Journal de la Société Française de Statistique

Application du coefficient d'information maximal à la cryptanalyse par canaux cachés

Title: Application of the Maximal Information Coefficent for side channel cryptanalysis

Yanis Linge[1,2], Cécile Dumas[1] et Sophie Lambert-Lacroix[2]

Résumé :

Dans le domaine des cartes à puce, les signaux émis par un composant pendant un calcul cryptographique peuvent compromettre la confidentialité des informations liées directement ou indirectement aux clés utilisées. Dans cet article nous commencerons par présenter les attaques par observations qui utilisent des méthodes statistiques afin d'exploiter ces signaux compromettants. Nous proposerons ensuite une nouvelle attaque utilisant une mesure de dépendance proposée récemment (Reshef et al., 2011), le Coefficient Maximal d'Information. Enfin nous comparerons l'ensemble des attaques présentées.

Abstract:

In smart card domain, the emanations of a component during a cryptographic computation may compromise the information that is directly or not linked to the secret keys. In this article we will first present the side channel attacks which use statistical methods to exploit the side channel. Then we will propose a new attack based on a measure of dependence exposed recently (Reshef et al., 2011), the Maximal Information Coefficient. Finaly we will compare the different attacks presented.

Mots-clés : Attaques par canaux cachés, Carte à puce, Corrélation, Cryptographie, Cryptanalyse, Information mutuelle, Mesure de dépendance

Keywords: Side Channel Analysis, Smart card, Correlation, Cryptography, Cryptanalysis, Mutual Information, Measure dependency

Classification AMS 2000 : 94A60

1. Introduction

La cryptologie, du grec Kruptos (caché) et Logos (discours), signifie la science du secret. La cryptologie est une science millénaire. Elle est sans doute née en même temps que le langage. En effet, la capacité pour un groupe de personnes, à communiquer sans que leurs ennemis puissent intercepter des informations sensibles, peut décider de la physionomie d'un conflit. Lors de la Seconde Guerre Mondiale, par exemple, les alliés ont été capables de briser le secret des communications allemandes grâce au travail acharné d'une équipe de cryptologues menée par Alan Turing. La réussite de cette équipe a été un des tournants de cette guerre (Singh, 1999).

[1] CEA-LETI/MINATEC, 17 rue des Martyrs,
38054 Grenoble Cedex 9, France.
E-mail : yanis.linge@cea.fr ; cecile.dumas@cea.fr

[2] UJF-Grenoble 1 / CNRS / UPMF / TIMC-IMAG
UMR 5525, Grenoble, F-38041, France.
E-mail : Sophie.Lambert@imag.fr

 ISSN: 2102-6238

L'avènement de l'informatique et de l'internet a vu exploser les techniques de cryptologie. Aujourd'hui, chacun utilise quotidiennement un grand nombre d'algorithmes cryptographiques sans en avoir conscience.

L'étude de la cryptologie s'articule autour de quatre grands piliers :

- La confidentialité : qui permet de communiquer un message dont la lecture ne sera possible que par un petit nombre d'entités choisies.
- L'authenticité : qui permet de s'assurer qu'un individu est bien l'auteur d'un message.
- L'intégrité : qui permet de s'assurer que le message n'a pas été modifié sans autorisation ou par erreur.
- La non-répudiation : qui permet de prouver que le message a bien été envoyé et/ou reçu.

Classiquement la cryptologie se divise en deux grandes branches : la **cryptographie** et la **cryptanalyse**.

La **cryptographie**, ou écriture secrète, consiste à élaborer des méthodes permettant d'assurer au moins l'un des quatre piliers précédents. Elle se divise en deux grandes familles d'algorithmes de chiffrement : les algorithmes symétriques et les algorithmes asymétriques.

La cryptographie symétrique, encore appelée cryptographie à clé secrète, est la première à avoir vu le jour. Son principe est d'utiliser une clé commune pour les opérations de chiffrement et de déchiffrement. La sécurité de la communication est alors assurée par le secret de cette clé. Ces algorithmes sont capables de chiffrer et de déchiffrer une quantité très importante de données en peu de temps. L'avantage de tels algorithmes est le rapport entre la taille des clés utilisées (moins de 256 bits généralement) et la robustesse des algorithmes. Le défaut principal de la cryptographie symétrique est la gestion des clés. En effet, il faut que les deux parties qui souhaitent communiquer conviennent d'une clé commune. Aujourd'hui, chacun utilise quotidiennement un grand nombre de communications chiffrées (téléphone cellulaire, distributeur de billets, badge d'accès, etc.). Il n'est donc plus possible de rencontrer la personne avec laquelle on veut communiquer afin de s'accorder sur une clé commune, comme le faisait par exemple les présidents des États-Unis et de l'U.R.S.S durant la Guerre Froide pour le fameux *Téléphone rouge* (Singh, 1999). Il a donc fallu inventer une nouvelle famille d'algorithmes de chiffrement afin de pouvoir contourner ce problème.

C'est en 1976 que Withfield Diffie et Martin Hellman (Diffie and Hellman, 1976) ont introduit le concept de cryptographie asymétrique ou cryptographie à clé publique. Malheureusement, leur article ne proposait pas réellement d'algorithme et il a fallu attendre 1978 et l'invention du *RSA* (Rivest et al. (1978) - pour Rivest Shamir Adleman) pour voir émerger cette nouvelle famille d'algorithmes. Les algorithmes à clé publique utilisent deux clés différentes, l'une pour chiffrer, l'autre pour déchiffrer et sont basés sur des problèmes mathématiques réputés difficiles à résoudre (la factorisation pour le *RSA* et le logarithme discret pour El-Gamal (Gamal, 1984) pour ne citer qu'eux). Pour chiffrer un message que l'on souhaite envoyer à un correspondant, il suffit de connaître sa clé publique qui servira à chiffrer le message. Dès lors, le message est illisible même avec la connaissance de cette clé. Lorsque le correspondant reçoit un message chiffré, il utilise sa clé privée, connue de lui seul, afin de le déchiffrer. La cryptographie à clé publique souffre de deux défauts majeurs. D'une part, il est difficile d'être sûr que l'on utilise bien la clé publique de notre correspondant et non celle d'une personne mal intentionnée et d'autre part, ces algorithmes sont plutôt lents et leur sécurité nécessite l'utilisation de clés de taille très

 ISSN: 2102-6238

importante (la recommandation de l'*ANSSI* [1] en 2010 est d'utiliser des clés de taille supérieure ou égale à 2048 bits pour l'algorithme *RSA*). L'application la plus courante de ces algorithmes est ce que l'on appelle le protocole d'échange de clé. Il permet à deux interlocuteurs de se mettre d'accord sur une clé commune afin de pouvoir utiliser un algorithme de chiffrement à clé privée pour pouvoir échanger plus de données.

D'une manière générale, il est très important que la sécurité des algorithmes de chiffrement repose bien sur la clé secrète et non sur le secret de l'algorithme. Ce principe a été introduit par Kerckhoffs dès la fin du XIX^e siècle (Kerckhoffs, 1883a,b). Lors de la Seconde Guerre Mondiale, les allemands et leurs alliés utilisaient pour chiffrer leur communication la machine *Enigma*. La sécurité de cette machine était basée principalement sur le secret de l'algorithme. Alan Turing et les alliés ont été capables de briser le chiffrement des communications allemandes grâce, en partie, à l'obtention de machines *Enigma* et à l'étude de l'algorithme secret utilisé par ces machines (Singh, 1999). Les algorithmes de chiffrement standard actuels sont tous publics. Ainsi chacun est libre de les tester et d'essayer de les attaquer. La force de ces algorithmes est d'avoir été testés par un grand nombre de personnes.

La seconde branche de la cryptologie, la **cryptanalyse**, vise à analyser les méthodes proposées par la cryptographie en vue de retrouver les clés secrètes utilisées ou à défaut d'être capable de retrouver un texte clair à partir d'un texte chiffré. On distingue deux types de cryptanalyse : la cryptanalyse mathématique et la cryptanalyse physique.

La cryptanalyse mathématique étudie la résistance des algorithmes en s'intéressant plus particulièrement à leurs propriétés mathématiques. Cette partie a été la première à avoir été explorée et développée.

La cryptanalyse dite physique, quant à elle, étudie plus particulièrement les faiblesses liées aux composants sur lesquels sont implantés les algorithmes cryptographiques.

Il existe de nombreux composants contenant des algorithmes cryptographiques, les cartes à puce, bien sûr, mais aussi les puces RFID, certains processeurs de nos ordinateurs, etc. Pour le lecteur désireux d'en savoir un peu plus sur la cryptographie, les ouvrages (Dubertret, 1998; Ferguson and Schneier, 2003; Dumas et al., 2007; Mangard et al., 2007; Menezes et al., 2010) permettent d'approfondir le sujet.

Dans la suite de cet article, nous nous intéresserons plus particulièrement à un type particulier de cryptanalyse physique, la cryptanalyse physique par observations.

Jusqu'au milieu des années 90, la sécurité des algorithmes cryptographiques implantés sur un composant était considérée comme équivalente à celle de l'algorithme. Un composant était alors étudié comme une boite noire. Or, la façon dont est implémenté un algorithme est très importante. En 1996, Paul Kocher a introduit les premières attaques par observations (Kocher et al., 1999) et a ainsi montré le rôle primordial de l'implémentation dans la sécurité d'un composant. L'idée de Kocher était d'utiliser la consommation électrique d'une carte à puce en vue de retrouver des informations secrètes. De nos jours, on utilise aussi d'autres émanations du composant, comme le rayonnement électromagnétique ou la température du composant. Dans la suite, ces émanations seront appelées *fuites*.

La Figure 1 représente la consommation électrique d'un composant lorsque que l'on utilise le *DES* (pour Data Encryption Standard) pour chiffrer une donnée. On peut y observer 16 motifs qui

[1] L'ANSSI assure la mission d'autorité nationale en matière de sécurité des systèmes d'information.

