}_{TR}} = 0$ which implies $(\sigma^2 - 4 \times \left(\frac{\sigma^2}{2^n} \times \frac{n}{2} + \frac{\sigma^4}{2^n}\right) = 0$. Notice that we recover here the results of the Subsect. 5.2.3.

In order to find the noise when the maximum occurs let us compute the derivative in σ^2:

$$\begin{aligned} &\frac{\Omega\left(m^{(\mathrm{SR})}_{\text{2O-CPA}} - m^{(\mathrm{SR})}_{\mathrm{MVA}_{TR}}\right)}{\Omega\sigma^2} = \\ &\left(\left(\alpha^{-2} - \frac{4\alpha^{-2}}{2^n} \times \frac{n}{2}\right) + \left(\frac{8\alpha^{-2}}{2^n} + 2\alpha^{-4} - \frac{8\alpha^{-4}}{2^n} \times \frac{n}{2}\right)\sigma^2 - \frac{12\alpha^{-4}\sigma^4}{2^n}\right) \\ &\quad \times \left(2\frac{\log(1-\mathrm{SR})}{\kappa(k^\star,k)}\right) \end{aligned}$$

The maximum occurs when $\frac{\Omega\left(m^{(\mathrm{SR})}_{\text{2O-CPA}} - m^{(\mathrm{SR})}_{\mathrm{MVA}_{TR}}\right)}{\Omega\sigma^2} = 0$ which not depends on the SR.

B.3 Proof of Theorem 5.3.1

Similarly to the Remark 19 we have $\forall i < \Omega$:

$$\begin{aligned} \left(X_{CS^\Omega_i} | M_i = m\right) = \frac{-2}{\Omega 2^n} \sum_{\substack{\omega \in \mathbb{F}_{2^n} \\ j \in [\![1,\Omega]\!]}} \Bigg[&\left(\mathrm{HW}[\Phi(\omega) \oplus m] + N^{(1)}_{(\omega,j)} - \frac{n}{2}\right) \\ &\times \left(\mathrm{HW}[\Phi(\omega)] + N^{(2)}_{(\omega,j)} - \frac{n}{2}\right)\Bigg] . \end{aligned}$$

As the i is fixed for each share we have removed it from the index position.

Lemma 10.

$$Var\left[\left(X_{CS^\Omega_i} | M_i = m\right)\right] = 4 \times \left(\frac{\sigma^2}{\Omega \times 2^n} \times \frac{n}{2} + \frac{\sigma^4}{\Omega \times 2^n}\right) , \tag{B.5}$$

where Ω is the number of share of the high-order masking scheme and $i < \Omega$.

Proof. Lemma 10 is a straightforward extension of Lemma 9. □

Exploiting Lemma 10 let us prove Theorem 5.3.1.

Proof. As Lemma 10 gives us the variance of the noise of the second-order leakage we have $\forall i < \Omega$

$$\begin{aligned}
&\mathrm{SNR}\left[X_{CS_i^\Omega}, M_i\right] \geqslant \mathrm{SNR}\left[X_i^{(3)}, M_i\right] \\
&\iff \frac{\mathrm{Var}\left[\mathrm{HW}[M]\right]}{\mathrm{Var}\left[\left(X_{CS_i^\Omega} | M_i = m\right)\right]} \geqslant \frac{\mathrm{Var}\left[\mathrm{HW}[M]\right]}{\mathrm{Var}\left[N_i^{(3)}\right]} \\
&\iff 4 \times \left(\frac{\sigma^2}{\Omega \times 2^n} \times \frac{n}{2} + \frac{\sigma^4}{\Omega \times 2^n}\right) \leqslant \sigma^2 \\
&\iff \left(n - \Omega \times 2^{n-1}\right) \frac{\sigma^2}{\Omega \times 2^{n-1}} + \frac{\sigma^4}{\Omega \times 2^{n-2}} \leqslant 0.
\end{aligned}$$

The upper bound of the interval are the σ^2 where $\sigma^2 \neq 0$ and:

$$\begin{aligned}
\frac{\sigma^4}{\Omega \times 2^{n-2}} &= \left(d \times 2^{n-1} - n\right) \frac{\sigma^2}{\Omega \times 2^{n-1}} \\
&\iff \sigma^2 = \frac{\left(d \times 2^{n-1} - n\right)}{2} \\
&\iff \sigma^2 = d \times 2^{n-2} - \frac{n}{2}.
\end{aligned}$$

It implies that the size of Useful Interval of Variance is given by $d \times 2^{n-2} - \frac{n}{2}$. □

B.4 Affine model

B.4.1 Proof of Lemma 5

Proof.

$$\begin{aligned}
&\mathbb{E}\left[\left(\Psi_\alpha\left(U\right) - \mathbb{E}\left[\Psi_\alpha\left(U\right)\right]\right) \times \left(\Psi_\beta\left(U \oplus z\right) - \mathbb{E}\left[\Psi_\beta\left(U \oplus z\right)\right]\right)\right] \\
&= \mathbb{E}\left[\left(\alpha \cdot U - \alpha \cdot \left(\frac{1}{2}\mathbf{1}\right)\right) \times \left(\beta \cdot (U \oplus z) - \beta \cdot \left(\frac{1}{2}\mathbf{1}\right)\right)\right] \\
&= \mathbb{E}\left[\left(\alpha \cdot \left(U - \frac{1}{2}\mathbf{1}\right)\right) \times \left(\beta \cdot \left((U \oplus z) - \frac{1}{2}\mathbf{1}\right)\right)\right] \\
&= \mathbb{E}\left[\left(\frac{1}{2}\alpha \cdot \overline{U}\right) \times \left(\frac{1}{2}\beta \cdot \left(\overline{U \oplus z}\right)\right)\right] \\
&= \frac{1}{4}\left(\alpha^{\mathsf{t}} \mathbb{E}\left[\overline{U}\left(\overline{U \oplus z}\right)^{\mathsf{t}}\right] \beta\right) \quad ,
\end{aligned}$$

where $\overline{U}$ denotes $2\left(U - \frac{1}{2}\mathbf{1}\right)$.

It can also be noticed that: $\overline{U} = -\left((-1)^{U_1}, \ldots, (-1)^{U_n}\right)$, and thus $\overline{(U \oplus z)} = -\left((-1)^{U_1 + z_1}, \ldots, (-1)^{U_n + z_n}\right)$. Moreover

$$\begin{aligned}
&\mathbb{E}\left[\overline{U}\left(\overline{U \oplus z}\right)^{\mathsf{t}}\right] = \mathsf{Cov}\left[\overline{U}, \left(\overline{U \oplus z}\right)^{\mathsf{t}}\right] \\
\implies &\left(\mathbb{E}\left[\overline{U}\left(\overline{U \oplus z}\right)^{\mathsf{t}}\right]\right)_{i,j} = \mathsf{Cov}\left[(-1)^{U_i}, (-1)^{(U_j + z_j)}\right] \\
\implies &\left(\mathbb{E}\left[\overline{U}\left(\overline{U \oplus z}\right)^{\mathsf{t}}\right]\right)_{i,j} = 0 \text{ if } i \neq j \text{ or } \left(\mathbb{E}\left[\overline{U}\left(\overline{U \oplus z}\right)^{\mathsf{t}}\right]\right)_{i,j} = (-1)^{z_j} \text{ if } i = j \ .
\end{aligned}$$

Eventually, we have:

$$\begin{aligned}
&\mathbb{E}\left[\left(\Psi_\alpha\left(U\right)-\mathbb{E}\left[\Psi_\alpha\left(U\right)\right]\right)\times\left(\Psi_\beta\left(U\oplus z\right)-\mathbb{E}\left[\Psi_\beta\left(U\oplus z\right)\right]\right)\right]\\
&\quad=-\frac{1}{4}\left(\alpha\odot\beta\right)\cdot\overline{z}=-\frac{1}{4}\left(\alpha\odot\beta\right)\cdot 2\left(z-\frac{1}{2}\mathbf{1}\right)\\
&\quad=-\frac{1}{2}\left(\alpha\odot\beta\right)\cdot z+\frac{1}{4}\left(\alpha\odot\beta\right)\cdot\mathbf{1}=-\frac{1}{2}\left(\alpha\odot\beta\right)\cdot z+\frac{1}{4}\alpha\cdot\beta\ .
\end{aligned}$$

□

B.4.2 Proof of the Theorem 5.4.1

Similarly to Eq. (B.1) we have:

$$\begin{aligned}
\left(X_{TR}|M=m\right)=-\,2\times\frac{1}{2^n}\sum_{\omega=0}^{2^n-1}\Bigg[&\left(\alpha\cdot\left(\Phi\left(\omega\right)\oplus m\right)+N_\omega^{(1)}-\frac{1}{2}\left(\alpha\cdot\mathbf{1}\right)\right)\\
&\times\left(\alpha\cdot\left(\Phi\left(\omega\right)\right)+N_\omega^{(2)}-\frac{1}{2}\left(\alpha\cdot\mathbf{1}\right)\right)\Bigg]\ .
\end{aligned}$$

Lemma 11.

$$Var\left[\left(X_{TR}|M=m\right)\right]=4\times\left(\frac{n}{2^{n+1}}\times\sigma^2+\frac{\sigma^4}{2^n}\right)\ .$$

Proof. Similar to proof of Lemma 9 (see Appendix B.1) using the affine model instead of the Hamming Weight and Assumption 1. □

Then we can prove the Theorem 5.4.1.

Proof. Lemma 11 gives us the value of the variance of the noise. Then by the definition of the SNR, we have:

$$\begin{aligned}
&\mathrm{SNR}\left[X_{TR},M\right]\geqslant\mathrm{SNR}\left[X^{(3)},M\right]\\
&\iff\frac{\mathrm{Var}\left[\alpha^2\cdot M\right]}{\mathrm{Var}\left[\left(X_{TR}|M=m\right)\right]}\geqslant\frac{\mathrm{Var}\left[\alpha\cdot M\right]}{\mathrm{Var}\left[N^{(3)}\right]}\\
&\iff\frac{\frac{1}{4}\|\alpha\|_4^4}{4\times\left(\frac{\sigma^2}{2^{n+1}}\times n+\frac{\sigma^4}{2^n}\right)}\geqslant\frac{\frac{1}{4}n}{\sigma^2}\\
&\iff\frac{\sigma^2}{4}\times\|\alpha\|_4^4-\frac{\sigma^2}{2^{n+1}}\times n^2-\frac{\sigma^4}{2^n}\times n\geqslant 0\\
&\iff\sigma^2\times\left(\frac{1}{4}\times\|\alpha\|_4^4-\frac{1}{2^{n+1}}\times n^2-\frac{\sigma^2}{2^n}\times n\right)\geqslant 0\\
&\iff\frac{1}{4}\times\|\alpha\|_4^4-\frac{1}{2^{n+1}}\times n^2-\frac{\sigma^2}{2^n}\times\|\alpha\|_2^2\geqslant 0\\
&\iff\frac{2^n}{4}\times\frac{\|\alpha\|_4^4}{n}-\frac{2^n}{2^{n+1}}\times\frac{n^2}{n}\geqslant\sigma^2\\
&\iff\|\alpha\|_4^4\times\frac{2^{n-2}}{n}-\frac{n}{2}\geqslant\sigma^2
\end{aligned}$$

□

B.4.3 Proof of Corollary 6

Let us first prove the following result:

Lemma 12. *Let $x \in \mathbb{R}^n$ and let p, q two integers such that $p > q > 0$. Then:*

$$n^{\frac{1}{p}-\frac{1}{q}} \|x\|_q \leqslant \|x\|_p \leqslant \|x\|_q \ . \tag{B.6}$$

These two bounds are tight. Indeed,

$$\begin{aligned} &\forall i, j, x_i = x_j && \implies n^{\frac{1}{p}-\frac{1}{q}}\|x\|_q = \|x\|_p \\ &\exists i / x_i \neq 0 \text{ and } \forall i \neq j, x_j = 0 && \implies \|x\|_p = \|x\|_q \ . \end{aligned}$$

Proof. Let us first prove the lower bound of Eq. (B.6). By the Hölder inequality we have:

$$\sum_i |x_i|^q \leqslant \left(\sum_i (|x_i|^q)^P\right)^{\frac{1}{P}} \left(\sum_i (1)^Q\right)^{\frac{1}{Q}} \text{ where } P = \frac{p}{q} \text{ and } Q = \frac{p}{p-q} \ .$$

So, we have, $\sum_i |x_i|^q \leqslant \|x\|_p^q n^{1-\frac{q}{p}}$, i.e., $\|x\|_p n^{\frac{1}{p}-\frac{1}{q}} \leqslant \|x\|_p$.

Then let us prove the upper bound. We have $\sum_i \frac{|x_i|^q}{\|x\|_q^q} = 1$. Hence, for all $1 \leqslant i \leqslant n$, $\frac{|x_i|^q}{\|x\|_q^q} \leqslant 1$. Therefore, for all i, $\frac{|x_i|^p}{\|x\|_q^p} \leqslant \frac{|x_i|^q}{\|x\|_q^q}$, hence $\sum_i \frac{|x_i|^p}{\|x\|_q^p} \leqslant \sum_i \frac{|x_i|^q}{\|x\|_q^q} = 1$, which yields the announced inequality: $\|x\|_p^p \leqslant \|x\|_q^p$.

Let us prove the sufficient conditions when the inequalities become equalities:

$$\begin{aligned} \forall i, j, x_i = x_j \implies \|x\|_p = |x_i| n^{\frac{1}{p}} \text{ and } \|x\|_q = |x_i| n^{\frac{1}{q}} \implies \|x\|_p = \|x\|_q n^{\frac{1}{p}-\frac{1}{q}} \\ \exists i, x_i \neq 0 \text{ and } \forall i \neq j, x_j = 0 \implies \|x\|_p = |x_i| \text{ and } \|x\|_q = |x_i| \implies \|x\|_p = \|x\|_q. \end{aligned}$$

□

The Corollary 6 is the application of Lemma 12 with $p = 4$ and $q = 2$.

Proof. Indeed we have by Theorem 5.4.1 that the useful interval of variance is $0 \leqslant \sigma^2 \leqslant \|\alpha\|_4^4 \times \frac{2^{n-2}}{n} - \frac{n}{2}$, where $\|\alpha\|_2^2 = n$ (recall Assumption 1). Then by Lemma 12:

$$\begin{aligned} &\left(\|\alpha\|_2 n^{\frac{1}{4}-\frac{1}{2}+\frac{1}{2}}\right)^4 \times \frac{2^{n-2}}{n} - \frac{n}{2} \leqslant \|\alpha\|_4^4 \times \frac{2^{n-2}}{n} - \frac{n}{2} \leqslant \|\alpha\|_2^4 \times \frac{2^{n-2}}{n} - \frac{n}{2} \\ \implies &\left(\|\alpha\|_2 n^{\frac{1}{4}-\frac{1}{2}}\right)^4 \times \frac{2^{n-2}}{n} - \frac{n}{2} \leqslant \|\alpha\|_4^4 \times \frac{2^{n-2}}{n} - \frac{n}{2} \leqslant n^2 \times \frac{2^{n-2}}{n} - \frac{n}{2} \\ \implies &\underbrace{2^{n-2} - \frac{n}{2}} \leqslant \|\alpha\|_4^4 \times \frac{2^{n-2}}{n} - \frac{n}{2} \leqslant n \times 2^{n-2} - \frac{n}{2} \end{aligned}$$

Value of Theorem 5.2.1.

□

B. APPENDIX OF MULTIVARIATE ATTACK.

APPENDIX C

Appendix of Mixed Order.

C.1 Computation of the Moments

C.1.1 Computation of μ_1

There is no computational difficulty:

$$\mu_1 = \mathbb{E}(f^{(0)}) + \mathbb{E}(f^{(1)}) + \sum_{i \in I} \mathbb{E}(f^{(i)}) + \sum_{j \in J} \mathbb{E}(f^{(j)}). \tag{C.1}$$

Now, when there is no φ in the R.V., then the expectation is only on M (indeed, $\frac{1}{2^n!} \sum_{\varphi \in S_{2^n}} 1 = 1$). Thus,

$$\mathbb{E}(f^{(0)}) = \frac{1}{2^n} \sum_{m \in \mathbb{F}_2^n} (x^{(0)} - \mathrm{HW}[S[t \oplus k] \oplus m])^2 = \frac{1}{2^n} \sum_{m \in \mathbb{F}_2^n} (x^{(0)} - \mathrm{HW}[m])^2, \tag{C.2}$$

which cannot further be simplified (in the simulations, it will be computed by the computer).

Similarly

$$\mathbb{E}(f^{(1)}) = \frac{1}{2^n} \sum_{m \in \mathbb{F}_2^n} (x^{(1)} - \mathrm{HW}[S[t \oplus k] \oplus m])^2 = \frac{1}{2^n} \sum_{m \in \mathbb{F}_2^n} (x^{(1)} - \mathrm{HW}[m])^2. \tag{C.3}$$

When there is an expectation on Φ, then at order one, it considers **only one value** $\Phi(\omega)$. It is uniformly distributed, hence one can replace the expectation on Φ by an expectation on

one value of φ, we call M'. For instance:

$$\begin{aligned}\mathbb{E}(f^{(i)}) &= \frac{1}{2^n!}\sum_{\varphi\in S_{2^n}}(x^{(i)} - \mathrm{HW}[\varphi(\omega)])^2 \\ &= \frac{1}{2^n}\sum_{m'\in\mathbb{F}_2^n}(x^{(i)} - \mathrm{HW}[m'])^2, \end{aligned} \tag{C.4}$$

which can thus be computed with the same *average* method as $\mathbb{E}(f^{(0)})$.

Lastly, when there is both M and $\Phi(\omega)$, then whichever variable can absorb the other one, since both are uniformly distributed on $\mathbb{F}_2^n$. This means that:

$$\begin{aligned}\mathbb{E}(f^{(j)}) &= \frac{1}{2^n}\sum_{m\in\mathbb{F}_2^n}\frac{1}{2^n!}\sum_{\varphi\in S_{2^n}}(x^{(j)} - \mathrm{HW}[\varphi(\omega)\oplus m])^2 \\ &= \frac{1}{2^{2n}}\sum_{m,m'\in\mathbb{F}_2^n}(x^{(j)} - \mathrm{HW}[m\oplus m'])^2 \\ &= \frac{1}{2^{2n}}\sum_{\widetilde{m},m'\in\mathbb{F}_2^n}(x^{(j)} - \mathrm{HW}[\widetilde{m}\oplus m'\oplus m'])^2 \qquad \text{where } \widetilde{m} = m\oplus m' \qquad \text{(C.5)} \\ &= \frac{1}{2^n}\sum_{\widetilde{m}\in\mathbb{F}_2^n}(x^{(j)} - \mathrm{HW}[\widetilde{m}])^2, \qquad \text{(C.6)}\end{aligned}$$

which is once again a similar computation as done for computing $\mathbb{E}(f^{(0)})$.

C.1.2 Computation of μ_2

Recall that only the key dependent terms of μ_2 are needed for ROPT_2 and ROPT_3.

Notice that the square terms are computed as the non-square terms. For instance,

$$\mathbb{E}({f^{(0)}}^2) = \frac{1}{2^n}\sum_{m\in\mathbb{F}_2^n}(x^{(0)} - \mathrm{HW}[S[t\oplus k]\oplus m])^4 = \frac{1}{2^n}\sum_{m\in\mathbb{F}_2^n}(x^{(0)} - \mathrm{HW}[m])^4, \tag{C.7}$$

which we drop since it does not depend on k. All in one, the only key-dependent term is:

$$\mathbb{E}(f^{(0)}\times f^{(1)}) = \frac{1}{2^n}\sum_{m\in\mathbb{F}_2^n}(x^{(0)} - \mathrm{HW}[S[t\oplus k]\oplus m])^2(x^{(1)} - \mathrm{HW}[m])^2, \tag{C.8}$$

which cannot be further simplified and will be computed by the computer. So, for the purpose of the attack, we have:

$$\mu_2 = \mathbb{E}(f^{(0)}\times f^{(1)}) + \mathrm{cst}. \tag{C.9}$$

C.1.3 Computation of μ_3

We shall consider only terms which depend on the key, hence product of three terms, one of which (at least) is $f^{(0)}$. Obviously, $\mathbb{E}({f^{(0)}}^3)$ does not depend on k, for the same reason as given in Eqn. (C.7). But the two terms:

1. $\mathbb{E}({f^{(0)}}^2 f^{(1)})$ and
2. $\mathbb{E}(f^{(0)}{f^{(1)}}^2)$

Notice that they are present $\binom{3}{2} = 3$ times each when developing the cube.

Interestingly, those are **not** the only cases where $f^{(0)}$ and $f^{(1)}$ are selected.

$$
\begin{aligned}
&\mathbb{E}(f^{(0)}f^{(1)}f^{(j)}) \\
&= \frac{1}{2^n}\sum_{m\in\mathbb{F}_2^n}\frac{1}{2^n!}\sum_{\varphi\in S_{2^n}}(x^{(0)} - \mathrm{HW}[S[t\oplus k]\oplus m])^2(x^{(1)} - \mathrm{HW}[m])^2(x^{(j)} - \mathrm{HW}[\varphi(\omega)\oplus m])^2 \\
&= \frac{1}{2^n}\sum_{m\in\mathbb{F}_2^n}\frac{1}{2^n}\sum_{m'\in\mathbb{F}_2^n}(x^{(0)} - \mathrm{HW}[S[t\oplus k]\oplus m])^2(x^{(1)} - \mathrm{HW}[m])^2(x^{(j)} - HWm'\oplus m)^2 \\
&= \frac{1}{2^n}\sum_{m\in\mathbb{F}_2^n}(x^{(0)} - \mathrm{HW}[S[t\oplus k]\oplus m])^2(x^{(1)} - \mathrm{HW}[m])^2\frac{1}{2^n}\sum_{m'\in\mathbb{F}_2^n}(x^{(j)} - \mathrm{HW}[m'\oplus m])^2 \\
&= \frac{1}{2^n}\sum_{m\in\mathbb{F}_2^n}(x^{(0)} - \mathrm{HW}[S[t\oplus k]\oplus m])^2(x^{(1)} - \mathrm{HW}[m])^2\frac{1}{2^n}\sum_{\widetilde{m'}\in\mathbb{F}_2^n}(x^{(j)} - \mathrm{HW}[\widetilde{m'}])^2 \qquad \text{(As in Eq. (C.5))} \\
&= \mathbb{E}(f^{(0)}f^{(1)})\mathbb{E}(f^{(j)}).
\end{aligned}
$$

Similarly, we have:

$$\mathbb{E}(f^{(0)}f^{(1)}f^{(i)}) = \mathbb{E}(f^{(0)}f^{(1)})\mathbb{E}(f^{(i)}).$$

Now, we consider products without $f^{(1)}$. Obviously, taking only $f^{(0)}$ and $f^{(i)}$ is not enough, since: $\mathbb{E}({f^{(0)}}^2 f^{(i)}) = \mathbb{E}({f^{(0)}}^2)\mathbb{E}(f^{(i)})$ and $\mathbb{E}(f^{(0)}{f^{(i)}}^2) = \mathbb{E}(f^{(0)})\mathbb{E}({f^{(i)}}^2)$ are key independent. The same goes for $\mathbb{E}({f^{(0)}}^2 f^{(j)})$ and $\mathbb{E}(f^{(0)}{f^{(j)}}^2)$. We are left with $\mathbb{E}(f^{(0)}f^{(i)}f^{(i')})$, $\mathbb{E}(f^{(0)}f^{(j)}f^{(j')})$, and $\mathbb{E}(f^{(0)}f^{(i)}f^{(j)})$.