 ISSN: 2102-6238

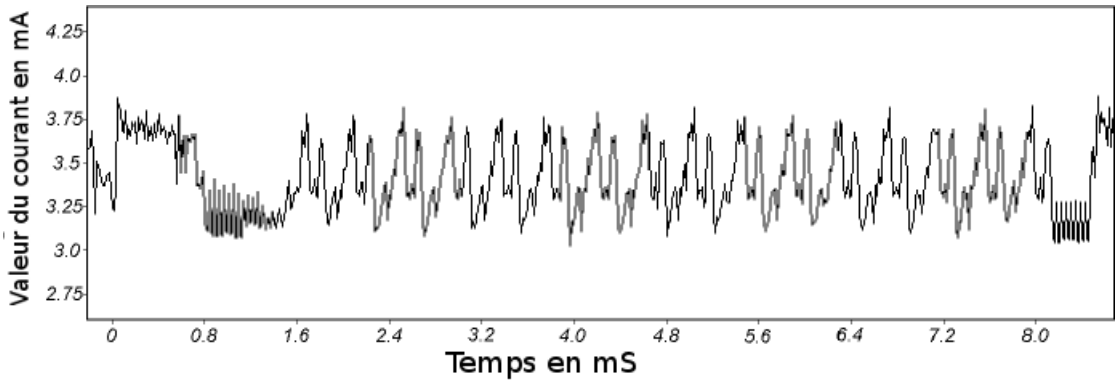

FIGURE 1. *Consommation électrique en mA d'un composant lors d'un chiffrement DES, extrait de l'article (Kocher et al., 1999).*

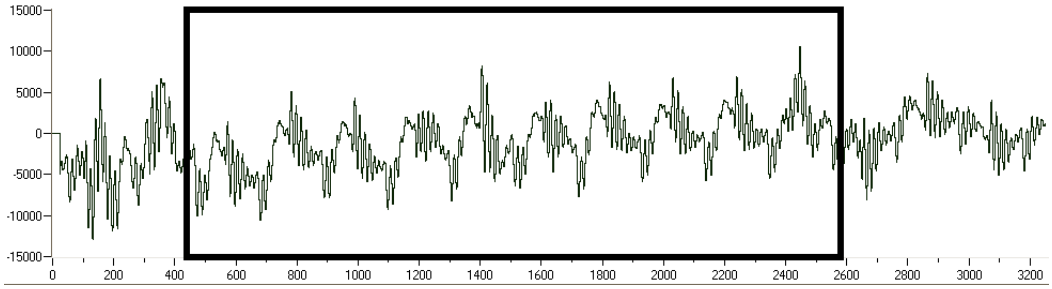

FIGURE 2. *Consommation électrique d'un composant lors d'un chiffrement AES (ParisTech, 2010).*

semblent beaucoup se ressembler et qui représentent les 16 itérations composant le *DES*.

La Figure 2 est issue du DPAContestV2 (ParisTech, 2010) et représente la consommation électrique du composant lorsqu'il chiffre des données grâce à un *AES* (pour Advanced Encryption Standard). On observe ici 10 motifs qui sont proches et représentent les 10 rondes de l'*AES*.

La mesure des émanations d'un composant peut permettre de reconnaitre quel algorithme est utilisé par le composant. Ces émanations sont liées aux opérations que le composant effectue mais aussi aux données qu'il manipule.

Une carte à puce ne peut pas manipuler la clé secrète toute entière car elle est bien trop grande. Actuellement, la plupart des processeurs embarqués sur les cartes à puce permet de manipuler des objets codés sur 8, 16 ou 32 bits. Cette particularité, liée au composant, permet à un attaquant d'utiliser la méthode *diviser pour mieux régner*, c'est-à-dire, dans notre cas, essayer de retrouver la clé par morceaux. Si l'on prend l'exemple d'un algorithme de chiffrement symétrique standard comme l'*AES*, les clés utilisées à chacune des 10 rondes sont composées de 128 bits, et il n'est pas possible d'énumérer les 2^{128} clés possibles. En revanche, si l'on divise ces clés en 16 morceaux de 8 bits chacun, il n'y a plus que $16 \times 2^8 = 4096$ clés à énumérer ce qui ne pose plus aucun problème.

La façon dont sont manipulées les données par le composant offre donc à un attaquant la

 ISSN: 2102-6238

possibilité d'utiliser la méthode *diviser pour mieux régner*. Ainsi, Kocher décrit la *SPA* (pour Simple Power Analysis), une méthode permettant de retrouver bit à bit la clé utilisée dans un algorithme *RSA*.

L'algorithme *RSA* est un algorithme de chiffrement asymétrique, dont la sécurité est basée sur la difficulté de factoriser des grands nombres. La construction des clés *RSA* peut être décrite de la manière suivante :

- Choisir deux nombres p et q premiers, leur produit pq sera noté n.
- Calculer l'indicatrice d'Euler de n : $\varphi(n) = (p-1) \times (q-1)$.
- Choisir e un entier premier avec $\varphi(n)$, e sera appelé l'exposant de chiffrement.
- Trouver un entier d tel que $d.e \equiv 1 \bmod \varphi(n)$. [2]

Le couple (n,e) sera la clé publique tandis que le couple (n,d) sera la clé privée.

On a maintenant à notre disposition une clé publique et une clé privée. Soit X un entier plus petit que n, qui représente le message que l'on souhaite chiffrer. Le message chiffré, noté C, sera calculé grâce à :

$$C \equiv X^e \bmod n.$$

Pour déchiffrer le message, qui sera un entier plus petit que n, il suffit au propriétaire de la clé secrète de calculer :

$$C^d \equiv X^{ed} \bmod \mathrm{n} \equiv X \bmod \mathrm{n}.$$

Cet algorithme est sûr si l'on considère que les nombres p et q sont *bien choisis* (Kaihara et al., 2009). En effet, il est très difficile de déterminer $\varphi(n)$ lorsque l'on ne connaît pas les nombres p et q, c'est-à-dire lorsque l'on n'est pas capable de factoriser n. Cet algorithme est très utilisé dans la cryptographie moderne.

Le problème est que lorsque que l'on va chercher à implémenter un algorithme *RSA*, on va vouloir faire en sorte qu'il soit le plus efficace possible et pour cela, essayer d'optimiser le calcul de l'exponentiation modulaire. Dans le cas de Kocher, cette opération était implémentée en utilisant l'algorithme dit exponentiation binaire *Left-to-Right* (Knuth, 1981). Dans cet algorithme, on commence par écrire e sous sa représentation binaire :

$$e = \sum_{i=0}^{l-1} a_i 2^i = a_{l-1}2^{l-1} + a_{l-2}2^{l-2} + \cdots + a_1 \cdot 2 + a_0$$

Dans cette représentation, on sait que $a_{l-1} = 1$ et que pour tout $i \geq l$, $a_i = 0$. Le calcul de M^e peut être donné par :

$$M^e = \prod_{i=0}^{l-1} \left(M^{2^i}\right)^{a_i} = (((\ldots(M^{a_{l-1}})^2 \times M^{a_{l-2}})^2 \ldots \times M^{a_i})^2 \ldots \times M^{a_1})^2 \times M^{a_0}.$$

L'algorithme d'exponentiation binaire *Left-to-Right* est décrit dans l'Algorithme 1.

Dans cet algorithme on observe que lorsqu'un bit de clé est nul, on a un calcul de moins à effectuer que pour un bit de clé valant 1. Si l'émanation du composant est différente entre l'opération *multiplication* et l'opération *mise au carré* on sera capable de distinguer si le composant est en train de traiter un bit nul ou un bit non nul. La Figure 3 illustre ce phénomène : elle montre les

[2] Le théorème de Bachet-Bezout permet de prouver l'existence d'un tel inverse

 ISSN: 2102-6238

Algorithm 1 Algorithme de l'exponentiation binaire *Left-to-Right*

Algorithme POWMOD(M, e, n)
  $C = M$
  pour $i \in \{l-2, \cdots, 0\}$ **faire**
    $C = C \times C$ mod n ▷ *Mise au carré.*
    si $a_i = 1$ **alors**
      $C = C \times M$ mod n ▷ *Multiplication.*
    fin si
  fin pour
retourner C
fin Algorithme

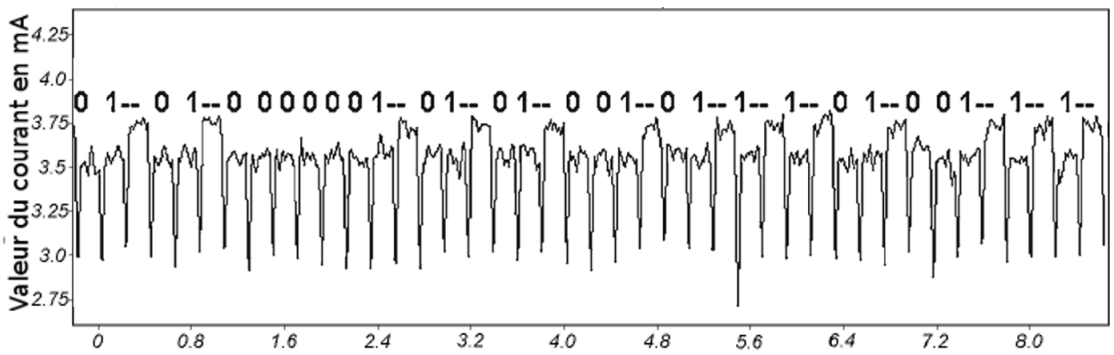

FIGURE 3. *Exemple d'une attaque SPA sur l'algorithme d'exponentiation binaire. En abscisse le temps en ms et en ordonnée la consommation électrique en mA (Kocher et al., 1999)*

résultats qu'avait obtenus Kocher en étudiant les fuites d'un algorithme *RSA* implémenté sous cette forme. Il est aisé, même à l'œil nu, de lire la clé utilisée par le composant, bit à bit. En particulier, on pourra retrouver un motif commun dans la consommation électrique du composant lorsque l'algorithme effectue une mise au carré et un autre motif pour la multiplication. Comme l'Algorithme 1 le laissait présager, pour traiter un bit valant 1, le composant a besoin de plus de temps car il a deux opérations à effectuer alors qu'il n'en fait qu'une pour un bit à 0, c'est pour cela que l'on observe un second pic suivant le motif commun lorsque le bit de clé manipulé vaut 1.