The term $\mathbb{E}(f^{(0)}f^{(i)}f^{(i')}) = \mathbb{E}(f^{(0)})\mathbb{E}(f^{(i)}f^{(i')}))$ does not depend on k, because there is no M in $f^{(i)}$.

The term $\mathbb{E}(f^{(0)}f^{(j)}f^{(j')})$ can also factorize as $\mathbb{E}(f^{(0)})\mathbb{E}(f^{(j)}f^{(j')}))$, hence it does not depend on k. The reason is more subtle, so we detail it:

$$
\begin{aligned}
\mathbb{E}(f^{(0)}f^{(j)}f^{(j')}) = \frac{1}{2^n}\sum_{m\in\mathbb{F}_2^n}&(x^{(0)} - \mathrm{HW}[S[t\oplus k]\oplus m])^2 \\
\times\frac{1}{2^n(2^n-1)}\sum_{\substack{(m',m'')\in\mathbb{F}_2^n\times\mathbb{F}_2^n \\ \text{s.t. } m'\neq m''}}&(x^{(j)} - \mathrm{HW}[m'\oplus m])^2(x^{(j')} - \mathrm{HW}[m''\oplus m])^2.
\end{aligned}
$$

Now, the second sum does not depend on m, as shown below:

$$\frac{1}{2^n(2^n-1)} \sum_{\substack{(m',m'')\in\mathbb{F}_2^n\times\mathbb{F}_2^n \\ \text{s.t. } m'\neq m''}} (x^{(j)} - \mathrm{HW}[m'\oplus m])^2 (x^{(j')} - \mathrm{HW}[m''\oplus m])^2 =$$

$$\frac{1}{2^n} \sum_{m'\in\mathbb{F}_2^n} (x^{(j)} - \mathrm{HW}[m'\oplus m])^2 \frac{1}{2^n-1} \sum_{m''\in\mathbb{F}_2^n\setminus\{m'\}} (x^{(j')} - \mathrm{HW}[m''\oplus m])^2 =$$

$$\frac{1}{2^n} \sum_{\widetilde{m'}\in\mathbb{F}_2^n} (x^{(i)} - \mathrm{HW}[\widetilde{m'}])^2 \frac{1}{2^n-1} \sum_{m''\in\mathbb{F}_2^n\setminus\{\widetilde{m'}\oplus m\}} (x^{(j')} - \mathrm{HW}[m''\oplus m])^2 =$$

$$\frac{1}{2^n} \sum_{\widetilde{m'}\in\mathbb{F}_2^n} (x^{(j)} - \mathrm{HW}[\widetilde{m'}])^2 \frac{1}{2^n-1} \sum_{\widetilde{m''}\in\mathbb{F}_2^n\setminus\{\widetilde{m''}\oplus\cancel{m\oplus m}\}} (x^{(j')} - \mathrm{HW}[\widetilde{m''}])^2.$$

Consequently, the last case is $\mathbb{E}(f^{(0)}f^{(i)}f^{(j)})$. We can subdivide it into two cases: $j = i + 2^n$ and $j \neq i + 2^n$. When $j = i + 2^n$, the permutation Φ is evaluated at the same ω in $f^{(i)}$ and $f^{(j)}$. We denote by M' the R.V. $\Phi(\omega)$, where $\omega = j - 2$. Hence:

$$\mathbb{E}(f^{(0)}f^{(i)}f^{(j=i+2^n)}) =$$
$$\frac{1}{2^n} \sum_{m\in\mathbb{F}_2^n} (x^{(0)} - \mathrm{HW}[S[t\oplus k]\oplus m])^2 \frac{1}{2^n} \sum_{m'\in\mathbb{F}_2^n} (x^{(i)} - \mathrm{HW}[m'])^2 (x^{(j)} - \mathrm{HW}[m'\oplus m])^2. \quad \text{(C.10)}$$

These terms (for all $j \in J$) correspond to the MVA_{TR} attack published at CHES 2015 (27).

Eventually, there are the terms for $j \neq i - 2^n$. They are actually key dependent, hence must be kept. They are equal to:

$$\mathbb{E}(f^{(0)}f^{(i)}f^{(j\neq i+2^n)}) = \frac{1}{2^n} \sum_{m\in\mathbb{F}_2^n} (x^{(0)} - \mathrm{HW}[S[t\oplus k]\oplus m])^2$$
$$\times \frac{1}{2^n} \frac{1}{2^n-1} \sum_{\substack{(m',m'')\in\mathbb{F}_2^n\times\mathbb{F}_2^n \\ \text{s.t. } m'\neq m''}} (x^{(i)} - \mathrm{HW}[m'])^2 (x^{(j)} - \mathrm{HW}[m''\oplus m])^2.$$

Interestingly, without the constraint $m' \neq m''$, this quantity does not depend on the key. So, the leakage which is exploited here is due to the fact Φ is not a random function, but a bijection. As, in μ_3, we are only interested in non constant terms, we can rewrite:

$$\mathbb{E}(f^{(0)}f^{(i)}f^{(j\neq i+2^n)}) = \mathrm{cst} - \frac{1}{2^n} \sum_{m\in\mathbb{F}_2^n} (x^{(0)} - \mathrm{HW}[S[t\oplus k]\oplus m])^2$$
$$\times \frac{1}{2^n} \frac{1}{2^n-1} \sum_{\substack{(m',m'')\in\mathbb{F}_2^n\times\mathbb{F}_2^n \\ \text{s.t. } m'=m''}} (x^{(i)} - \mathrm{HW}[m'])^2 (x^{(j)} - \mathrm{HW}[m''\oplus m])^2$$
$$= \mathrm{cst} - \frac{1}{2^n} \sum_{m\in\mathbb{F}_2^n} (x^{(0)} - \mathrm{HW}[S[t\oplus k]\oplus m])^2$$
$$\times \frac{1}{2^n-1} \sum_{m'\in\mathbb{F}_2} (x^{(i)} - \mathrm{HW}[m'])^2 (x^{(j)} - \mathrm{HW}[m'\oplus m])^2. \quad \text{(C.11)}$$

The non-constant term is similar to Eqn. (C.10) provided a scaling by $-(2^n-1)/2^n$ is done.

So, for the purpose of the attack, we have:

$$\mu_3 = \text{cst} + 3\mathbb{E}({f^{(0)}}^2 f^{(1)}) + 3\mathbb{E}(f^{(0)}{f^{(1)}}^2) + 3!\mathbb{E}(f^{(0)} \times f^{(1)})\left(\sum_{i\in I}\mathbb{E}(f^{(i)}) + \sum_{j\in J}\mathbb{E}(f^{(j)})\right) + 3!\sum_{i=2}^{2^n+1}\mathbb{E}(f^{(0)}f^{(i)}f^{(j=i+2^n)}) + 3!\sum_{i=2}^{2^n+1}\sum_{j\in\{2+2^n,\ldots,2^{n+1}+1\}\setminus\{i+2^n\}}\mathbb{E}(f^{(0)}f^{(0)}f^{(j)}). \quad \text{(C.12)}$$

C.2 Complexity Proofs

C.2.1 Proof of Lemma 7

In order to prove Lemma 7 let us first introduce a preliminary result.

Lemma 13. *The quantity $\binom{\Pi}{\ell}$ is increasing if $\ell < \lceil \Pi/2 \rceil$ and its maximum is $\binom{\Pi}{\lceil \frac{\Pi}{2} \rceil}$.*

Proof.

$$\binom{\Pi}{\ell+1} = \frac{\Pi!}{(\Pi-\ell-1)!(\ell+1)!} = \frac{\Pi-\ell-1}{\ell+1}\binom{\Pi}{\ell},$$

and the factor $\frac{\Pi-\ell-1}{\ell+1}$ is strictly greater than 1. Indeed,

$$\frac{\Pi-\ell-1}{\ell+1} > 1 \iff \Pi > 2(\ell+1) \iff \ell < \lceil \Pi/2 \rceil .$$

□

Finally we can prove Lemma 7.

Proof. Let us first assume that one dimension leaks at most one element of the permutation. We can thus develop the expression of μ_ℓ, and we denote the complexity under the braces.

$$\begin{aligned}\mu_\ell &= \mathbb{E}_R\left(\|x - y(t,k,R)\|^{2\ell}\right) \\ &= \underbrace{\sum_{k_1+\ldots+k_D=\ell}}_{\binom{D+\ell-1}{\ell}} \frac{\ell!}{\prod_{d=1}^{D} k_d!} \underbrace{\mathbb{E}_R}_{2^{(\Omega-1)n}\left(\lceil \frac{\Pi}{2} \rceil\right)} \underbrace{\left(\prod_{d=1}^{D} {f^{(d)}}^{k_d}\right)}_{\min(D,\ell)}\end{aligned}$$

As $k_1 + \ldots + k_D = \ell$ there are at most D indices $k_d, 1 \le d \le D$ such that $k_d \neq 0$. Hence there are at most $\min(D,\ell)$ elements in the product.

Each dimensions which leaks an element of the permutation can also leaks the masks. The worst case in terms of complexity is when all the permutation leakages depend also on the masks. Let us denote by i such that $1 \le i \le \min(D,\ell)$ the number of those terms. Then

the expectation is computed over $2^{(\Omega-1)n}\frac{\Pi!}{(\Pi-i)!}$. Nevertheless by taking into account the commutativity properties of the product one can only compute $2^{(\Omega-1)n}\binom{\Pi}{i}$.

By Lemma 13 we have that is value $\binom{\Pi}{i}$ is maximum with $\binom{\Pi}{\ell}$ when $\ell \leq \lceil \frac{\Pi}{2} \rceil$. When $\ell > \frac{\Pi}{2}+1$ the maximum is $\binom{\Pi}{\lceil \frac{\Pi}{2} \rceil}$.

Finally as there are $\binom{D+\ell-1}{\ell}$ elements in the sum.

The complexity of μ_ℓ is lower than $\mathcal{O}\left(\binom{D+\ell-1}{\ell} 2^{(\Omega-1)n}\binom{\Pi}{\min(\lceil \frac{\Pi}{2} \rceil, \ell)}\right)$. □

C.2.2 Proof of Proposition 34

In order to prove Lemma 34 let us first introduce a preliminary result.

Lemma 14. *The quantity* $\binom{D-1+\ell}{\ell}$ *is increasing with* ℓ *if* $D > 1$.

Proof. We have that :

$$\binom{D-1+\ell+1}{\ell+1} = \frac{D+\ell}{\ell+1}\binom{D-1+\ell}{\ell},$$

where $\forall \ell$, $\frac{D+\ell}{\ell+1} > 1$ provided $D > 1$. □

Finally let us prove Prop. 34.

Proof. Complexity of OPT:

Following Eq. (6.2) we have that the computation for a key guess of OPT is:

$$\underbrace{\sum_{q=1}^{Q}}_{Q} \log \underbrace{\mathbb{E}}_{\Pi! 2^{n(\Omega-1)}} \exp \underbrace{\frac{-\|x - y(t,k,R)\|^2}{2\sigma^2}}_{D}. \tag{C.13}$$

We assume that the computation of the log and the exp is constant. As a consequence the complexity of the optimal distinguisher is $\mathcal{O}\left(Q \cdot (2^n)^{\Omega-1} \cdot \Pi! \cdot D\right)$

Complexity of ROPT_L:The computation of ROPT_L involves the computation of the μ_ℓ with $\ell \leq L$ (Eq. (54) and Eq. (53)). By Lemma 7 and Lemma 14 all these terms have a complexity lower than $\mathcal{O}\left(\binom{D+L-1}{L} \cdot 2^{(\Omega-1)n} \cdot \binom{\Pi}{\min(\lceil \frac{\Pi}{2} \rceil, L)}\right)$ (Eq. (6.16)).

As a consequence the complexity of ROPT_L is lower than

$$\mathcal{O}\left(Q \cdot L\binom{D+L-1}{L} \cdot 2^{(\Omega-1)n} \cdot \binom{\Pi}{\min\left(\lceil \frac{\Pi}{2} \rceil, L\right)}\right). \tag{C.14}$$

□

C.2.3 Proof of Proposition 35

Proof. Let us develop all the product in the term μ_ℓ in order to compute the expectation in the minimum number of values.

$$\mu_\ell = \mathbb{E}_M\Big(\big(\sum_{d=1}^{D}(x^{(d)} - y^{(0)})^2\big)^\ell\Big)$$
$$= \sum_{\substack{\ell_1,\ell_2,\ldots,\ell_D \\ \sum_{d=1}^{D}=\ell}} \frac{\ell!}{\prod_{d=1}^{D}\ell_d!}\mathbb{E}_M\big((x^{(1)} - y^{(1)})^{2\ell_1}\cdots(x^{(D)} - y^{(D)})^{2\ell_D}\big).$$

Moreover $(x^{(d)} - y^{(d)}(t,k,M))^{2\ell_d} = \sum_{i=0}^{2\ell_d}\binom{2\ell_d}{i}{x^{(d)}}^{2\ell_d - i}{y^{(\delta)}(t,k,M)}^i$

$$\mu_\ell = \sum_{\substack{\ell_1,\ell_2,\ldots,\ell_D \\ \sum_{d=1}^{D}\ell_d=\ell}} \frac{\ell!}{\prod_{d=1}^{D}\ell_d!}\mathbb{E}_M\left(\prod_{d=1}^{D}\left(\sum_{i=0}^{2\ell_d}\binom{2\ell_d}{i}{x^{(d)}}^{2\ell_1 - i}{y^{(d)}(t,k,M)}^i\right)\right)$$
$$= \sum_{\substack{\ell_1,\ell_2,\ldots,\ell_D \\ \sum_{d=1}^{D}\ell_d=\ell}} \frac{\ell!}{\prod_{d=1}^{D}\ell_d!} \sum_{\substack{i_1\le 2\ell_1 \\ \vdots \\ i_D\le 2\ell_D}} \prod_{d=1}^{D}\left(\binom{2\ell_d}{i_d}{x^{(d)}}^{2\ell_d - i_d}\right) \underbrace{\mathbb{E}_M\left(\prod_{d=1}^{D}{y^{(d)}(t,k,M)}^{i_d}\right)}_{\text{can be precomputed}}.$$

□

C.2.4 Proof of Proposition 36

Proof. In our case study the size of the permutation is $\Pi = 2^n$.

Then the complexity of OPT is given by a straightforward application of Eq. (6.17).

From Eq. (6.14) we have that for ROPT_2 the computation for one key guess and one trace is given by $\mathbb{E}(f^{(0)} \times f^{(1)}$. In this equation the expectation is computed over 2^n values (Eq. (C.7)).

From Eq. (6.15) we have that for ROPT_3 the computation for one key guess and one trace is given by $\mu_2^{(q)}(1 + \gamma\mu_1^{(q)}) - \gamma\frac{\mu_3^{(q)}}{3}$. It can be seen in Eq. (C.2), Eq. (C.3), Eq. (C.4) and Eq. (C.6) that the expectation of μ_1 is computed over 2^n values. The dominant term in μ_3 (Eq. (C.12)) is :

$$\underbrace{\sum_{i=2}^{2^n+1}\sum_{j\in\{2+2^n,\ldots,2^{n+1}+1\}\setminus\{i+2^n\}}}_{2^{2n}} \underbrace{\mathbb{E}}_{2^{2n}}(f^{(0)}f^{(i)}f^{(j)}).$$

The expectation in this term is computed over 2^{2n} values (Eq. (C.11)). The sum is computed on less than 2^{2n}. □

C.2.5 Time and complexity

The times of the section are expressed in seconds. All the attacks have been run on Intel Xeon X5660 running at 2.67 GHz. All the implementations are mono-thread. The model of the simulations is the one describe in Sec. 6.6. For each distinguisher the attacks are computed 1000 times on 1000 traces.

Attack	Dimension	Time (in seconds)	Computational Complexity
2O-CPA	2	39	$\mathcal{O}(Q)$
ROPT_2	2	295	$\mathcal{O}(Q)$
$\text{OPT}_{2\text{O}}$	2	9473	$\mathcal{O}(Q \cdot (2^n))$
MVA_{TR}	$2^{n+1}+1$	130	$\mathcal{O}(Q \cdot 2^n)$
ROPT_3	$2^{n+1}+2$	2495	$\mathcal{O}\left(Q \cdot 2^{2n}\right)$
OPT	$2^{n+1}+2$	Not computable	$\mathcal{O}\left(Q \cdot (2^n) \cdot 2^n! \cdot \left(2^{n+1}+2\right)\right)$

Table C.1: Time and complexity

C.3 Analysis of the DPAcontest.

Recently an open implementation of a masking scheme with shuffling has been presented in the DPA contest v4.2 (169). In this implementation the execution of the different states is performed in an random order.

An attacker can target the integrated leakages of the different states in order to counter the shuffling (35, 142).

A better approach is to take into account the possible leakages of the permutation. In this case the optimal distinguisher will be not computable as it involves an expectation over 16! values. In this case the rounded optimal attack will reduced this complexity.

Let us define the leakages of such implementations.

- $X^{(0)} = y^{(0)}(t,k,R) + N^{(0)}$ with $y^{(0)}(t,k,R) = \text{HW}[M]$,
- $X^{(1)} = y^{(1)}(t,k,R) + N^{(1)}$ with $y^{(1)}(t,k,R) = \text{HW}[S[\pi(T \oplus k)] \oplus M]$,
- $X^{(i)} = y^{(i)}(t,k,R) + N^{(i)}$, for $i = 2,\ldots,18$ with $y^{(i)}(t,k,R) = \text{HW}[\Phi(i-2)]$,

Then similarly to the Appendix C.1 we have that:

$$\mu_1 = \mathbb{E}(f^{(0)}) + \mathbb{E}(f^{(1)}) + \sum_{i \in I} \mathbb{E}(f^{(i)}), \tag{C.15}$$

$$\mu_2 = \mathbb{E}(f^{(0)} \times f^{(1)}) + \text{cst}. \tag{C.16}$$

Additionally as it is a low entropy masking scheme the secret key can leaked in an univariate high order attack. Depending on the number of masks involve in the masking scheme it could be at order 2, 3 or more. For simplicity let us assume it is at order 3. In such cases

$$\mu_3 = \mathbb{E}({f^{(1)}}^3) + 3\mathbb{E}({f^{(0)}}^2 f^{(1)}) + 3\mathbb{E}(f^{(0)}{f^{(1)}}^2) + 3! \sum_{i=2}^{2^n+1} \mathbb{E}(f^{(0)} f^{(1)} f^{(i)}) + \text{cst.} \tag{C.17}$$

Of course an attacker can additionally exploit all the leakages of the different states in order to increase the success of the attacks.

In some particular low entropy masking schemes the same masks are reused several time or are linked by deterministic relations (e.g the first version of the DPAcontest). In this context it could be interesting to combine the leakages of different states (20). In this case our method could benefit of the multiple possible points combinations.

Bibliography

[1] Mehdi-Laurent Akkar and Christophe Giraud. An Implementation of DES and AES Secure against Some Attacks. In LNCS, editor, *Proceedings of CHES'01*, volume 2162 of *LNCS*, pages 309–318. Springer, May 2001. Paris, France. 21, 78

[2] Cédric Archambeau, Éric Peeters, François-Xavier Standaert, and Jean-Jacques Quisquater. Template Attacks in Principal Subspaces. In *CHES*, volume 4249 of *LNCS*, pages 1–14. Springer, October 10-13 2006. Yokohama, Japan. 17, 37, 39, 47, 56

[3] Lejla Batina, Benedikt Gierlichs, and Kerstin Lemke-Rust. Differential Cluster Analysis. In Christophe Clavier and Kris Gaj, editors, *Cryptographic Hardware and Embedded Systems – CHES 2009*, volume 5747 of *Lecture Notes in Computer Science*, pages 112–127, Lausanne, Switzerland, 2009. Springer-Verlag. 15

[4] Lejla Batina, Benedikt Gierlichs, Emmanuel Prouff, Matthieu Rivain, François-Xavier Standaert, and Nicolas Veyrat-Charvillon. Mutual Information Analysis: a Comprehensive Study. *J. Cryptology*, 24(2):269–291, 2011. 120

[5] Lejla Batina, Jip Hogenboom, and Jasper G. J. van Woudenberg. Getting more from PCA: first results of using principal component analysis for extensive power analysis. In Dunkelman (48), pages 383–397. 17, 39, 56, 61, 62

[6] Lejla Batina and Matthew Robshaw, editors. *Cryptographic Hardware and Embedded Systems - CHES 2014 - 16th International Workshop, Busan, South Korea, September*

23-26, 2014. Proceedings, volume 8731 of *Lecture Notes in Computer Science*. Springer, 2014. 180, 183, 186

[7] Aurélie Bauer, Éliane Jaulmes, Emmanuel Prouff, Jean-René Reinhard, and Justine Wild. Horizontal collision correlation attack on elliptic curves - extended version -. *Cryptography and Communications*, 7(1):91–119, 2015. 29

[8] Pierre Belgarric, Shivam Bhasin, Nicolas Bruneau, Jean-Luc Danger, Nicolas Debande, Sylvain Guilley, Annelie Heuser, Zakaria Najm, and Olivier Rioul. Time-Frequency Analysis for Second-Order Attacks. In Francillon and Rohatgi (56), pages 108–122. 62

[9] Olivier Benoît and Thomas Peyrin. Side-Channel Analysis of Six SHA-3 Candidates. In *CHES*, volume 6225 of *Lecture Notes in Computer Science*, pages 140–157. Springer, August 17-20 2010. Santa Barbara, CA, USA. 81

[10] Alexandre Berzati, Cécile Canovas-Dumas, and Louis Goubin. Public Key Perturbation of Randomized RSA Implementations. In *CHES*, volume 6225 of *Lecture Notes in Computer Science*, pages 306–319. Springer, August 17-20 2010. Santa Barbara, CA, USA. 7

[11] Régis Bevan and Erik Knudsen. Ways to Enhance Differential Power Analysis. In *ICISC*, volume 2587 of *Lecture Notes in Computer Science*, pages 327–342. Springer, November 28-29 2002. Seoul, Korea. 14

[12] Shivam Bhasin, Jean-Luc Danger, Sylvain Guilley, and Zakaria Najm. NICV: Normalized Inter-Class Variance for Detection of Side-Channel Leakage. In *International Symposium on Electromagnetic Compatibility (EMC '14 / Tokyo)*. IEEE, May 12-16 2014. Session OS09: EM Information Leakage. Hitotsubashi Hall (National Center of Sciences), Chiyoda, Tokyo, Japan. 60

[13] Shivam Bhasin, Jean-Luc Danger, Sylvain Guilley, and Zakaria Najm. Side-channel Leakage and Trace Compression Using Normalized Inter-class Variance. In *Proceedings of the Third Workshop on Hardware and Architectural Support for Security and Privacy*, HASP '14, pages 7:1–7:9, New York, NY, USA, 2014. ACM. 38, 60

[14] Eli Biham, Ross J. Anderson, and Lars R. Knudsen. Serpent: A new block cipher proposal. pages 222–238. 3