Il est possible de se prémunir de cette analyse en ajoutant des *calculs fantômes* (Clavier and Joye, 2001) dans les algorithmes afin que la carte à puce effectue dans tous les cas le même nombre de cycles. Néanmoins on peut adapter la *SPA* pour obtenir des attaques efficaces même sur des implémentations bénéficiant de ce genre de protection.

La *SPA* est l'une des attaques par canaux cachés la plus simple et met en exergue l'existence d'un lien entre les émanations du composant à un instant donné et les opérations effectuées à cet instant.

La seconde attaque proposée par Kocher, appelée *DPA* (pour Differential Power Analysis), est plus complexe et fait appel à des notions de statistique. Cette attaque permet d'obtenir des résultats même lorsque la *SPA* ne permet pas d'identifier d'informations dans les émanations du composant que l'on cible. L'idée est non plus de regarder une unique courbe à l'œil nu, mais

 ISSN: 2102-6238

plusieurs et d'en retirer de l'information grâce à des méthodes statistiques.

De même que pour la *SPA*, cette attaque vise une opération intermédiaire de l'algorithme qui n'implique qu'une petite partie de la clé secrète. Pour cette analyse, l'attaquant va procéder en plusieurs phases. Dans un premier temps, il va demander au composant visé d'effectuer des opérations cryptographiques et enregistrer les émanations qui en résultent. Parallèlement, il construit pour chaque valeur de clés possibles un modèle représentant la fuite supposée en sortie d'une fonction intermédiaire choisie par l'attaquant. Dans Kocher et al. (1999), Kocher propose d'utiliser un modèle monobit en sortie des boites-S du *DES*. Enfin, l'attaquant étudie les relations possibles entre les différents modèles et les émanations issues du composant. Dans le cas de Kocher il suffit de faire la différence entre deux moyennes comme nous le présenterons dans la Section 2.2. La fuite étant liée à la clé secrète, le modèle qui sera le plus lié aux émanations sera celui obtenu avec la clé secrète.

Le but de cet article est de présenter les différentes méthodes statistiques mises en œuvre pour étudier le lien entre l'émanation et les modèles. Dans la suite de cette article, nous commencerons par introduire quelques notations afin d'offrir un cadre plus formel à notre analyse. Nous présenterons ensuite les attaques basées sur la corrélation puis celles basées sur les lois de probabilités sous-jacentes et nous présenterons en particulier l'utilisation du coefficient maximal d'information dans le cadre des attaques par observations. Dans la Section 3, avant de conclure, nous présenterons les résultats que nous avons obtenus sur des données réelles.

2. Méthodes basées sur des mesures de dépendance : les attaques par observations

2.1. Notations

On notera $g : \mathscr{X} \times \mathscr{K} \to \mathscr{Z}$ la fonction intermédiaire visée par l'attaque. $\mathscr{K} = \{0, \cdots, \mathbf{k}-1\}$ est l'ensemble des valeurs possibles de la clé et $k^\star$ est la valeur de la clé secrète. Cette ensemble $\mathscr{K}$ ne contient qu'un petit nombre d'éléments (généralement moins de 256) car l'attaquant vise une fonction qui n'utilise qu'une partie de la clé secrète. $\mathscr{X} = \{0, \cdots, \mathbf{x}-1\}$ est l'ensemble des entrées possibles de g et $\mathscr{Z} = \{0, \cdots, \mathbf{z}-1\}$ celui de la sortie de g. Soit X une variable aléatoire qui représente l'entrée de la fonction intermédiaire, et Z la variable aléatoire représentant le résultat de cette fonction. On considère que l'attaquant a accès aux valeurs de la variable aléatoire X. On dira que les attaques par observations sont des attaques à textes clairs connus.

La fonction intermédiaire g est connue par l'attaquant, déterministe, et dépend d'un paramètre $k \in \mathscr{K}$. Elle peut être très diverse : une permutation, une addition modulo 256, une fonction non linéaire, etc. Finalement, on notera $L(Z) = L(g(X,k)) = \Phi(X,k)$, la variable aléatoire représentant la fuite générée par le calcul de Z. La fonction Φ est inconnue et non déterministe, car elle représente une mesure physique.

Le but d'une attaque par observations est d'estimer la clé secrète $k^\star$. On dispose pour cela de n réalisations notées $l_i = \phi(x_i, k^\star)$, $i = 1, \cdots, n$ de $L(Z) = \Phi(X, k^\star)$. On considère que ces réalisations proviennent de variables aléatoires indépendantes. Ces réalisations sont obtenues à partir de mesures physiques fortement bruitées. D'autre part, on modélise par la fonction $m(X,k)$, la fuite pour une clé $k \in \mathscr{K}$. Notons que les valeurs de la fonction $m(X,k)$ sont discrètes. Plus cette modélisation sera proche de Φ, meilleure sera notre estimation de $k^\star$.

 ISSN: 2102-6238

Lors d'une attaque par observation on doit choisir une fonction qui représente la manière dont fuit le composant en fonction des variables manipulées. On notera cette fonction φ.

Les bits des variables manipulées sont stockés dans les composants grâce à la présence ou non de courant dans les transistors. Lorsque l'on change la valeur d'une variable, ces transistors se déchargent ou se chargent afin de stocker la valeur voulue. Le *poids de Hamming* (Π) est une fonction qui prend en entrée une suite de bits et qui renvoie le nombre de bits non nuls, c'est à dire le nombre de composants à charger pour stocker la suite de bits. Dans le cadre d'une attaque par observations, la fuite est souvent liée au poids de Hamming des données manipulées en raison de la façon dont elles sont stockées, c'est pourquoi ce modèle de fuite est tant utilisé. La variable $m(X,k)$ est calculée à partir de la variable X, de l'algorithme et de la manière dont les données manipulées laisseront fuir de l'information φ. Par exemple, si l'on considère que le modèle de fuite est le poids de Hamming, on aura $m(X,k) = \varphi(g(X,k)) = \Pi(g(X,k))$. La variable $m(X,k)$ sera de nature discrète. On notera $\mathscr{M}$ l'ensemble des valeurs possibles du modèle.

On dispose de n réalisations $m(x_i,k)$ pour différentes clés à tester dans un ensemble $\mathscr{K}$. A partir de ces données, l'attaquant va chercher à déterminer une mesure de *dépendance* entre les variables $L(Z)$ et $m(X,k)$. On retiendra la (ou les) clé(s) qui maximise(nt) cette mesure de *dépendance* comme une estimation de $k^\star$. Les attaques par observations visent à mesurer la dépendance entre une variable continue, les émanations d'un composant lors de calcul cryptographique et une variable discrète, un modèle de fuite pour chaque sous-clé possible.

2.2. *Différence de moyennes*

L'attaque originelle proposée par Kocher dans Kocher et al. (1999) est basée sur la différence de moyennes. Kocher propose d'attaquer la valeur d'un bit de la sortie d'une boite-S du *DES* lors de la première ronde. Ainsi, les variables aléatoires $m(X,k)$, $k \in \mathscr{K}$, n'ont ici que deux valeurs possibles, 0 et 1. L'attaquant souhaite estimer la différence $E[L|m(X,k)=1] - E[L|m(X,k)=0]$. Pour cela, il calcule :

$$\hat{\Delta}_k = \frac{\sum_{i:m(x_i,k)=1} l_i}{\#\{i : m(x_i,k)=1\}} - \frac{\sum_{i:m(x_i,k)=0} l_i}{\#\{i : m(x_i,k)=0\}} = \frac{\sum_{i=1}^n m(x_i,k) l_i}{\sum_{i=1}^n m(x_i,k)} - \frac{\sum_{i=1}^n (1-m(x_i,k) l_i)}{\sum_{i=1}^n (1-m(x_i,k))}.$$

Lorsque $m(X,k)$ et $L(Z)$ sont indépendantes, $E[L|m(X,k)=1] = E[L|m(X,k)=0]$. Lorsque Δ_k est proche de 0, $m(X,k)$ et $L(Z)$ sont considérées comme étant "indépendantes". En revanche, lorsque la valeur de Δ_k est élevée, il est probable qu'il existe une relation de dépendance entre les deux variables qui sont considérées. Ainsi, une estimation de la bonne clé sera donnée par :

$$\hat{k^*} = \underset{k \in \mathscr{K}}{argmax}(|\hat{\Delta}_k|).$$

La variable aléatoire modélisant la fuite ne peut prendre ici que deux valeurs, ce qui est extrêmement restrictif. En effet, les données calculées sont le plus souvent constituées de plusieurs bits. Le modèle proposé est donc très éloigné de Φ. L'une des premières améliorations proposées vise à permettre un plus vaste choix pour les valeurs du modèle. De plus, le nombre de réalisations nécessaires pour pouvoir découvrir une relation entre le modèle correspondant à la bonne clé $m(X,k^\star)$ et la fuite $L(Z)$ est important dans le contexte des attaques par canaux cachés. Enfin, la recherche académique concernant les attaques par observations, a été longtemps axée autour

 ISSN: 2102-6238

de la généricité des relations entre les deux variables aléatoires considérées (linéaire, d'ordre, fonctionnelle, etc.). Les mesures de dépendance proposées pour améliorer l'attaque de Kocher peuvent être divisées en deux grandes familles. La première est la détection de structure de corrélation entre les observations. La seconde est basée sur les lois de probabilités sous-jacentes aux observations.