[15] Eli Biham and Adi Shamir. Differential Fault Analysis of Secret Key Cryptosystems. In *CRYPTO*, volume 1294 of *LNCS*, pages 513–525. Springer, August 1997. Santa Barbara, California, USA. DOI: 10.1007/BFb0052259. 7

[16] Johannes Blömer, Jorge Guajardo, and Volker Krummel. Provably Secure Masking of AES. In Helena Handschuh and M. Anwar Hasan, editors, *Selected Areas in Cryptography*, volume 3357 of *Lecture Notes in Computer Science*, pages 69–83. Springer, 2004. 21, 78

[17] Dan Boneh, Richard A. DeMillo, and Richard J. Lipton. On the Importance of Checking Cryptographic Protocols for Faults. In *Proceedings of Eurocrypt'97*, volume 1233 of *LNCS*, pages 37–51. Springer, May 11-15 1997. Konstanz, Germany. DOI: 10.1007/3-540-69053-04. 7

[18] Éric Brier, Christophe Clavier, and Francis Olivier. Correlation power analysis with a leakage model. In Marc Joye and Jean-Jacques Quisquater, editors, *Cryptographic Hardware and Embedded Systems - CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings*, volume 3156 of *Lecture Notes in Computer Science*, pages 16–29. Springer, 2004. 14

[19] Éric Brier, Christophe Clavier, and Francis Olivier. Correlation Power Analysis with a Leakage Model. In *CHES*, volume 3156 of *LNCS*, pages 16–29. Springer, August 11–13 2004. Cambridge, MA, USA. 57

[20] Nicolas Bruneau, Jean-Luc Danger, Sylvain Guilley, Annelie Heuser, and Yannick Teglia. Boosting Higher-Order Correlation Attacks by Dimensionality Reduction. In Rajat Subhra Chakraborty, Vashek Matyas, and Patrick Schaumont, editors, *Security, Privacy, and Applied Cryptography Engineering - 4th International Conference, SPACE 2014, Pune, India, October 18-22, 2014. Proceedings*, volume 8804 of *Lecture Notes in Computer Science*, pages 183–200. Springer, 2014. 39, 55, 173

[21] Nicolas Bruneau, Sylvain Guilley, Annelie Heuser, Damien Marion, and Olivier Rioul. Less is More – Dimensionality Reduction from a Theoretical Perspective. In Handschuh and Güneysu (72). 122

[22] Nicolas Bruneau, Sylvain Guilley, Annelie Heuser, Damien Marion, and Olivier Rioul. Less is more - dimensionality reduction from a theoretical perspective. In Güneysu and Handschuh (65), pages 22–41. 37, 83, 84

[23] Nicolas Bruneau, Sylvain Guilley, Annelie Heuser, and Olivier Rioul. Masks Will Fall Off – Higher-Order Optimal Distinguishers. In Palash Sarkar and Tetsu Iwata, editors, *Advances in Cryptology – ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014, Proceedings, Part II*, volume 8874 of *Lecture Notes in Computer Science*, pages 344–365. Springer, 2014. 24, 77, 86, 120, 124, 130, 134, 135

[24] Nicolas Bruneau, Sylvain Guilley, Annelie Heuser, and Olivier Rioul. Masks Will Fall Off: Higher-Order Optimal Distinguishers. In *ASIACRYPT*, volume 8874 of *LNCS*, pages 344–365. Springer, December 2014. P. Sarkar and T. Iwata (Eds.): ASIACRYPT 2014, PART II. 26

[25] Nicolas Bruneau, Sylvain Guilley, Annelie Heuser, Olivier Rioul, François-Xavier Standaert, and Yannick Teglia. Taylor expansion of maximum likelihood attacks for masked and shuffled implementations. In *Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I*, pages 573–601, 2016. 119

[26] Nicolas Bruneau, Sylvain Guilley, Zakaria Najm, and Yannick Teglia. Multi-variate high-order attacks of shuffled tables recomputation. In Güneysu and Handschuh (65), pages 475–494. 89

[27] Nicolas Bruneau, Sylvain Guilley, Zakaria Najm, and Yannick Teglia. Multi-variate High-Order Attacks of Shuffled Tables Recomputation. In Handschuh and Güneysu (72). 121, 123, 127, 131, 168

[28] Carolynn Burwick, Don Coppersmith, Edward D'Avignon, Rosario Gennaro, Shai Halevi, Charanjit Jutla, Stephen M. Matyas, Luke O'Connor, Mohammad Peyravian, David Safford, and Nevenko Zunic. The mars encryption algorithm, 1999. 3

[29] Suresh Chari, Charanjit S. Jutla, Josyula R. Rao, and Pankaj Rohatgi. Towards Sound Approaches to Counteract Power-Analysis Attacks. In *CRYPTO*, volume 1666 of *LNCS*. Springer, August 15-19 1999. Santa Barbara, CA, USA. ISBN: 3-540-66347-9. 19, 21, 23, 78, 91

[30] Suresh Chari, Charanjit S. Jutla, Josyula R. Rao, and Pankaj Rohatgi. Towards Sound Approaches to Counteract Power-Analysis Attacks. In Wiener (182), pages 398–412. 19, 25, 120, 127, 130

[31] Suresh Chari, Josyula R. Rao, and Pankaj Rohatgi. Template Attacks. In *CHES*, volume 2523 of *LNCS*, pages 13–28. Springer, August 2002. San Francisco Bay (Redwood City), USA. 13, 15, 16, 38, 40, 56

[32] Omar Choudary and Markus G. Kuhn. Efficient template attacks. In Francillon and Rohatgi (56), pages 253–270. 18, 38

[33] Mathieu Ciet and Marc Joye. (Virtually) Free randomization techniques for elliptic curve cryptography. In *Information and Communications Security (ICICS 2003)*, volume 2836 of *Lecture Notes in Computer Science*. Springer-Verlag, 10 2003. 22

[34] C. Clavier and M. Joye. Universal exponentiation algorithm - a first step towards provable spa-resistance. In *In Proceedings of CHES*, pages 300–308. Springer-Verlag, 2001. 22

[35] Christophe Clavier, Jean-Sébastien Coron, and Nora Dabbous. Differential Power Analysis in the Presence of Hardware Countermeasures. In Çetin Kaya Koç and Christof Paar, editors, *CHES*, volume 1965 of *Lecture Notes in Computer Science*, pages 252–263. Springer, 2000. 172

[36] Christophe Clavier, Jean-Luc Danger, Guillaume Duc, M. Abdelaziz Elaabid, Benoît Gérard, Sylvain Guilley, Annelie Heuser, Michael Kasper, Yang Li, Victor Lomné, Daisuke Nakatsu, Kazuo Ohta, Kazuo Sakiyama, Laurent Sauvage, Werner Schindler, Marc Stöttinger, Nicolas Veyrat-Charvillon, Matthieu Walle, and Antoine Wurcker. Practical improvements of side-channel attacks on AES: feedback from the 2nd DPA contest. *Journal of Cryptographic Engineering*, pages 1–16, 2014. 50

[37] Christophe Clavier, Benoit Feix, Georges Gagnerot, Christophe Giraud, Mylène Roussellet, and Vincent Verneuil. ROSETTA for single trace analysis. In *Progress in Cryptology - INDOCRYPT 2012, 13th International Conference on Cryptology in India, Kolkata, India, December 9-12, 2012. Proceedings*, pages 140–155, 2012. 29

[38] Jean-Sébastien Coron. Resistance against Differential Power Analysis for Elliptic Curve Cryptosystems. In Çetin Kaya Koç and Christof Paar, editors, *CHES*, volume 1717 of *LNCS*, pages 292–302. Springer, 1999. 12, 19, 21, 22, 23

[39] Jean-Sébastien Coron. Higher Order Masking of Look-Up Tables. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT*, volume 8441 of *Lecture Notes in Computer Science*, pages 441–458. Springer, 2014. 21, 78, 85, 93, 102, 103

[40] Jean-Sébastien Coron, Emmanuel Prouff, and Matthieu Rivain. Side Channel Cryptanalysis of a Higher Order Masking Scheme. In Paillier and Verbauwhede (123), pages 28–44. 102

[41] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, July 18 2006. ISBN-10: ISBN-10: 0471241954, ISBN-13: 978-0471241959, 2nd edition. 40

[42] Jean-Luc Danger, Nicolas Debande, Sylvain Guilley, and Youssef Souissi. High-order timing attacks. In *Proceedings of the First Workshop on Cryptography and Security in Computing Systems*, CS2 '14, pages 7–12, New York, NY, USA, 2014. ACM. 38

[43] U. Datta and A.S. Muktibodh. *Algebra And Trigonometry*. Prentice-Hall Of India Pvt. Limited, 2006. 116

[44] Nicolas Debande, Youssef Souissi, M. Abdelaziz Elaabid, Sylvain Guilley, and Jean-Luc Danger. Wavelet transform based pre-processing for side channel analysis. In *45th Annual IEEE/ACM International Symposium on Microarchitecture, MICRO 2012, Workshops Proceedings, Vancouver, BC, Canada, December 1-5, 2012*, pages 32–38. IEEE Computer Society, 2012. 38

[45] Whitfield Diffie and Martin Edward Hellman. New Directions in Cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, November 1976. 5, 8

[46] A. Adam Ding, Liwei Zhang, Yunsi Fei, and Pei Luo. A statistical model for higher order DPA on masked devices. In Batina and Robshaw (6), pages 147–169. 31, 92, 95, 96, 105, 120

[47] Alexandre Duc, Sebastian Faust, and François-Xavier Standaert. Making masking security proofs concrete - or how to evaluate the security of any leaking device. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I*, volume 9056 of *Lecture Notes in Computer Science*, pages 401–429. Springer, 2015. 123, 125

[48] Orr Dunkelman, editor. *Topics in Cryptology - CT-RSA 2012 - The Cryptographers' Track at the RSA Conference 2012, San Francisco, CA, USA, February 27 - March 2, 2012. Proceedings*, volume 7178 of *Lecture Notes in Computer Science*. Springer, 2012. 175, 192

[49] François Durvaux, François-Xavier Standaert, Nicolas Veyrat-Charvillon, Jean-Baptiste Mairy, and Yves Deville. Efficient Selection of Time Samples for Higher-Order DPA with Projection Pursuits. Cryptology ePrint Archive, Report 2014/412, 2014. `http://eprint.iacr.org/2014/412`. To appear at COSADE 2015 (LNCS), April 13-14 2015, Berlin, Germany. 39

[50] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. pages 10–18. 6

[51] Guangjun Fan, Yongbin Zhou, Hailong Zhang, and Dengguo Feng. How to Choose Interesting Points for Template Attacks? *IACR Cryptology ePrint Archive*, 2014:332, 2014. 17

[52] Yunsi Fei, A. Adam Ding, Jian Lao, and Liwei Zhang. A statistics-based success rate model for DPA and CPA. *J. Cryptographic Engineering*, 5(4):227–243, 2015. 96

[53] Yunsi Fei, Qiasi Luo, and A. Adam Ding. A Statistical Model for DPA with Novel Algorithmic Confusion Analysis. In Prouff and Schaumont (134), pages 233–250. 31, 43, 93, 99

[54] Pierre-Alain Fouque, Denis Réal, Frédéric Valette, and M'hamed Drissi. The carry leakage on the randomized exponent countermeasure. In *Cryptographic Hardware and Embedded Systems - CHES 2008, 10th International Workshop, Washington, D.C., USA, August 10-13, 2008. Proceedings*, pages 198–213, 2008. 22

[55] Pierre-Alain Fouque and Frederic Valette. The Doubling Attack, Why Upwards Is Better than Downwards. pages 269–280, 2003. ISBN: 978-3-540-40833-8. 23

[56] Aurélien Francillon and Pankaj Rohatgi, editors. *Smart Card Research and Advanced Applications - 12th International Conference, CARDIS 2013, Berlin, Germany, November 27-29, 2013. Revised Selected Papers*, volume 8419 of *LNCS*. Springer, 2014. 176, 179