2.3. Méthodes basées sur la corrélation

2.3.1. Correlation Power Analysis : corrélation linéaire estimée grâce au coefficient de Pearson

Lorsque l'on parle de *dépendance* entre deux variables aléatoires, la première mesure qui vient à l'esprit est celle donnée par le coefficient de corrélation linéaire entre $m(x_\cdot,k)$ et $l_\cdot$. Ce coefficient est défini par :

$$\rho_k = \frac{\frac{1}{n}\sum_{i=1}^{n}(l_i-\bar{l})(m(x_i,k)-\bar{m_k})}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(l_i-\bar{l})^2\frac{1}{n}\sum_{i=1}^{n}(m(x_i,k)-\bar{m_k})^2}}. \tag{1}$$

où $\bar{l}$ est la moyenne empirique associée aux l_i et $\bar{m_k}$ celle associée aux $m(x_i,k)$. Les valeurs $\rho_k = 1$ ou $\rho_k = -1$ suggèrent une relation fonctionnelle linéaire forte entre le modèle considéré et la fuite. La valeur ρ_k^2 mesure la proximité des n points à la droite de régression dont le signe de la pente est donné par le signe de ρ_k. La bonne clé sera donc estimée comme étant la clé pour laquelle on aura, en valeur absolue, le coefficient de Pearson le plus élevé.

$$\hat{k^\star} = \underset{k\in\mathscr{K}}{argmax}(|\rho_k|).$$

La *CPA* (Brier et al. (2004) - pour Correlation Power Analysis) est actuellement l'attaque par observations la plus populaire. Cette popularité est principalement due au fait que les modèles utilisés le plus couramment sont souvent effectivement proches de la fuite à une fonction linéaire près. Comme montré dans l'article Le et al. (2006), l'utilisation de la CPA avec un modèle monobit revient à la DPA de Kocher. Elle est également très résistante au bruit présent dans les réalisations de la variable $L(Z)$ que l'on utilise. En effet, le bruit aura tendance à être moyenné par le calcul du coefficient de Pearson. Ainsi, si l'on augmente le nombre de réalisations dont on dispose, on va diminuer de plus en plus la contribution du bruit. Néanmoins, afin de prévenir cette attaque, les fabricants de cartes à puce essayent de mettre en œuvre des contre-mesures visant à casser cette linéarité. Il est donc important pour un attaquant d'avoir à sa disposition d'autres tests statistiques lui permettant d'identifier des relations entre $m(X,k)$ et $L(Z)$ même lorsque ces relations ne sont pas linéaires.

2.3.2. Corrélation des rangs de Spearman

Outre le fait que la relation entre la fuite et le modèle correspondant à la bonne clé peut exister sans être linéaire, il est important de noter la nature des variables aléatoires considérées. $L(Z)$ est continue alors que $m(X,k)$ est discrète. Bien sûr, $m(X,k)$ est discrète car le modèle que l'on considère n'est pas assez précis. Néanmoins, d'une manière générale, le coefficient de corrélation

 ISSN: 2102-6238

linéaire n'est guère recommandé pour comparer des variables de natures différentes. L'utilisation de la corrélation des rangs semble être une bonne alternative car elle permet d'étudier les relations entre une variable discrète et une variable continue, mais également de détecter non seulement les relations linéaires mais aussi les relations d'ordre. Le coefficient de corrélation des rangs ou *coefficient de Spearman*, est calculé en remplaçant dans l'Équation (2.3.1) l_i et $m(x_i,k)$ par leur rang. Le rang d'une observation est donné par sa position lorsque l'on a trié par ordre croissant toutes les observations. Si l'on considère les données $(3,7),(2,9),(7,10)$ les rangs associés seront : $(2,1),(1,2),(3,3)$. Lorsque plusieurs observations ont la même valeur, on remplace cette valeur par leur rang moyen. Le *coefficient de Spearman* s'interprète de la même façon que le coefficient de Pearson. Une proximité de 1 traduit une relation positive (variation dans le même sens) entre les deux variables considérées, alors qu'une proximité de -1 traduit une relation négative (variation en sens inverse). Enfin une proximité de 0 implique une indépendance entre les variables.

La bonne clé sera donc estimée, comme précédemment, comme étant la clé pour laquelle le *coefficient de Spearman* (Batina et al., 2008) est le plus élevé en valeur absolue.

$$\hat{k^\star} = \underset{k\in\mathscr{K}}{argmax}(|r_k|).$$

La corrélation des rangs ne repose pas sur l'hypothèse d'une relation linéaire entre les données mais simplement sur une relation d'ordre. De plus, elle permet de comparer deux variables aléatoires qui ne sont pas de même nature. Malheureusement, en présence de nombreux ex æquo, la corrélation des rangs n'est pas toujours très adaptée. Or, dans les attaques par observations, que ce soit pour la variable représentant le modèle ou pour la variable représentant la fuite, les ex æquo sont très nombreux.

2.3.3. Tau de Kendall

Le *tau de Kendall* (Kendall, 1938) permet de déceler des relations d'ordre entre deux variables aléatoires, y compris de nature différente. Le tau de Kendall est la différence estimée entre la probabilité que les données observées aient le même rang pour les deux variables et la probabilité que les données observées aient des rangs différents. Le principe de ce tau est différent des précédents. On commence par trier une des deux séries d'observations et les valeurs des rangs de la seconde série sont mises en regard de la première. Une fois la première série triée, on ne s'intéresse plus qu'à la seconde. On remplace la valeur de chaque observation par le nombre de valeurs suivantes qui lui sont supérieures auquel on retranche le nombre de valeurs suivantes qui lui sont inférieures.

Si l'on considère par exemple les couples suivant $(9,4),(0,3),(1,1),(2,2)$. On commence par ordonner ces couples suivants la première série et l'on obtient $(0,3),(1,1),(2,2),(9,4)$. Maintenant, on ne regarde plus que la seconde série. On attribue à 3 le score de $1-2=-1$, à 1 celui de $2-0=2$, à 2 celui de $1-0=1$ et enfin à 4 le score de $0-0=0$. On obtient ainsi une nouvelle série $-1,2,1,0$. La somme S des valeurs de cette série nous donne des indications sur les relations entre les deux variables considérées au départ. En effet, si $S=n(n-1)/2$, S sera la somme des n premiers entiers naturels. On en déduit que l'ordre est totalement respecté. De

 ISSN: 2102-6238

même si $S = -n(n-1)/2$, l'ordre est parfaitement inversé. Si en revanche S est nulle, on aura indépendance entre les deux variables étudiées.

Partant de S, on définit le *tau de Kendall* de la manière suivante :

$$\tau = \frac{2S}{n(n-1)}.$$

Le *tau de Kendall* est compris entre -1 et 1 et s'interprète de la même manière que le coefficient des rangs ou celui de Pearson ; plus il est proche de 1, plus on est certain qu'il existe une corrélation positive entre nos données. Plus il est proche de -1, plus on peut supposer qu'il existe une corrélation négative. Lorsque le *tau de Kendall* est proche de zéro, l'existence d'une liaison monotone entre nos données sera peu probable.

La clé secrète utilisée par le composant sera donc estimée par (Batina et al., 2008) :

$$\hat{k^\star} = \underset{k \in \mathscr{K}}{argmax}(|\tau_k|).$$

Le *tau de Kendall* permet de détecter certaines relations qui ne sont pas découvertes par le *coefficient de Spearman*, néanmoins, lorsque l'on est en présence de nombreux ex æquo, il est recommandé d'utiliser le coefficient gamma.

2.3.4. Coefficient gamma

Le *coefficient gamma* (Batina et al., 2008), comme le *tau de Kendall*, permet de détecter des relations ordinales entre deux variables. Il se calcule en étudiant la concordance ou la discordance entre deux paires. On dit qu'une paire, (U_1, V_1) et (U_2, V_2), est concordante lorsque $U_2 - U_1$ et $V_2 - V_1$ sont de même signe. Respectivement, on dira qu'une paire et discordante lorsque les deux différences n'ont pas le même signe. Si les deux différences sont nulles, la paire est considérée comme concordante. Si l'une est nulle mais pas l'autre, la paire n'est ni concordante ni discordante. On note N_c le nombre de paires concordantes dans nos observations et N_d le nombre de paires discordantes. Il est intéressant de noter que, si certaines paires ne sont ni concordantes ni discordantes, $n \neq N_c + N_d$.

Le coefficient gamma est calculé de la manière suivante :

$$\Gamma = \frac{N_c - N_d}{N_c + N_d}.$$

Lorsque le coefficient gamma sera proche de 0, on considérera que les deux variables sont indépendantes. En revanche, plus la valeur de ce coefficient sera proche de 1 ou de -1 plus la relation entre les variables sera forte. La clé secrète sera donc estimée par :

$$\hat{k^\star} = \underset{k \in \mathscr{K}}{argmax}(|\Gamma_k|).$$

Le *coefficient de Spearman*, le *tau de Kendall* et le *coefficient gamma* permettent de détecter des relations ordinales entre les variables. Néanmoins, ils ne permettent pas de détecter des relations fonctionnelles qui ne sont pas des relations d'ordre (par exemple : une relation elliptique, sinusoïdale ou encore des combinaisons de relations, ...). Dans la Section 2.4, nous allons présenter d'autres méthodes permettant de détecter de nombreuses relations y compris non ordinales.

 ISSN: 2102-6238

2.4. *Méthodes basées sur des lois de probabilité*

2.4.1. *Mutual Information Analysis (MIA) : mesure des dépendances existantes entre deux variables grâce à l'*information mutuelle

Avant de présenter l'*information mutuelle* (Gierlichs et al., 2008), il est nécessaire de faire quelques rappels concernant la théorie de l'information.

L'entropie de Shannon est une manière de mesurer la quantité d'information contenue dans une variable aléatoire. On note $\mathbb{P}[E]$ la probabilité de l'événement E. L'entropie d'une variable aléatoire discrète U dans l'espace $\mathscr{U}$ est définie par :

$$H(U) = -\sum_{u\in\mathscr{U}} \mathbb{P}[U=u] log_2(\mathbb{P}[U=u]).$$

L'entropie d'une variable aléatoire continue est définie de manière similaire. L'entropie est maximale lorsque la loi de la variable est uniforme. L'*information mutuelle* entre deux variables aléatoires U et V est définie de la manière suivante :

$$I(U;V) = H(U) + H(V) - H(U,V).$$

Dans le cadre des attaques par canaux cachés, on cherchera à estimer l'information mutuelle entre la variable aléatoire continue $m(X,k)$ et la variable continue $L(Z)$. Afin d'alléger les notations, nous noterons $L(Z) = L$ et $m(X,k) = M_k$. L'information mutuelle entre nos deux variables sera définie par :

$$I_k(L;M_k) = \sum_{m\in\mathscr{M}} \mathbb{P}[M_k=m] \int_{l\in\mathscr{L}} f_{L|M_k=m}(l) log_2\left(\frac{f_{L|M_k=m}(l)}{f_L(l)}\right) dl.$$

où f_L est la densité de probabilité de $L(Z)$ et $f_{L|M_k=m}$ est la densité de probabilité conditionnelle de $L(Z)$ sachant $m(X,k) = m$. La quantité $I_k(L;M_k)$ est toujours positive ou nulle. Si $I_k(L;M_k)$ est nulle, $L(Z)$ et $m(X,k)$ sont indépendantes. La valeur de l'*information mutuelle* est toujours inférieure à l'entropie de chacune des variables étudiées avec égalité si, et seulement si, l'une des variables est une fonction déterministe de l'autre. Plus l'information mutuelle sera forte, plus $L(Z)$ et $m(X,k)$ seront liées.

En pratique, pour une attaque par observations, on cherchera $k \in \mathscr{K}$ qui maximisera l'information mutuelle $I_k(L;M_k)$. La clé secrète sera donc estimée par :

$$\hat{k^\star} = \underset{k\in\mathscr{K}}{argmax}(I_k(L;M_k)).$$

Il est en général difficile de calculer l'*information mutuelle* entre $L(Z)$ et $m(X,k)$ car la loi de la variable $L(Z)$ n'est pas connue. On peut aussi remarquer que :

$$\int_{l\in\mathscr{L}} f_{L|M_k=m}(l) log_2\left(\frac{f_{L|M_k=m}(l)}{f_L(l)}\right) dl = E_{L|M_k=m}\left[log_2\left(\frac{f_{L|M_k=m}(L)}{f_L(L)}\right)\right].$$

où $E_{L|M_k=m}$ signifie que l'espérance est calculée sous la loi de $L(Z)$ sachant que $m(X,k) = m$. Cette espérance peut être estimée modulo une constante de renormalisation, par :

$$\sum_{l_i\in L(Z);m(x_i,k)=m} log_2\left(\frac{\hat{f}_{L|M_k=m}(l_i)}{\hat{f}_L(l_i)}\right).$$

 ISSN: 2102-6238

où $\hat{f}_{L|M_k=m}$ et $\hat{f}_L$ sont des estimateurs de $f_{L|M_k=m}$ et f_L.

Le choix des estimateurs $\hat{f}_{L|M_k=m}$ et $\hat{f}_L$ est crucial. Dans la suite, nous allons présenter les estimateurs les plus utilisés dans les attaques par canaux cachés. Dans le cadre des attaques par observations, on utilise le plus souvent des approches non paramétriques. Ainsi les estimateurs les plus utilisés sont les estimateurs à base de noyaux. Afin de présenter cette méthode, on se place dans un contexte simple où l'on a accès à n observations d'une variable aléatoire continue U. On notera $u_i, i \in \{1, \cdots n\}$ ces observations. Un estimateur de la densité de probabilité associée à U est donné par :

$$\hat{f}_U(u) = \frac{1}{nh}\sum_i^n K(\frac{u-u_i}{h}).$$

où K est un noyau et h une fenêtre qui doit être choisie avec soin. La valeur optimale de h est celle qui minimise l'AMISE (Asymptotic Mean Integrated Squared Error) qui dépend de la variable aléatoire dont on souhaite estimer la densité de probabilité. Le noyau, quant à lui, est une fonction réelle telle que $\int_{-\infty}^{\infty} K(u)du = 1$ et $K(u) = K(-u)$. Il existe de nombreux noyaux. Le choix du noyau dépend essentiellement de la forme de la variable dont on souhaite estimer la densité de probabilité. Dans le cadre d'une attaque par observations, on utilise très souvent le noyau gaussien (Batina et al., 2011) défini par :

$$K(u) = \frac{1}{\sqrt{2\pi}} exp(-\frac{1}{2}u^2).$$

Dans Veyrat-Charvillon and cois Xavier Standaert (2009) les auteurs proposent d'utiliser comme largeur de fenêtre :

$$h = 1.06 \ \ min\left(\hat{\sigma}, \frac{IQR}{1.34}\right) \ \ n^{-\frac{1}{5}},$$

où $\hat{\sigma}(U)$ est l'écart type empirique et $IQR(U)$ représente l'écart interquartile empirique. [3] Un autre estimateur très populaire dans le cadre des attaques par observations est basé sur les histogrammes. Le noyau permettant d'obtenir cet estimateur est donné par :

$$K(u) = \mathbb{1}_{]-\frac{1}{2};\frac{1}{2}[}(u)$$

où $\mathbb{1}$ est la fonction indicatrice. [4] Dans Gierlichs et al. (2008), les auteurs proposent d'utiliser simplement un histogramme dont le nombre de classes sera égal au nombre de valeurs différentes possibles pour $m(X,k)$. D'autres estimateurs ont été proposés, comme dans Venelli (2010) où l'auteur a étudié les résultats obtenus grâce aux B-splines. L'auteur de Venelli (2010) montre que l'utilisation de B-splines permet d'avoir une bonne estimation des densités de probabilité y compris lorsque les données sont bruitées.

L'*information mutuelle* permet d'obtenir des attaques extrêmement puissantes car aucune supposition sur la relation entre $m(X,k^\star)$ et $L(Z)$ n'est nécessaire. Néanmoins, en pratique, ces attaques sont difficiles à mettre en ?uvre car il faut bien choisir les estimateurs utilisés ainsi que la taille de la fenêtre. La capacité à bien estimer une densité de probabilité est un facteur très

[3] Si l'on note Q_2 la médiane de nos observations, Q_1 la médiane de nos observations inférieures à Q_2 et Q_3 la médiane de nos observations supérieures à Q_2. L'écart interquartile empirique est donné par : $IQR = Q_3 - Q_1$

[4] $\mathbb{1}_E(u) = 1$ si $u \in E$ et 0 sinon .

 ISSN: 2102-6238

important dans la réussite de telles attaques. Dans le cadre des attaques par observations, les réalisations de la variable $L(Z)$ dont on dispose sont obtenues à partir de mesures physiques et peuvent être fortement bruitées. Or l'information mutuelle n'est pas une méthode très résistante au bruit. En effet, lorsque les réalisations de la variable $L(Z)$ que l'on considère sont très bruitées, la variance de la loi $L(Z)$ va augmenter, alors que celle de la loi $m(X,k^\star)$ va rester la même. Ainsi, contrairement à la corrélation linéaire, l'information mutuelle ne moyenne pas le bruit mais y est sensible. Au sens de la théorie de l'information, l'information mutuelle est la meilleure méthode possible (Prouff and Rivain, 2010), mais en pratique, les données dont on dispose ne se prêtent pas très bien à cette méthode en particulier, car les réalisations de $L(Z)$ sont très bruitées.

2.4.2. Statistique de Kolmogorov-Smirnov

En considérant l'*information mutuelle*, on peut s'apercevoir qu'elle peut s'interpréter comme la moyenne (sur le modèle $m(X,k)$) des distances de Kullback-Leibler [5] entre la distribution de $L(Z)$ et la distribution de $L(Z)$ sachant $m(X,k)=m$. L'idée de l'approche de Kolmogorov-Smirnov est de remplacer cette distance par une distance plus simple (L_∞) et d'utiliser des fonctions de répartition en lieu et place des estimateurs de densité de probabilité :

$$D_k^{KS} = \sum_{m \in \mathscr{M}} \mathbb{P}[M_k = m] \sup_l |F_{L|m_k=m}(l) - F_L(l)| \tag{2}$$

où F_L est la fonction de répartition associée $L(Z)$ et $F_{L|m_k=m}$ celle associée à $L(Z)$ sachant que $m(X,k)=m$. Finalement, la statistique de Kolmogorov - Smirnov, $D_k^{\hat{KS}}$, est obtenue en estimant les fonctions de répartition par leur version empirique dans l'Équation (2). Lorsque les deux variables que l'on observe sont indépendantes, la valeur de la statistique de Kolmogorov - Smirnov est proche de 0. Ainsi, dans le cas où k est la bonne clé, on s'attend à ce que cette distance soit grande. Ainsi, la clé secrète sera estimée grâce à (Whitnall et al., 2011) :

$$\hat{k^\star} = \underset{k \in \mathscr{K}}{argmax}(D_k^{\hat{KS}}).$$

La *statistique de Kolmogorov - Smirnov* ne nécessite pas l'estimation de densités de probabilité puisque l'on utilise la fonction de répartition empirique. Cette fonction est en fait une manière d'estimer la fonction de répartition en utilisant un histogramme avec un choix particulier de fenêtre. Il est intéressant de noter que l'on peut tout à fait substituer à la distance L_∞ une autre distance comme, par exemple, la distance L_1 ou la distance du χ^2. Ainsi, en utilisant le carré de la distance euclidienne, on obtient le critère de Cramér - von Mises. Dans Veyrat-Charvillon and cois Xavier Standaert (2009), les auteurs ont montré que les résultats obtenus grâce au critère de Cramér - von Mises sont très proches de ceux obtenus avec la statistique de Kolmogorov - Smirnov.

2.4.3. MICA : Maximal Information Coefficient Analysis

L'idée du coefficient maximal d'information (MIC) est de calculer l'information mutuelle grâce à différents histogrammes en s'autorisant à considérer des histogrammes ayant des classes de tailles

[5] $d(f_1,f_2) = \int_{-\infty}^{\infty} f_1(x) ln\left(\frac{f_1(x)}{f_2(x)}\right) dx$

 ISSN: 2102-6238

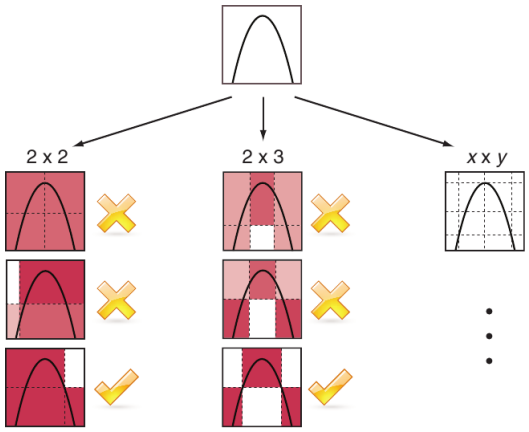

FIGURE 4. *Figure extraite de l'article Reshef et al. (2011).*

	[-1,0[	[0,1[
[0,2[	0	2
[2,4[	1	1
[4,6[	1	1

TABLEAU 1. *Distribution de D sur G pour l'exemple proposé.*

différentes. Dans la Figure 4 les auteurs de Reshef et al. (2011) cherchent à maximiser la valeur de *l'information mutuelle* pour un nombre de classes fixé. On constate que la façon dont sont choisis les histogrammes pour estimer l'*information mutuelle* peut jouer un grand rôle.

Le but du *MIC* est de détecter s'il existe une relation entre deux variables aléatoires U et V. Pour cela on dispose de l'observation d'un échantillon de taille n de variable parente U (resp. V), $\{u_1,\cdots u_n\}$(resp. $\{v_1,\cdots v_n\}$), de U (resp. V). On notera D l'ensemble des paires $(u_i,v_i),\forall i \in \{0,\cdots,n\}$ ordonnées par ordre croissant pour U puis en cas d'égalité pour V. Les auteurs de Reshef et al. (2011) appellent *p-par-q* grille la partition des couples $(u_i,v_i),\forall i \in \{0,\cdots,n\}$ avec un histogramme à p classes pour la première composante et q classes pour la seconde. On notera par $\tilde{D}^G$ la distribution empirique du couple (U,V) associée à la grille G (comme dans le Tableau 1). On notera $\mathscr{G}_{(p,q)}$ l'ensemble de toutes les grilles de taille *p-par-q*. Si l'on considère :

$$U = (4,0,1,3,5,2),$$

$$V = (-0.3,0.3,0.01,-0.5,0.4,0.1),$$

$$D = ((0,0.3),(1,0.01),(2,0.1),(3,-0.5),(4,-0.3),(5,0.4)),$$

et

$$G = ([0,2[,[2,4[,[4,6[) \times ([-1,0[,[0,1[),$$

une grille de taille 3-*par*-2.

La distribution empirique de (U,V) associée à cette grille est donnée dans le Tableau 1. On notera $\hat{I}(\tilde{D}^G)$ l'information mutuelle obtenue grâce aux distributions empiriques associées à la

 ISSN: 2102-6238

grille G. Pour p et q fixés, le maximum d'information mutuelle sur l'ensemble des p-par-q grilles est défini par :

$$I^*(D,p,q) = \max_{G\in\mathscr{G}_{(p,q)}} (\hat{I}(\tilde{D}^G)). \tag{3}$$

Il n'est pas possible de comparer directement deux maxima d'information mutuelle, $I^*(D,p,q)$ et $I^*(D,p',q')$, lorsque la taille des grilles n'est pas la même. En effet, sur une grille de taille p-par-q la valeur maximale pouvant être atteinte est $log_2(min(p,q))$. Pour pallier ce problème, les auteurs proposent de normaliser ces maxima avant de les comparer. $I^*(D,p,q)$ est borné par $log_2(min(p,q))$. Une normalisation naturelle des maxima d'information mutuelle est donc donnée par :

$$M(D)_{p,q} = \frac{I^*(D,p,q)}{log_2(min(p,q))}. \tag{4}$$

Il est maintenant possible de comparer deux maxima d'information mutuelle pour des grilles de tailles différentes. On peut donc calculer le maximum des $I^*(D,p,q)$ pour tous les p et q possibles.

Les valeurs p et q sont bornées par le nombre d'observations, n, que l'on possède. Le nombre de grilles possibles est donc borné par n^n. Ce nombre devient très vite trop important pour que l'on puisse calculer l'information mutuelle de l'ensemble de ces grilles. Après une étude empirique, les auteurs ont proposé de borner le nombre de grilles à $n^{0.6}$ et finalement défini le coefficient maximal d'information par :

$$MIC(D) = \max_{\forall p,q|pq\leq n^{0.6}} (M(D)_{p,q}).$$

La valeur de ce coefficient est comprise entre 0 et 1. Un *MIC* de 0 indique une très probable indépendance entre les deux variables considérées. A contrario, un *MIC* de 1 indique une relation forte entre les deux variables.

Lorsque l'on compare le *MIC* à l'*information mutuelle*, on remarque tout d'abord une complexité de calcul bien plus importante pour le *MIC* que pour l'*information mutuelle*. En effet pour obtenir le *MIC*, on doit calculer de multiples fois l'*information mutuelle* à partir de différents histogrammes. Néanmoins, le *MIC* peut être calculé sans connaissances préalables sur les variables considérées. Nous avons effectué des simulations dans lesquelles nous avons fait varier l'influence de chaque bit sur la fuite du composant. Nous avons alors constaté un excellent comportement du MIC. Le MIC semble donc offrir une alternative intéressante dans le cas où il est difficile à l'attaquant de modéliser la fuite.

De plus, le *MIC* reste plus stable en présence de bruit que l'*information mutuelle*. Dans le cadre des attaques par observations, le bruit est un problème important, le *MIC* offre donc une alternative très intéressante à l'*information mutuelle*.

3. Applications numériques

Dans cette section nous allons présenter les résultats que nous avons obtenus en appliquant les différentes méthodes introduites dans les paragraphes précédents. Pour cela, nous avons utilisé 1000 courbes de consommation de courant en fonction du temps, proposées dans la version 1 de DPAContest (ParisTech, 2008). Nous avons choisi de prendre comme fonction g la sortie du

 ISSN: 2102-6238

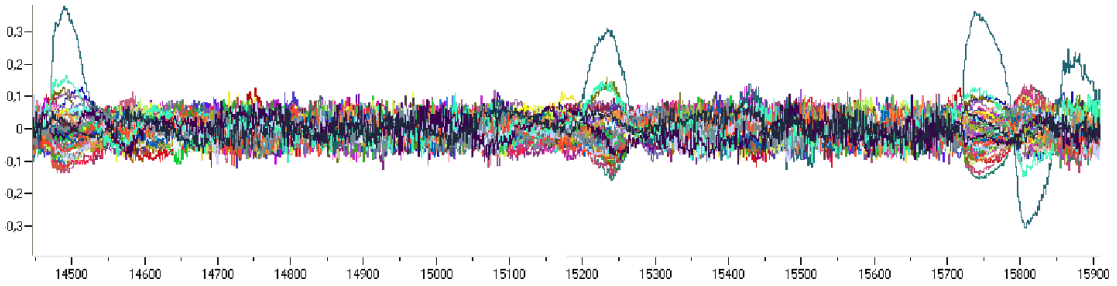

FIGURE 5. *En abscisse le temps en milliseconde, en ordonnée la valeur de la* corrélation linéaire.

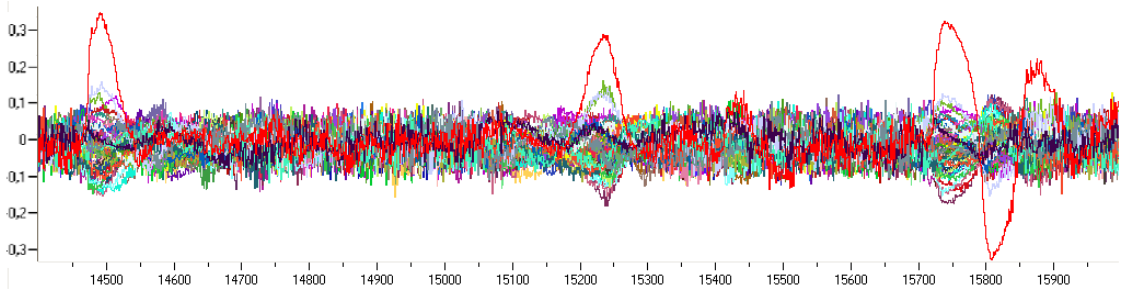

FIGURE 6. *En abscisse le temps en milliseconde, en ordonnée la valeur de la* corrélation des rangs.

$XorRL$ de la dernière ronde de l'algorithme DES (of STANDARDS, 1977). Il y a alors 64 valeurs possibles pour la clé secrète, $k^\star$. Nous avons donc besoin de comparer les résultats des différentes méthodes pour 64 modèles différents. Les Figures 5 à 11 représentent les résultats que nous avons obtenus. Nous présentons ici uniquement la dernière des seize rondes du DES car c'est celle durant laquelle le composant va calculer la fonction g que nous avons choisie. Chaque courbe représente le résultat obtenu pour un modèle. On constate que pour chaque méthode, une des courbes se détache. Cette courbe représente la clé secrète utilisée par le composant.

La Figure 5 représente la valeur du coefficient de Pearson. La puissance du coefficient semble faible même lorsque l'on considère le modèle correspondant à la bonne clé. Néanmoins, on constate que la différence entre les valeurs du coefficient de Pearson pour la bonne clé et pour les autres clés est suffisamment significative pour pouvoir conclure.

Sur les Figures 6, 7 et 8 nous avons présenté les résultats des autres méthodes basées sur les rangs. Ces résultats sont très proches de ceux obtenus grâce à la corrélation linéaire, car la relation liant le modèle correspondant à la clé secrète et les données considérées est plutôt linéaire. À certains instants, la relation entre le modèle correspondant à la clé secrète et les données est positive, à d'autres elle est négative. De plus, on peut observer qu'aux instants où le coefficient de Pearson est maximal pour le modèle correspondant à la bonne clé, d'autres modèles semblent reliés de manière linéaire aux données considérées, même si cette relation est moins forte. On appelle ce phénomène *pic harmonique* et il est dû à la fonction cryptographique que l'on a choisie.

La Figure 9 présente les résultats obtenus grâce à l'information mutuelle que l'on a estimée avec un noyau gaussien en utilisant la fenêtre proposée dans Veyrat-Charvillon and cois Xavier Stan-

 ISSN: 2102-6238

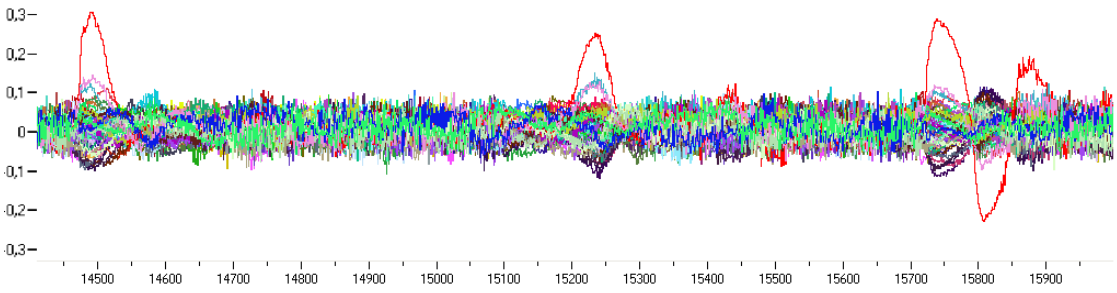

FIGURE 7. *En abscisse le temps en milliseconde, en ordonnée la valeur du* tau de Kendall.

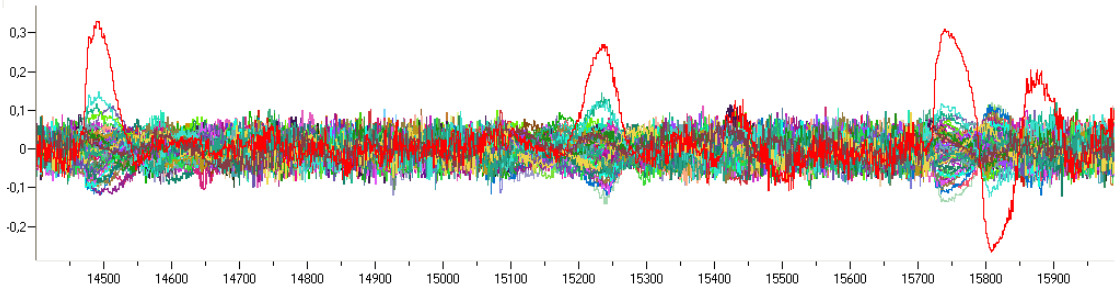

FIGURE 8. *En abscisse le temps en milliseconde, en ordonnée la valeur du* coefficient gamma.

daert (2009). Ici, les résultats issus du modèle correspondant à la bonne clé se détachent davantage. Néanmoins, cette méthode a été bien plus compliquée à mettre en ?uvre que les précédentes et un mauvais choix de noyaux et/ou de largeur de fenêtres entraîne l'absence de résultats concluants.

La Figure 10 présente les résultats obtenus en utilisant la statistique de Kolmogorov-Smirnov. On observe, comme pour les méthodes basées sur la corrélation, le phénomène de *pic harmonique* même s'il est bien moins prononcé. Les résultats que nous présentons pour l'*information mutuelle* ont été obtenus grâce à un noyau gaussien. Même si l'échelle n'est pas identique, le modèle correspondant à la clé secrète ressort moins que pour l'information mutuelle.

Enfin la Figure 11 présente les résultats issus du *MIC*. Comme pour *l'information mutuelle*, on n'observe plus le phénomène de *pic harmonique*. La valeur du *MIC* lorsque le modèle est erroné est plus importante que pour *l'information mutuelle*. Le *MIC* propose donc des résultats très similaire à *l'information mutuelle* mais ne nécessite pas de choisir une taille de fenêtre ou un noyau particulier. De plus, le *MIC* est moins sensible au bruit introduit lors des acquisition que la *statistique de Kolmogorov-Smirnov* ou que *l'information mutuelle* (Whitnall and Oswald, 2011). Dans le cadre d'une attaque par observations, le MIC est donc une alternative intéressante pour un attaquant même si le temps de calcul est bien plus important.

Le Tableau 2 présente le temps nécessaire pour obtenir les graphiques de cette section en seconde. Les calculs ont été effectués sur un ordinateur équipé d'un processeur XEON X3430 possédant 4 c ?urs cadencés à 2.4 GHz.

 ISSN: 2102-6238

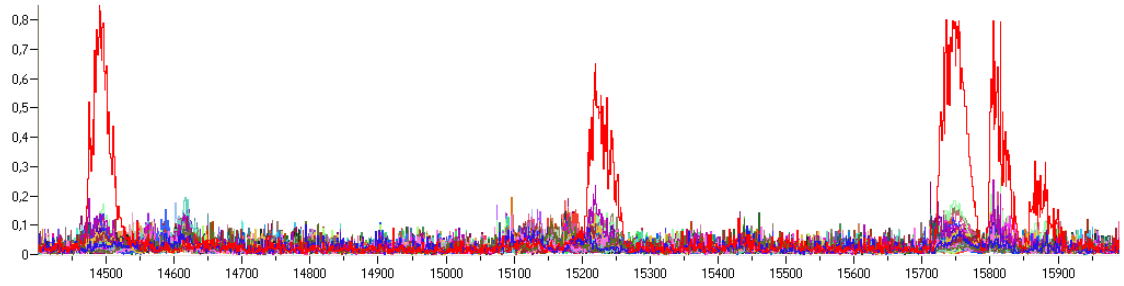

FIGURE 9. *En abscisse le temps en milliseconde, en ordonnée la valeur de* l'information mutuelle *obtenue grâce à un noyau gaussien et la fenêtre proposée dans Veyrat-Charvillon and cois Xavier Standaert (2009).*

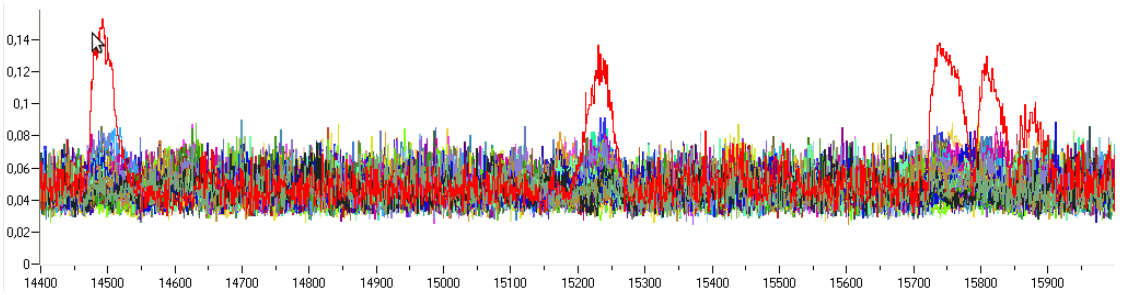

FIGURE 10. *En abscisse le temps en milliseconde, en ordonnée la valeur de la statistique de* Kolmogorov-Smirnov.

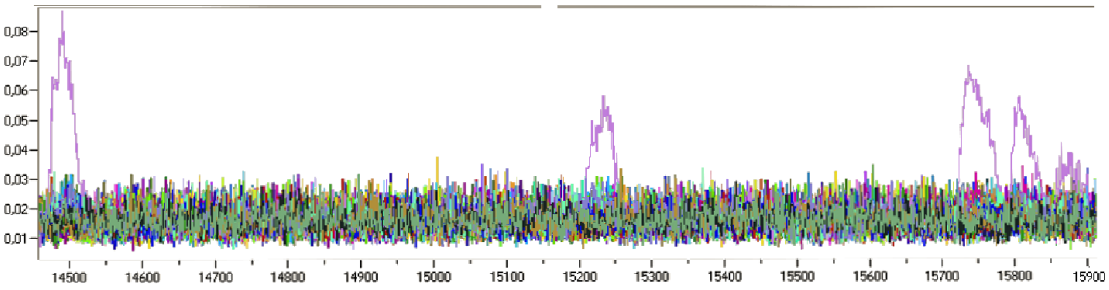

FIGURE 11. *En abscisse le temps en milliseconde, en ordonnée la valeur du* MIC.

Attaque	Temps
Corrélation linéaire	20
Corrélation des rangs	190
Tau de Kendall	270
Coefficient Gamma	30
Information mutuelle	230
Statistique de Kolmogorov-Smirnov	130
Coefficient maximal d'information	30700

TABLEAU 2. *Temps requis pour chaque attaque en seconde.*

 ISSN: 2102-6238

4. Conclusion

Nous avons présenté dans cet article les attaques par observations. Ces attaques sont aujourd'hui couramment utilisées en cryptanalyse. Elles présentent deux difficultés majeures, l'acquisition de la fuite par l'attaquant mais aussi le traitement des données obtenues. Nous avons exposé ici les différentes méthodes statistiques fréquemment utilisées pour traiter ces données. D'une manière générale, il est important d'utiliser différentes méthodes et de ne pas conclure à l'impossibilité d'une attaque après l'échec d'un seul test. En effet, chacune de ces méthodes présentent des avantages et des inconvénients. A l'origine, Kocher a proposé d'utiliser la différence des moyennes. Le modèle qu'il proposait alors était très simple. Le principal atout de cette méthode vient de sa simplicité. Il est effet très facile de calculer la différence des moyennes. En revanche, du fait même de sa simplicité, cette méthode est souvent peu efficace sur des composants cryptographiques modernes. Le coefficient de Pearson est lui aussi facile à calculer et offre la possibilité d'utiliser des modèles de fuites plus complexes et ainsi de mieux représenter la fuite attendue. Lorsque le modèle de fuite est bien choisi, le coefficient de Pearson donne d'excellents résultats dans le cadre des attaques par observations. Malheureusement, il ne permet pas de détecter des relations non linéaires entre deux variables.

Le *coefficient de Spearman*, le *tau de Kendall* et le *coefficient gamma* détectent quant à eux des relations d'ordre. Bien sûr, cette plus grande capacité de détection entraîne une plus grande complexité de calcul mais autorise aussi l'attaquant à être moins précis dans le choix de son modèle de fuite. Dans le contexte des attaques par canaux cachés, il est très courant d'avoir un grand nombre d'ex-aequo il est donc recommandé d'utiliser plutôt le *tau de Kendall* et le *coefficient gamma*. Ces méthodes reposent sur la structure de corrélation qui peut exister entre les deux variables étudiées.

Les méthodes qui sont basées sur les lois de probabilités sous-jacentes aux variables considérées sont plus compliquées à mettre en ?uvre mais peuvent se révéler très efficaces car elles permettent de détecter des relations sans a priori sur leurs formes. L'*information mutuelle* est la première de ces méthodes à avoir été utilisée dans le cadre des attaques par observations. Néanmoins, même si elle peut se montrer très efficace, elle est difficile à mettre en pratique. L'attaquant doit se montrer très attentif au choix fait pour estimer les différentes densités de probabilité utilisées, le bruit pouvant être un facteur qui perturbe grandement l'estimation. Le *test de Kolmogorov-Smirnov* permet, en changeant la distance utilisée, d'avoir une très large famille de test permettant de détecter des relations entre les variables considérées. Ces tests sont plus faciles à mettre en ?uvre que l'information mutuelle car ils reposent sur des fonctions de répartition estimées de manière empirique. Il n'y a donc pas de choix d'estimateur à faire. Enfin, le *MIC* semble être une alternative intéressante à l'*information mutuelle* puisque qu'il est moins sensible au bruit et demande moins de connaissance préalable sur les données à l'attaquant pour réaliser son attaque. Le principal défaut du *MIC* est la complexité des calculs mis en oeuvre.

Le coefficient de Pearson reste aujourd'hui le principal test utilisé dans les attaques par observations car le choix du modèle de fuite, fait par l'attaquant, est souvent judicieux. En particulier en règle générale, la relation entre le modèle correspondant à la clé secrète et les observations est linéaire. Néanmoins si ce modèle de fuite est mal choisi, le coefficient de Pearson peut être mis en défaut. Un attaquant ne devrait donc pas se restreindre à un seul type de tests même si certains sont bien plus complexes que le coefficient de Pearson et le MIC est un bon

 ISSN: 2102-6238

choix tout en restant dans une approche générale.

Références

Batina, L., Gierlichs, B., and Lemke-Rust, K. (2008). Comparative Evaluation of Rank Correlation Based DPA on an AES Prototype Chip. In *ISC*, pages 341–354.

Batina, L., Gierlichs, B., Prouff, E., Rivain, M., Standaert, F.-X., and Veyrat-Charvillon, N. (2011). Mutual Information Analysis : a Comprehensive Study. *J. Cryptology*, 24 :269–291.

Brier, E., Clavier, C., and Olivier, F. (2004). Correlation Power Analysis with a Leakage Model. In *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.

Clavier, C. and Joye, M. (2001). Universal Exponentiation Algorithm. In *Cryptographic Hardware and Embedded Systems - CHES 2001, Third International Workshop, Paris, France, May 14-16, 2001, Proceedings*, volume 2162 of *Lecture Notes in Computer Science*, pages 300–308. Springer.

Diffie, W. and Hellman, M. (1976). New Directions in Cryptography. Information Theory, IEEE Transactions on 22(6) : 644-654.

Dubertret, G. (1998). *Initiation à la Cryptographie*. Vuibert supérieur.

Dumas, J.-G., Roch, J.-L., Tannier, É., and Varrette, S. (2007). *Théorie des Codes-Compression, Cryptage, Correction : Compression, Cryptage, Correction*. Dunod.

Ferguson, N. and Schneier, B. (2003). *Practical Cryptography*, volume 141. Wiley New York.

Gamal, T. E. (1984). A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. In *Advances in Cryptology, Proceedings of CRYPTO '84, Santa Barbara, California, USA, August 19-22, 1984, Proceedings*, volume 196 of *Lecture Notes in Computer Science*, pages 10–18. Springer.

Gierlichs, B., Batina, L., Tuyls, P., and Preneel, B. (2008). Mutual Information Analysis. In *Cryptographic Hardware and Embedded Systems - CHES 2008, 10th International Workshop, Washington, D.C., USA, August 10-13, 2008. Proceedings*, volume 5154 of *Lecture Notes in Computer Science*, pages 426–442. Springer.

Kaihara, M., Kleinjung, T., Lenstra, A., and Montgomery, P. (2009). On the Security of 1024-bit RSA and 160-bit Elliptic Curve Cryptography : version 2.1. *cryptology eprint archive*, report 2009/389,.

Kendall, M. (1938). A New Measure of Rank Correlation. *Biometrika*, 30 (1/2) :81–93.

Kerckhoffs, A. (1883a). La cryptographie militaire. *Journal des sciences militaires*, IX janvier :5–38.

Kerckhoffs, A. (1883b). La cryptographie militaire. *Journal des sciences militaires*, IX fÃ©vrier :161–191.

Knuth, D. (1981). *The Art of Computer Programming*, volume 2, chapter Seminumerical Algorithms. Addison-Wesley, Reading, Massachusetts.

Kocher, P., Jaffe, J., and Jun, B. (1999). Differential power analysis. In *Advances in Cryptology-CRYPTO '99*, pages 388–397. Springer.

Le, T.-H., Clédière, J., Canovas, C., Robisson, B., Servière, C., and Lacoume, J.-L. (2006). A Proposition for Correlation Power Analysis Enhancementth. In *Cryptographic Hardware and Embedded Systems - CHES 2006, 8th International Workshop*, volume 4249 of *Lecture Notes in Computer Science*, pages 174–186. Springer.

Mangard, S., Oswald, E., and Popp, T. (2007). *Power analysis attacks : Revealing the secrets of smart cards*, volume 31. Springer.

Menezes, A. J., Van Oorschot, P. C., and Vanstone, S. A. (2010). *Handbook of applied cryptography*. CRC press.

of STANDARDS, U. S. N. B. (1977). The data encryption standard. fips pub 46. Technical report, National Bureau of Standards, Washington, DC, USA.

ParisTech, T. (2008). Dpa contest v1. http ://www.dpacontest.org/index.php.

ParisTech, T. (2010). Dpa contest v2. http ://www.dpacontest.org/v2/index.php.

Prouff, E. and Rivain, M. (2010). Theoretical and Practical Aspects of Mutual Information-Based Side Channel Analysis. *International Journal of Applied Cryptography*, 2(2) :121–138.

Reshef, D., Reshef, Y., Finucane, H., Grossman, S., Vean, G. M., Turnbaugh, P., Lander, E., Mitzenmacher, M., and Sabeti, P. (2011). Detecting novel associations in large data sets. *science*, 334(6062) :1518–1524.

Rivest, R., Shamir, A., and Adleman, L. (1978). A Method for Obtaining Digital Signatures and Public-key Cryptosystems. Communications of the ACM, 21 : 120-126.

Singh, S. (1999). *Histoire des codes secrets. De l'Égypte des pharaons Ã l'ordinateur quantique*. Jean-Claude LattÃ¨s.

Venelli, A. (2010). Efficient Entropy Estimation for Mutual Information Analysis Using B-splines. In *Information Security Theory and Practices. Security and Privacy of Pervasive Systems and Smart Devices*, pages 17–30. Springer.

 ISSN: 2102-6238

Veyrat-Charvillon, N. and cois Xavier Standaert, F. (2009). Mutual Information Analysis : How, When and Why ? In *Cryptographic Hardware and Embedded Systems - CHES 2009, 11th International Workshop, Lausanne, Switzerland, September 6-9, 2009, Proceedings*, volume 5747 of *Lecture Notes in Computer Science*, pages 429–443. Springer.

Whitnall, C. and Oswald, E. (2011). A comprehensive evaluation of mutual information analysis using a fair evaluation framework. In *Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference*, volume 6841 of *Lecture Notes in Computer Science*, page 311. Springer.

Whitnall, C., Oswald, E., and Mather, L. (2011). An Exploration of the Kolmogorov-Smirnov Test as a Competitor to Mutual Information Analysis. In Springer, editor, *Smart Card Research and Advanced Applications*, pages 234–251.

 ISSN: 2102-6238