[57] Guillaume Fumaroli, Ange Martinelli, Emmanuel Prouff, and Matthieu Rivain. Affine masking against higher-order side channel analysis. Cryptology ePrint Archive, Report

2010/523, 2010. http://eprint.iacr.org/2010/523. To be published at SAC'2010 (PDF). 20

[58] Karine Gandolfi, Christophe Mourtel, and Francis Olivier. Electromagnetic Analysis: Concrete Results. In *CHES*, volume 2162 of *LNCS*, pages 251–261. Springer, May 14-16 2001. Paris, France. 7

[59] Benedikt Gierlichs, Lejla Batina, Pim Tuyls, and Bart Preneel. Mutual information analysis. In *CHES, 10th International Workshop*, volume 5154 of *Lecture Notes in Computer Science*, pages 426–442. Springer, August 10-13 2008. Washington, D.C., USA. 15

[60] Benedikt Gierlichs, Kerstin Lemke-Rust, and Christof Paar. Templates vs. Stochastic Methods. In *CHES*, volume 4249 of *LNCS*, pages 15–29. Springer, October 10-13 2006. Yokohama, Japan. 16, 37, 38

[61] Jovan Dj. Golić and Christophe Tymen. Multiplicative Masking and Power Analysis of AES. In *CHES*, volume 2523 of *Lecture Notes in Computer Science*, pages 198–212. Springer, August 13-15 2002. San Francisco, USA. 20

[62] Sylvain Guilley, Sumanta Chaudhuri, Laurent Sauvage, Philippe Hoogvorst, Renaud Pacalet, and Guido Marco Bertoni. Security Evaluation of WDDL and SecLib Countermeasures against Power Attacks. *IEEE Transactions on Computers*, 57(11):1482–1497, nov 2008. 39

[63] Sylvain Guilley, Annelie Heuser, and Olivier Rioul. A Key to Success - Success Exponents for Side-Channel Distinguishers. In Alex Biryukov and Vipul Goyal, editors, *Progress in Cryptology - INDOCRYPT 2015 - 16th International Conference on Cryptology in India, Bangalore, India, December 6-9, 2015, Proceedings*, volume 9462 of *Lecture Notes in Computer Science*, pages 270–290. Springer, 2015. 31, 98, 99, 105, 159, 182

[64] Sylvain Guilley, Annelie Heuser, and Olivier Rioul. A Key to Success – Success Exponents for Side-Channel Distinguishers (extended version of (63)). Cryptology ePrint Archive, Report 2016/987, October 24 2016. http://eprint.iacr.org/2016/987. 98, 159

[65] Tim Güneysu and Helena Handschuh, editors. *Cryptographic Hardware and Embedded Systems - CHES 2015 - 17th International Workshop, Saint-Malo, France, September 13-16, 2015, Proceedings*, volume 9293 of *Lecture Notes in Computer Science*. Springer, 2015. 177, 178, 188

[66] Suvadeep Hajra and Debdeep Mukhopadhyay. Multivariate leakage model for improving non-profiling DPA on noisy power traces. In Dongdai Lin, Shouhuai Xu, and Moti Yung, editors, *Information Security and Cryptology - 9th International Conference, Inscrypt 2013, Guangzhou, China, November 27-30, 2013, Revised Selected Papers*, volume 8567 of *Lecture Notes in Computer Science*, pages 325–342. Springer, 2013. 39

[67] Suvadeep Hajra and Debdeep Mukhopadhyay. Pushing the limit of non-profiling dpa using multivariate leakage model. *IACR Cryptology ePrint Archive*, 2013:849, 2013. 61

[68] Suvadeep Hajra and Debdeep Mukhopadhyay. SNR to Success Rate: Reaching the Limit of Non-Profiling DPA. Cryptology ePrint Archive, Report 2013/865, 2013. `http://eprint.iacr.org/2013/865/`. 39

[69] Suvadeep Hajra and Debdeep Mukhopadhyay. On the optimal pre-processing for non-profiling differential power analysis. In Prouff (129), pages 161–178. 37, 39

[70] Suvadeep Hajra and Debdeep Mukhopadhyay. On the Optimal Pre-processing for Non-profiling Differential Power Analysis. In *COSADE*, Lecture Notes in Computer Science. Springer, April 14-15 2014. Paris, France. 56, 62, 72

[71] Suvadeep Hajra and Debdeep Mukhopadhyay. Reaching the limit of nonprofiling DPA. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 34(6):915–927, 2015. 39

[72] Helena Handschuh and Tim Güneysu, editors. *Cryptographic Hardware and Embedded Systems - CHES 2015 - 17th International Workshop, Saint-Malo, France, September 13-16, 2015. Proceedings*, volume 9293 of *Lecture Notes in Computer Science*. Springer, 2015. 177, 178

[73] Christoph Herbst, Elisabeth Oswald, and Stefan Mangard. An AES Smart Card Implementation Resistant to Power Analysis Attacks. In Jianying Zhou, Moti Yung, and Feng Bao, editors, *ACNS*, volume 3989 of *Lecture Notes in Computer Science*, pages 239–252, 2006. 120

[74] Annelie Heuser, Olivier Rioul, and Sylvain Guilley. Good Is Not Good Enough - Deriving Optimal Distinguishers from Communication Theory. In Batina and Robshaw (6), pages 55–74. 13, 40

[75] Annelie Heuser and Michael Zohner. Intelligent Machine Homicide - Breaking Cryptographic Devices Using Support Vector Machines. In Werner Schindler and Sorin A. Huss, editors, *COSADE*, volume 7275 of *LNCS*, pages 249–264. Springer, 2012. 13

[76] Johann Heyszl, Andreas Ibing, Stefan Mangard, Fabrizio De Santis, and Georg Sigl. Clustering Algorithms for Non-Profiled Single-Execution Attacks on Exponentiations. In *CARDIS*, Lecture Notes in Computer Science. Springer, November 2013. Berlin, Germany. 29

[77] Gabriel Hospodar, Benedikt Gierlichs, Elke De Mulder, Ingrid Verbauwhede, and Joos Vandewalle. Machine learning in side-channel analysis: a first study. *Journal of Cryptographic Engineering*, 1:293–302, 2011. 10.1007/s13389-011-0023-x. 13

[78] Yuval Ishai, Amit Sahai, and David Wagner. Private Circuits: Securing Hardware against Probing Attacks. In *CRYPTO*, volume 2729 of *Lecture Notes in Computer Science*, pages 463–481. Springer, August 17–21 2003. Santa Barbara, California, USA. 19, 120

[79] Ian T. Jolliffe. *Principal Component Analysis*. Springer Series in Statistics, 2002. ISBN: 0387954422. 17, 59

[80] Marc Joye and Christophe Tymen. Protections against Differential Analysis for Elliptic Curve Cryptography. In *CHES*, volume 2162 of *Lecture Notes in Computer Science*, pages 377–390. Springer, May 14-16 2001. Paris, France. 23

[81] Peter Karsmakers, Benedikt Gierlichs, Kristiaan Pelckmans, Katrien De Cock, Johan Suykens, Bart Preneel, and Bart De Moor. Side channel attacks on cryptographic devices as a classification problem. COSIC technical report, 2009. 39

[82] Neal Koblitz. Elliptic curve cryptosystems. *Mathematic of Computation*, 48:203–209, 1987. 6

[83] Paul Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. In *Advances in Cryptology - CRYPTO'99*, pages 388–397. Springer-Verlag, 1999. 7, 8

[84] Paul C. Kocher. Timing attacks on implementations of diffie-hellman, rsa, dss, and other systems. In Neal Koblitz, editor, *Advances in Cryptology - CRYPTO '96, 16th Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 1996, Proceedings*, volume 1109 of *Lecture Notes in Computer Science*, pages 104–113. Springer, 1996. 7, 8, 22, 23

[85] Paul C. Kocher. Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. In *Proceedings of CRYPTO'96*, volume 1109 of *LNCS*, pages 104–113. Springer-Verlag, 1996. 12

[86] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. In Wiener (182), pages 388–397. 41

[87] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential Power Analysis. In *Proceedings of CRYPTO'99*, volume 1666 of *LNCS*, pages 388–397. Springer-Verlag, 1999. 56, 79

[88] Oliver Kömmerling and Markus G. Kuhn. Design Principles for Tamper-Resistant Smartcard Processors. In *WOST '99 (USENIX Workshop on Smartcard Technology)*, pages 9–20, Berkeley, CA, USA, May 10-11 1999. USENIX Association. Chicago, Illinois, USA (On-line paper). ISBN: 1-880446-34-0. 7

[89] Boris Köpf and David Basin. An information-theoretic model for adaptive side-channel attacks. In *CCS'07: Proceedings of the 14th ACM conference on Computer and communications security*, pages 286–296, New York, NY, USA, 2007. ACM. 30

[90] Xuejia Lai and James L. Massey. A proposal for a new block encryption standard. In *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques on Advances in Cryptology*, EUROCRYPT '90, pages 389–404, New York, NY, USA, 1991. Springer-Verlag New York, Inc. 3

[91] Kerstin Lemke-Rust and Christof Paar. Analyzing side channel leakage of masked implementations with stochastic methods. In Joachim Biskup and Javier Lopez, editors, *Computer Security - ESORICS 2007, 12th European Symposium On Research In Computer Security, Dresden, Germany, September 24-26, 2007, Proceedings*, volume 4734 of *Lecture Notes in Computer Science*, pages 454–468. Springer, 2007. 120

[92] Kerstin Lemke-Rust and Christof Paar. Gaussian Mixture Models for Higher-Order Side Channel Analysis. In Paillier and Verbauwhede (123), pages 14–27. 120

[93] Liran Lerman, Romain Poussier, Gianluca Bontempi, Olivier Markowitch, and François-Xavier Standaert. Template attacks vs. machine learning revisited (and the curse of dimensionality in side-channel analysis). In Stefan Mangard and Axel Y. Poschmann, editors, *Constructive Side-Channel Analysis and Secure Design - 6th International Workshop,*

COSADE 2015, Berlin, Germany, April 13-14, 2015. Revised Selected Papers, volume 9064 of *Lecture Notes in Computer Science*, pages 20–33. Springer, 2015. 13

[94] Victor Lomné, Emmanuel Prouff, Matthieu Rivain, Thomas Roche, and Adrian Thillard. How to Estimate the Success Rate of Higher-Order Side-Channel Attacks. In Batina and Robshaw (6), pages 35–54. 31

[95] Victor Lomné, Emmanuel Prouff, and Thomas Roche. Behind the Scene of Side Channel Attacks. In Kazue Sako and Palash Sarkar, editors, *ASIACRYPT (1)*, volume 8269 of *Lecture Notes in Computer Science*, pages 506–525. Springer, 2013. 14

[96] Louis Goubin and Jacques Patarin. DES and Differential Power Analysis - The "Duplication" Method, 1999. 19

[97] Houssem Maghrebi, Emmanuel Prouff, Sylvain Guilley, and Jean-Luc Danger. A First-Order Leak-Free Masking Countermeasure. Cryptology ePrint Archive, Report 2012/028, 2012. `http://eprint.iacr.org/2012/028`. 78

[98] Stefan Mangard, Elisabeth Oswald, and Thomas Popp. *Power Analysis Attacks: Revealing the Secrets of Smart Cards*. Springer, December 2006. ISBN 0-387-30857-1, `http://www.dpabook.org/`. 6, 17

[99] Stefan Mangard, Elisabeth Oswald, and Thomas Popp. *Power analysis attacks - revealing the secrets of smart cards*. Springer, 2007. 120

[100] Daniel P. Martin, Luke Mather, Elisabeth Oswald, and Martijn Stam. *Characterisation and Estimation of the Key Rank Distribution in the Context of Side Channel Evaluations*, pages 548–572. Springer Berlin Heidelberg, Berlin, Heidelberg, 2016. 30

[101] Luke Mather, Elisabeth Oswald, and Carolyn Whitnall. Multi-target DPA attacks: Pushing DPA beyond the limits of a desktop computer. In Palash Sarkar and Tetsu Iwata, editors, *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, volume 8873 of *Lecture Notes in Computer Science*, pages 243–261. Springer, 2014. 28

[102] Marcel Medwed and Elisabeth Oswald. Template attacks on ECDSA. In *Information Security Applications, 9th International Workshop, WISA 2008, Jeju Island, Korea, September 23-25, 2008, Revised Selected Papers*, pages 14–27, 2008. 29

[103] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, October 1996. 4

[104] Thomas S. Messerges. *Power Analysis Attacks and Countermeasures for Cryptographic Algorithms*. PhD thesis, University of Illinois at Chicago, USA, 2000. 468 pages. 14

[105] Thomas S. Messerges. Securing the AES Finalists Against Power Analysis Attacks. In *Fast Software Encryption'00*, pages 150–164. Springer-Verlag, April 2000. New York. 21, 78

[106] Thomas S. Messerges. Using second-Order Power Analysis to Attack DPA resistant Software. In *CHES*, volume 1965 of *LNCS*, pages 71–77. Springer, August 17-18 2000. Worcester, MA, USA. 23, 130

[107] Thomas S. Messerges. Using Second-Order Power Analysis to Attack DPA Resistant Software. In *CHES*, volume 1965 of *LNCS*, pages 238–251. Springer-Verlag, August 17-18 2000. Worcester, MA, USA. 25, 91

[108] Thomas S. Messerges. Using Second-Order Power Analysis to Attack DPA Resistant Software. In Christof Paar and Çetin Kaya Koç, editors, *Cryptographic Hardware and Embedded Systems - CHES 2000*, volume 1965 of *LNCS*, pages 238–251. Springer, 2000. 26

[109] Thomas S. Messerges, Ezzy A. Dabbish, and Robert H. Sloan. Investigations of Power Analysis Attacks on Smartcards. In *USENIX — Smartcard'99*, pages 151–162, May 10–11 1999. Chicago, Illinois, USA. 61

[110] Victor S. Miller. Use of elliptic curves in cryptography. In *CRYPTO*, volume 218 of *Lecture Notes in Computer Science*, pages 417–426. Springer, August 18-22 1985. Santa Barbara, California, USA. 6

[111] Amir Moradi. Statistical tools flavor side-channel collision attacks. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT*, volume 7237 of *Lecture Notes in Computer Science*, pages 428–445. Springer, 2012. 139

[112] Amir Moradi, Sylvain Guilley, and Annelie Heuser. Detecting Hidden Leakages. In Ioana Boureanu, Philippe Owesarski, and Serge Vaudenay, editors, *ACNS*, volume 8479. Springer, June 10-13 2014. 12th International Conference on Applied Cryptography and Network Security, Lausanne, Switzerland. 64

[113] Amir Moradi and François-Xavier Standaert. Moments-correlating DPA. *IACR Cryptology ePrint Archive*, 2014:409, June 2 2014. 127, 139

[114] Amir Moradi and Alexander Wild. Assessment of hiding the higher-order leakages in hardware - what are the achievements versus overheads? In Güneysu and Handschuh (65), pages 453–474. 139

[115] Cédric Murdica. *Sécurité Physique de la Cryptographie sur Courbes Elliptiques*. PhD thesis, TELECOM-ParisTech & Secure-IC S.A.S., February 13 2014. Paris, France. Lien: `http://www.theses.fr/2014ENST0008`. 23

[116] Svetla Nikova, Christian Rechberger, and Vincent Rijmen. Threshold Implementations Against Side-Channel Attacks and Glitches. In *ICICS*, volume 4307 of *LNCS*, pages 529–545. Springer, December 4-7 2006. Raleigh, NC, USA. 21

[117] Svetla Nikova, Vincent Rijmen, and Martin Schläffer. Secure hardware implementation of nonlinear functions in the presence of glitches. *J. Cryptology*, 24(2):292–321, 2011. 56, 122

[118] NIST. AES Proposal: Rijndael (now FIPS PUB 197), April 2003. `http://csrc.nist.gov/archive/aes/rijndael/Rijndael-ammended.pdf`. 3

[119] NIST/ITL/CSD. Data Encryption Standard. FIPS PUB 46-3, Oct 1999. `http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf`. 3

[120] David Oswald and Christof Paar. Improving Side-Channel Analysis with Optimal Linear Transforms. In Stefan Mangard, editor, *CARDIS*, volume 7771 of *Lecture Notes in Computer Science*, pages 219–233. Springer, 2012. 37, 39, 57, 62

[121] Elisabeth Oswald and Stefan Mangard. Template Attacks on Masking — Resistance Is Futile. In Masayuki Abe, editor, *CT-RSA*, volume 4377 of *Lecture Notes in Computer Science*, pages 243–256. Springer, 2007. 23, 25, 91, 120, 130

[122] Elisabeth Oswald, Stefan Mangard, Christoph Herbst, and Stefan Tillich. Practical Second-Order DPA Attacks for Masked Smart Card Implementations of Block Ciphers. In Pointcheval (128), pages 192–207. 38

[123] Pascal Paillier and Ingrid Verbauwhede, editors. *Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-13, 2007, Proceedings*, volume 4727 of *LNCS*. Springer, 2007. 180, 185

[124] Jing Pan, Jerry I. den Hartog, and Jiqiang Lu. You cannot hide behind the mask: Power analysis on a provably secure S-box implementation. In Heung Youl Youm and Moti Yung, editors, *Information Security Applications, 10th International Workshop, WISA 2009, Busan, Korea, August 25-27, 2009, Revised Selected Papers*, volume 5932 of *Lecture Notes in Computer Science*, pages 178–192. Springer, 2009. 26, 27, 128

[125] Éric Peeters, François-Xavier Standaert, Nicolas Donckers, and Jean-Jacques Quisquater. Improved Higher-Order Side-Channel Attacks with FPGA Experiments. In *CHES*, volume 3659 of *LNCS*, pages 309–323. Springer, 2005. 120

[126] Andrea Pellegrini, Valeria Bertacco, and Todd M. Austin. Fault-based attack of RSA authentication. In *DATE*, pages 855–860. IEEE, 2010. 7

[127] Guilherme Perin, Laurent Imbert, Lionel Torres, and Philippe Maurine. Attacking Randomized Exponentiations Using Unsupervised Learning. In Prouff (129), pages 144–160. 29

[128] David Pointcheval, editor. *Topics in Cryptology - CT-RSA 2006, The Cryptographers' Track at the RSA Conference 2006, San Jose, CA, USA, February 13-17, 2006, Proceedings*, volume 3860 of *LNCS*. Springer, 2006. 188, 191

[129] Emmanuel Prouff, editor. *Constructive Side-Channel Analysis and Secure Design - 5th International Workshop, COSADE 2014, Paris, France, April 13-15, 2014. Revised Selected Papers*, volume 8622 of *Lecture Notes in Computer Science*. Springer, 2014. 183, 189

[130] Emmanuel Prouff and Matthieu Rivain. A Generic Method for Secure SBox Implementation. In Sehun Kim, Moti Yung, and Hyung-Woo Lee, editors, *WISA*, volume 4867 of *Lecture Notes in Computer Science*, pages 227–244. Springer, 2007. 21, 78

[131] Emmanuel Prouff and Matthieu Rivain. Theoretical and Practical Aspects of Mutual Information Based Side Channel Analysis. In Springer, editor, *ACNS*, volume 5536 of *LNCS*, pages 499–518, June 2-5 2009. Paris-Rocquencourt, France. 15

[132] Emmanuel Prouff and Matthieu Rivain. Masking against Side Channel Attacks: a Formal Security Proof. In *EUROCRYPT*, volume 7881 of *LNCS*, pages 142–159. Springer, May 2013. Athens, Greece. 88

[133] Emmanuel Prouff, Matthieu Rivain, and Régis Bevan. Statistical Analysis of Second Order Differential Power Analysis. *IEEE Trans. Computers*, 58(6):799–811, 2009. 23, 25, 63, 66, 80, 92, 95, 130, 156

[134] Emmanuel Prouff and Patrick Schaumont, editors. *Cryptographic Hardware and Embedded Systems - CHES 2012 - 14th International Workshop, Leuven, Belgium, September 9-12, 2012. Proceedings*, volume 7428 of *LNCS*. Springer, 2012. 181, 190

[135] Jean-Jacques Quisquater and David Samyde. ElectroMagnetic Analysis (EMA): Measures and Counter-Measures for Smard Cards. In I. Attali and T. P. Jensen, editors, *Smart Card Programming and Security (E-smart 2001)*, volume 2140 of *LNCS*, pages 200–210. Springer-Verlag, September 2001. Nice, France. ISSN 0302-9743. 7

[136] Jean-Jacques Quisquater and David Samyde. *Eddy current for Magnetic Analysis with Active Sensor*. Springer, 2002. 7

[137] Mathieu Renauld, François-Xavier Standaert, Nicolas Veyrat-Charvillon, Dina Kamel, and Denis Flandre. A formal study of power variability issues and side-channel attacks for nanoscale devices. In Kenneth G. Paterson, editor, *Advances in Cryptology - EUROCRYPT 2011 - 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011. Proceedings*, volume 6632 of *Lecture Notes in Computer Science*, pages 109–128. Springer, 2011. 39

[138] Oscar Reparaz, Benedikt Gierlichs, and Ingrid Verbauwhede. Selecting Time Samples for Multivariate DPA Attacks. In Prouff and Schaumont (134), pages 155–174. 39

[139] Kyung Hyune Rhee and DaeHun Nyang, editors. *Information Security and Cryptology - ICISC 2010 - 13th International Conference, Seoul, Korea, December 1-3, 2010, Revised Selected Papers*, volume 6829 of *Lecture Notes in Computer Science*. Springer, 2011. 192

[140] Matthieu Rivain. On the Exact Success Rate of Side Channel Analysis in the Gaussian Model. In *Selected Areas in Cryptography*, volume 5381 of *LNCS*, pages 165–183. Springer, August 14-15 2008. Sackville, New Brunswick, Canada. 31

[141] Matthieu Rivain and Emmanuel Prouff. Provably Secure Higher-Order Masking of AES. In Stefan Mangard and François-Xavier Standaert, editors, *CHES*, volume 6225 of *LNCS*, pages 413–427. Springer, 2010. 21, 78

[142] Matthieu Rivain, Emmanuel Prouff, and Julien Doget. Higher-Order Masking and Shuffling for Software Implementations of Block Ciphers. In *CHES*, volume 5747 of *Lecture Notes in Computer Science*, pages 171–188. Springer, September 6-9 2009. Lausanne, Switzerland. 104, 120, 172

[143] Ronald L. Rivest, M. J. B. Robshaw, and Yiqun Lisa Yin. Rc6 as the aes, 2000. 3

[144] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978. 5, 8

[145] Thomas Roche and Victor Lomné. Collision-Correlation Attack against Some 1st-Order Boolean Masking Schemes in the Context of Secure Devices. In Emmanuel Prouff, editor, *COSADE*, volume 7864 of *Lecture Notes in Computer Science*, pages 114–136. Springer, 2013. 88

[146] Akashi Satoh. Side-channel Attack Standard Evaluation Board, SASEBO-GII. Project of the AIST – RCIS (Research Center for Information Security), `http://www.rcis.aist.go.jp/special/SASEBO/SASEBO-GII-en.html` [Accessed on May 31, 2015]. 50

[147] Werner Schindler, Kerstin Lemke, and Christof Paar. A Stochastic Model for Differential Side Channel Cryptanalysis. In LNCS, editor, *CHES*, volume 3659 of *LNCS*, pages 30–46. Springer, Sept 2005. Edinburgh, Scotland, UK. 13, 58

[148] Jörn-Marc Schmidt and Michael Hutter. Optical and EM Fault-Attacks on CRT-based RSA: Concrete Results. In Johannes Wolkerstorfer Karl C. Posch, editor, *Austrochip 2007, 15th Austrian Workhop on Microelectronics, Graz, Austria, Proceedings*, pages 61–67. Verlag der Technischen Universität Graz, October 11th 2007. 7

[149] Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall, and Niels Ferguson. *The Twofish Encryption Algorithm: A 128-bit Block Cipher*. John Wiley & Sons, Inc., New York, NY, USA, 1999. 3

[150] Kai Schramm and Christof Paar. Higher Order Masking of the AES. In Pointcheval (128), pages 208–225. 102

[151] Claude E. Shannon. Communication theory of secrecy systems. *Bell System Technical Journal, Vol 28, pp. 656–715*, October 1949. 3

[152] Sergei P. Skorobogatov and Ross J. Anderson. Optical Fault Induction Attacks. volume 2523 of *LNCS*, pages 2–12. Springer, august 2002. CA USA. 7

[153] Arthur Sorkin. LUCIFER, a cryptographic algorithm. 8(1):22–42, January 1984. See also erratum, Cryptologia **7**, 1978, p. 118. 3

[154] Youssef Souissi, Shivam Bhasin, Sylvain Guilley, Maxime Nassar, and Jean-Luc Danger. Towards Different Flavors of Combined Side Channel Attacks. In Dunkelman (48), pages 245–259. 56

[155] Youssef Souissi, Nicolas Debande, Sami Mekki, Sylvain Guilley, Ali Maalaoui, and Jean-Luc Danger. On the Optimality of Correlation Power Attack on Embedded Cryptographic Systems. In Ioannis G. Askoxylakis, Henrich Christopher Pöhls, and Joachim Posegga, editors, *WISTP*, volume 7322 of *Lecture Notes in Computer Science*, pages 169–178. Springer, June 20-22 2012. 39

[156] Youssef Souissi, Maxime Nassar, Sylvain Guilley, Jean-Luc Danger, and Florent Flament. First Principal Components Analysis: A New Side Channel Distinguisher. In Rhee and Nyang (139), pages 407–419. 39

[157] Youssef Souissi, Maxime Nassar, Sylvain Guilley, Jean-Luc Danger, and Florent Flament. First Principal Components Analysis: A New Side Channel Distinguisher. In Rhee and Nyang (139), pages 407–419. 57

[158] François-Xavier Standaert and Cédric Archambeau. Using Subspace-Based Template Attacks to Compare and Combine Power and Electromagnetic Information Leakages. In *CHES*, volume 5154 of *Lecture Notes in Computer Science*, pages 411–425. Springer, August 10–13 2008. Washington, D.C., USA. 18, 39, 49, 56

[159] François-Xavier Standaert, Benedikt Gierlichs, and Ingrid Verbauwhede. Partition vs. Comparison Side-Channel Distinguishers: An Empirical Evaluation of Statistical Tests for Univariate Side-Channel Attacks against Two Unprotected CMOS Devices. In *ICISC*, volume 5461 of *LNCS*, pages 253–267. Springer, December 3-5 2008. Seoul, Korea. 15

[160] François-Xavier Standaert, Tal Malkin, and Moti Yung. A Unified Framework for the Analysis of Side-Channel Key Recovery Attacks. In *EUROCRYPT*, volume 5479 of *LNCS*, pages 443–461. Springer, April 26-30 2009. Cologne, Germany. 133

[161] François-Xavier Standaert, Tal G. Malkin, and Moti Yung. A unified framework for the analysis of side-channel key recovery attacks. Cryptology ePrint Archive, Report 2006/139, 2006. `http://eprint.iacr.org/`. 30

[162] François-Xavier Standaert, Nicolas Veyrat-Charvillon, Elisabeth Oswald, Benedikt Gierlichs, Marcel Medwed, Markus Kasper, and Stefan Mangard. The World is Not Enough: Another Look on Second-Order DPA. In *ASIACRYPT*, volume 6477 of *LNCS*, pages 112–129. Springer, December 5-9 2010. Singapore. `http://www.dice.ucl.ac.be/~fstandae/PUBLIS/88.pdf`. 21, 78

[163] François-Xavier Standaert, Gilles Piret, Gaël Rouvroy, Jean-Jacques Quisquater, and Jean-Didier Legat. ICEBERG : An involutional cipher efficient for block encryption in reconfigurable hardware. In Bimal K. Roy and Willi Meier, editors, *Fast Software Encryption, 11th International Workshop, FSE 2004, Delhi, India, February 5-7, 2004, Revised Papers*, volume 3017 of *Lecture Notes in Computer Science*, pages 279–299. Springer, 2004. 117

[164] François-Xavier Standaert, Nicolas Veyrat-Charvillon, Elisabeth Oswald, Benedikt Gierlichs, Marcel Medwed, Markus Kasper, and Stefan Mangard. The world is not enough: Another look on second-order DPA. In Masayuki Abe, editor, *Advances in Cryptology - ASIACRYPT 2010 - 16th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 5-9, 2010. Proceedings*, volume 6477 of *Lecture Notes in Computer Science*, pages 112–129. Springer, 2010. 120, 135

[165] Daehyun Strobel, David Oswald, Bastian Richter, Falk Schellenberg, and Christof Paar. Microcontrollers as (in)security devices for pervasive computing applications. *Proceedings of the IEEE*, 102(8):1157–1173, 2014. 39

[166] Alan Stuart and Keith Ord. *Kendall's Advanced Theory of Statistics: Distribution Theory*. Wiley-Blackwell, June 2 1994. 6th Edition. ISBN-10: 0470665300; ISBN-13: 978-0470665305. 125

[167] Takeshi Sugawara, Naofumi Homma, Takafumi Aoki, and Akashi Satoh. Profiling attack using multivariate regression analysis. *IEICE Electronics Express*, 7(15):1139–1144, 2010. 39, 43

[168] TELECOM ParisTech. DPA Contest, 2nd edition. `http://www.DPAcontest.org/v2/` [Accessed on May 31, 2015]. 38, 50

[169] TELECOM ParisTech SEN research group. DPA Contest (4th edition), 2013–2014. `http://www.DPAcontest.org/v4/`. 11, 57, 64, 172

[170] Adrian Thillard, Emmanuel Prouff, and Thomas Roche. Success through Confidence: Evaluating the Effectiveness of a Side-Channel Attack. In Guido Bertoni and Jean-Sébastien Coron, editors, *CHES*, volume 8086 of *Lecture Notes in Computer Science*, pages 21–36. Springer, 2013. 31

[171] Elena Trichina. Combinational Logic Design for AES SubBytes Transformation on Masked Data, 2003. Not published elsewhere. e.v.trichina@samsung.com 12368 received 11 Nov 2003. 21

[172] Elena Trichina and Antonio Bellezza. Implementation of elliptic curve cryptography with built-in counter measures against side channel attacks. In Burton S. Kaliski, Jr., Çetin Koç, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems - CHES 2002*, volume 2523 of *Lecture Notes in Computer Science*, pages 297–312. Springer Berlin / Heidelberg, 2003. 22

[173] Michael Tunstall, Carolyn Whitnall, and Elisabeth Oswald. Masking Tables - An Underestimated Security Risk. *IACR Cryptology ePrint Archive*, 2013:735, 2013. 26

[174] Michael Tunstall, Carolyn Whitnall, and Elisabeth Oswald. Masking Tables - An Underestimated Security Risk. In Shiho Moriai, editor, *FSE*, volume 8424 of *Lecture Notes in Computer Science*, pages 425–444. Springer, 2013. 27, 81, 86, 117, 128, 130

[175] Nicolas Veyrat-Charvillon, Benoît Gérard, Mathieu Renauld, and François-Xavier Standaert. An optimal Key Enumeration Algorithm and its Application to Side-Channel Attacks. Cryptology ePrint Archive, Report 2011/610, 2011. `http://eprint.iacr.org/2011/610/`. 30

[176] Nicolas Veyrat-Charvillon, Marcel Medwed, Stéphanie Kerckhof, and François-Xavier Standaert. Shuffling against Side-Channel Attacks: A Comprehensive Study with Cautionary Note. In Xiaoyun Wang and Kazue Sako, editors, *ASIACRYPT*, volume 7658 of *Lecture Notes in Computer Science*, pages 740–757. Springer, 2012. 120

[177] Jason Waddle and David Wagner. Towards Efficient Second-Order Power Analysis. In *CHES*, volume 3156 of *LNCS*, pages 1–15. Springer, 2004. Cambridge, MA, USA. 23, 88, 130

[178] Jason Waddle and David Wagner. Fault Attacks on Dual-Rail Encoded Systems. In *ACSAC*, pages 483–494. IEEE Computer Society, 2005. 7

[179] Colin D. Walter. Sliding Windows Succumbs to Big Mac Attack. In Çetin Kaya Koç, David Naccache, and Christof Paar, editors, *CHES*, volume 2162 of *Lecture Notes in Computer Science*, pages 286–299. Springer, 2001. 29

[180] Eric W Weisstein. Cumulant. From MathWorld–A Wolfram Web Resource. `http://mathworld.wolfram.com/Cumulant.html`. 125

[181] Carolyn Whitnall and Elisabeth Oswald. A Fair Evaluation Framework for Comparing Side-Channel Distinguishers. *J. Cryptographic Engineering*, 1(2):145–160, 2011. 60, 100

[182] Michael J. Wiener, editor. *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*. Springer, 1999. 179, 185

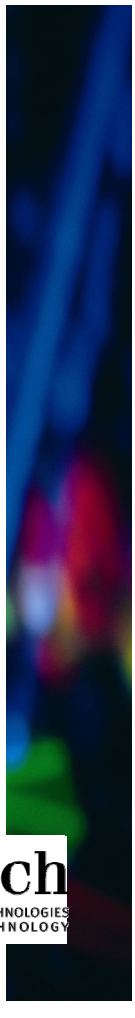
[183] Sung-Ming Yen and Marc Joye. Checking before output may not be enough against fault-based cryptanalysis. *IEEE Trans. Comput.*, 49(9):967–970, 2000. 7

Attaques par Canaux Auxiliaires Multivariées, Multi-cibles et d'ordre élévé

Nicolas BRUNEAU

ABSTRACT : Side Channel Attacks are a classical threat against cryptographic algorithms in embedded systems. They aim at exploiting the physical leakages unintentionally emitted by the devices during the execution of their embedded programs to recover sensitive data. As such attacks represent a real threat against embedded systems different countermeasures have been developed. In thesis we investigate their security in presence of multiple leakages. Indeed there often are in the leakage measurements several variables which can be exploited to mount Side Channel Attacks. In particular we show in this thesis the optimal way to exploit multiple leakages of a unique variable. This dimensionality reduction comes with no loss on the overall exploitable information. Based on this result we investigate further how such dimensionality reduction methods can be applied in the case of protected implementations. We show that the impact of such methods increases with the security "level" of the implementation. We also investigate how to exploit the leakages of multiple variables in order to improve the results of Side Channel Analysis. We start by improving the attacks against masking schemes, with a precomputed table recomputation step. Some protections have been developed to protect such schemes. As a consequence we investigate the security provided by these protections. In this context we present results which show that the main parameter to evaluate the security of the masking schemes is not sufficient to estimate the global security of the implementation. Finally we show that in the context of masking scheme with shuffling the optimal attack is not computable. As a consequence we present a truncated version of this attack with a better effectiveness.

KEY-WORDS : Side Channel Attacks, Masking scheme, Multivariate Attacks, Optimal Attack



TELECOM ParisTech

EDiTE

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY