Cifração autenticada utilizando PUFs

Amanda Cristina Davi Resende[1], **Diego F. Aranha**[1]

[1]Departamento de Ciência da Computação – Universidade de Brasília (UnB)
Caixa Postal 4466 – 70919-970 – Brasília, DF – Brasil

amandadavi7@aluno.unb.br,dfaranha@unb.br

Abstract. *This paper proposes constructions for encryption and authentication based on Physical Unclonable Functions (PUFs). A block cipher is built from 4-round Luby-Rackoff ciphers involving PUFs and universal hash functions and improves the state of the art in PUF-based encryption both in security and ciphertext length. A Message Authentication Code (MAC) is derived from a classical fixed-length MAC and a universal hash function. The security of the two constructions is analyzed considering standard notions in the literature. Since PUFs implicitly encode cryptographic keys, a natural application of the techniques is to perform authenticated encryption of hard disks or mobile devices, with the main advantage of increased resistance against key leakage.*

Resumo. *Este artigo propõe construções para cifração e autenticação baseadas em Funções Fisicamente Não-Clonáveis (*Physical Unclonable Functions – PUFs*). Uma cifra de bloco é construída a partir de uma cifra Luby-Rackoff com 4 rodadas envolvendo PUFs e funções de* hash *universal e aprimora o estado da arte de cifração baseada em PUFs tanto em segurança quanto no comprimento do criptograma resultante. Um Código de Autenticação de Mensagem (MAC) é construído pela combinação de um MAC clássico de tamanho fixo com uma função de* hash *universal. Em ambos os casos, análises de segurança são fornecidas considerando noções padronizadas na literatura. Como as PUFs codificam chaves criptográficas implícitas, as técnicas apresentadas podem ser empregadas em esquemas de cifração autenticada de discos rígidos ou dispositivos móveis, com incremento de segurança por resistência ao vazamento de* bits *da chave.*

1. Introdução

A criptografia moderna desenvolveu diversas ferramentas e metodologias úteis para se analisar construções e fornecer argumentos formais da segurança de sistemas criptográficos. Em argumentos dessa natureza, um modelo de adversário é definido a partir de noções de segurança e um limitante superior de poder computacional e reduções entre problemas relacionam a segurança das construções à dificuldade da resolução de certos problemas. Além disso, as primitivas criptográficas que compõem o sistema são comumente substituídas por versões ideais com propriedades matemáticas precisas, mas muitas vezes não observadas ou demonstradas na prática. Na maioria dos casos, utiliza-se o que chamamos de "abordagem em caixa preta", onde o adversário tem conhecimento do algoritmo utilizado, mas não tem acesso ao seu estado interno ou chave criptográfica.

Esta abordagem nem sempre se aplica ao se implementar sistemas criptográficos, quando chaves criptográficas de longa duração podem ser armazenadas tanto

em memória não-volátil (ROM, EEPROM, *flash*) quanto em memória volátil (RAM). Em ambos os casos, há clara dificuldade em se armazenar chaves de forma segura, uma vez que adversários com acesso físico ao equipamento podem montar ataques de canal lateral para recuperar *bits* da chave a partir do estado latente da memória [Halderman et al. 2009] ou pelo monitoramento de características físicas do sistema [Kocher et al. 1999]. Desta forma, os *bits* restantes podem ser recuperados por ataques de busca exaustiva de menor complexidade [Bernstein 2005] ou por heurísticas mais avançadas [Bonneau and Mironov 2006, Heninger and Shacham 2009, Patsakis 2013].

Uma possível contramedida para se evitar o vazamento de *bits* da chave é a utilização de primitivas criptográficas com chave implícita, como Funções Fisicamente Não-Clonáveis (*Physical Unclonable Functions* – PUFs). PUFs foram introduzidas em [Pappu et al. 2002, Gassend et al. 2002] para se construir primitivas criptográficas que dependam de hipóteses tanto físicas quanto computacionais. Como as funções são determinadas pela estrutura física única de um dispositivo eletrônico, sintetizado por um processo intrinsicamente aleatório e com alto número de grau de liberdade, PUFs apresentam as propriedades de não-clonabilidade e inviolabilidade. Qualquer tentativa de reproduzir ou manipular uma PUF pré-existente deve causar a destruição de sua estrutura física, tornando-a inútil ou transformando-a em uma nova PUF [Tuyls et al. 2006].

O protocolo de acesso a uma PUF é do tipo desafio-resposta, onde as entradas são os desafios e as saídas constituem as respostas [Maes and Verbauwhede 2010]. Implementações reais de PUFs costumam produzir funções ruidosas e independentes [Armknecht et al. 2009], onde a mesma entrada pode gerar saídas diferentes e não-uniformes, desde que a distância de Hamming entre as possíveis saídas seja limitada superiormente; e saídas de diferentes PUFs para uma mesma entrada devem ser indistinguíveis entre si. A correspondência entre desafios e respostas, determinada pelo processo de fabricação da PUF, codifica implicitamente um segredo permanente. Por essa razão, diversos trabalhos na literatura já utilizaram PUFs para construir primitivas criptográficas ou aplicações de segurança, como por exemplo: cifra de bloco resistente ao vazamento de chave [Armknecht et al. 2009], autenticação de dispositivos [Suh and Devadas 2007], geração de chaves criptográficas [Maes et al. 2012], geração de números aleatórios [Odonnell et al. 2004] e proteção de propriedade intelectual [Guajardo et al. 2007].

Como a utilização da PUF em protocolos criptográficos requer a posse física da PUF, uma aplicação convencional de cifração exigiria a transmissão da PUF por algum meio físico, algo inviável em um cenário realista. Por esse motivo, a aplicação de cifração ideal para PUFs é garantir sigilo de dados armazenados em meio fisicamente acessível ao adversário. Entretanto, a posse física da PUF implica na possibilidade de decifração. Logo, PUF e meio de armazenamento devem ser desacoplados. É possível também combinar PUFs com senhas para obter um efeito similar à autenticação com múltiplos fatores [Frikken et al. 2009]. Além disso, vale salientar que autenticar o meio cifrado impede um adversário de realizar alterações sem ser detectado.

Neste trabalho, revisitamos a aplicação de PUFs para cifração autenticada de disco ou dispositivos móveis, aprimorando o estado-da-arte tanto em segurança como em eficiência (tamanho do criptograma). O documento está dividido da seguinte forma: na Seção 2, apresentamos as definições utilizadas no desenvolvimento do trabalho, e na Seção 3 a

formalização empregada de PUFs. Nas Seções 4 e 5, apresentamos nossas propostas para cifração e autenticação, respectivamente, com as análises de segurança correspondentes. Na Seção 6, discutimos as conclusões e apontamos direções para trabalhos futuros.

2. Definições preliminares

Nas definições a seguir, o conjunto $\mathcal{K}$ denota o espaço de chaves e o conjunto $\mathcal{M}$ o espaço de mensagens. Para um conjunto S, a notação $x \xleftarrow{R} S$ indica que x foi amostrado a partir de S com probabilidade uniforme. O adversário $\mathcal{A}^{\varepsilon_k(\cdot)}$ possui acesso a um oráculo de cifração que, em resposta a uma consulta x, retorna $y = \varepsilon_k(x)$ e o adversário $\mathcal{A}^{\varepsilon_k(\$|\cdot|)}$ possui acesso a um oráculo de cifração que, em resposta a uma consulta x, escolhe $x' \xleftarrow{R} \{0,1\}^{|x|}$ e então retorna $y = \varepsilon_k(x')$.

2.1. Noções de segurança

As principais noções de segurança utilizadas na literatura são a indistinguibilidade contra ataques de texto claro escolhido (*Indistinguishability under Chosen Plaintext Attack* – IND-CPA) e a indistinguibilidade contra ataques de criptograma escolhido (*Indistinguishability under Chosen Ciphertext Attack* – IND-CCA). Existem outras noções de segurança, tais como: *Left or Right* (LOR), *Find then Guess* (FTG) e segurança semântica (SEM) [Bellare et al. 1997]. Essas noções de segurança podem ser equivalentes ou implicar umas nas outras. Por exemplo, sabe-se que segurança semântica e indistinguibilidade são noções equivalentes [Goldwasser and Micali 1984]. Neste trabalho, utilizaremos a noção de segurança *Real or Random* (ROR), por sua conveniência na análise de construções de cifras de bloco e equivalência com a noção de indistinguibilidade.

Indistinguibilidade

A noção de IND-CPA reflete a dificuldade computacional de um adversário diferenciar a cifração de duas mensagens de sua escolha, mesmo que o próprio tenha acesso a um oráculo de cifração para cifrar mensagens arbitrárias. Na definição de IND-CCA, o adversário tenta realizar a mesma tarefa, mas agora com acesso a um oráculo de decifração que decifra criptogramas também arbitrários.

Na versão não-adaptativa (IND-CCA1), o adversário pode fazer consultas aos oráculos somente até que ele receba o criptograma de desafio. Já na definição adaptativa (IND-CCA2), o adversário pode continuar a fazer consultas aos oráculos mesmo após ter recebido o criptograma de desafio, com a restrição de que ele não pode consultar o oráculo de decifração com o criptograma de desafio pois, caso contrário, a tarefa seria trivial.

Apresentamos abaixo a noção formal de segurança para ataques de texto claro escolhido, que pode ser facilmente adaptada para ataques de criptograma escolhido, acrescentando o acesso ao oráculo de decifração por parte do adversário e restrições correspondentes.

Definição 2.1. Indistinguibilidade sob ataques de texto claro escolhido – IND-CPA. [Katz and Lindell 2008, Adaptada da Definição 3.21].

Sejam $\mathcal{A}$ um adversário com poder computacional polinomial, $\Gamma = (\varepsilon, \Delta)$ uma cifra simétrica e $Privk^{cpa}_{\mathcal{A},\Gamma}(n)$ a execução de um experimento com $\mathcal{A}$, parametrizado pelo nível de segurança n:

- A chave k é gerada fazendo-se $k \xleftarrow{R} \mathcal{K}$;
- $\mathcal{A}$ recebe 1^n, o oráculo de cifração $\varepsilon_k(\cdot)$ e produz $m_0, m_1 \in \mathcal{M}$ com $|m_0| = |m_1|$;
- Um *bit* $b \xleftarrow{R} \{0, 1\}$ é escolhido, e então o criptograma de desafio $c = \varepsilon_k(m_b)$ é calculado e entregue à $\mathcal{A}$;
- O adversário $\mathcal{A}$ continua a ter acesso ao oráculo $\varepsilon_k(\cdot)$, e produz *bit* b';
- A saída do experimento é 1 se $b' = b$ e 0 caso contrário. $\mathcal{A}$ tem sucesso quando $Privk^{cpa}_{\mathcal{A},\Gamma}(n) = 1$.

Uma cifra Γ é indistinguível sob ataques de texto claro escolhido se para todo adversário $\mathcal{A}$, existe função λ desprezível no parâmetro de segurança n:

$$Adv^{cpa}_{\mathcal{A},\Gamma} \stackrel{def}{=} |\Pr[k \xleftarrow{R} \mathcal{K} : \mathcal{A}^{\varepsilon_k(m_0)} = 1] - \Pr[k \xleftarrow{R} \mathcal{K} : \mathcal{A}^{\varepsilon_k(m_1)} = 1]| \leq \lambda(n).$$

Real or Random

Na noção de segurança ROR, o adversário não pode diferenciar a cifração de uma mensagem de sua escolha da cifração de uma mensagem escolhida aleatoriamente com vantagem não desprezível. Temos que considerar dois diferentes jogos. No jogo real, começamos escolhendo uma chave aleatória $k \xleftarrow{R} \mathcal{K}$. Então é fornecido para o adversário $\mathcal{A}$ um oráculo que, quando consultado com uma cadeia de caracteres $x \in \mathcal{M}$, responde com uma cifração de x sob a chave k. Já no jogo aleatório, começamos escolhendo uma chave aleatória $k \xleftarrow{R} \mathcal{K}$, como no jogo real. Então é fornecido para o adversário $\mathcal{A}$ um oráculo que, quando consultado com $x \in \mathcal{M}$, responde com uma cifração de uma cadeia de caracteres aleatória de tamanho $|x|$.

Um sistema de cifração é considerado seguro para essa noção de segurança se nenhum adversário pode obter vantagem significativa em distinguir entre os jogos real e aleatório em um limite de tempo polinomial.

Definição 2.2. *Real or Random* (ROR). Um sistema criptográfico $\Gamma = (\varepsilon, \Delta)$ possui segurança ROR se qualquer adversário com poder computacional polinomial possui vantagem limitada por uma função λ desprezível no parâmetro de segurança n:

$$Adv^{rr}_{\mathcal{A},\Gamma} \stackrel{def}{=} \Pr[k \xleftarrow{R} \mathcal{K} : \mathcal{A}^{\varepsilon_k(\cdot)} = 1] - \Pr[k \xleftarrow{R} \mathcal{K} : \mathcal{A}^{\varepsilon_k(\$|\cdot|)} = 1] \leq \lambda(n).$$

Inforjabilidade

As noções de segurança para primitivas de autenticação são ligeiramente diferentes, pois refletem a dificuldade computacional enfrentada por um adversário que objetiva forjar um autenticador para uma mensagem de sua escolha. O adversário tipicamente possui acesso a um oráculo de cálculo de autenticadores, sob a restrição de que a mensagem oferecida por ele como evidência de forja não pode ter sido consultada como entrada do oráculo.

Definição 2.3. Sejam $\mathcal{A}$ um adversário com poder computacional polinomial, $\Gamma =$ *(Mac, Vrfy)* um código de autenticação de mensagem e *Mac-forge*$_{\mathcal{A},\Gamma}(n)$ a execução de um experimento com $\mathcal{A}$, parametrizado pelo nível de segurança n:

- A chave k é gerada fazendo uma escolha uniformemente aleatória $k \xleftarrow{R} \mathcal{K}$;

- $\mathcal{A}$ recebe 1^n, acesso ao oráculo Mac_k e produz (m, t) após $\mathcal{Q}$ consultas ao oráculo;
- A saída do experimento é 1 se *Vrfy* $(m, Mac(m, t)) = 1$, com $m \neq \mathcal{Q}$, e 0 caso contrário.

Um código de autenticação de mensagens Γ é existencialmente inforjável contra ataque adaptativo de mensagem escolhida se, para todo adversário $\mathcal{A}$, existe função λ desprezível do parâmetro de segurança n:

$$\Pr[\textit{Mac-forge}_{\mathcal{A},\Gamma}(n) = 1] \leq \lambda(n).$$

2.2. Funções de *hash* universal

Funções de *hash* universal são funções escolhidas aleatoriamente de uma família $\mathcal{H}$ de funções. Os valores ϵ_1 e ϵ_2 são desprezíveis em função de um parâmetro de segurança n. Para maiores detalhes e classificação, ver [Ramzan 2001]. A seguir, serão apresentadas apenas definições para os tipos de funções que serão utilizados.

Definição 2.4. Dizemos que $\mathcal{H}$ é uma família ϵ_1-universal de funções de *hash* se para todo $x \neq y \in \mathcal{D}$ (domínio), $z \in \mathcal{R}$ (imagem),

$$\Pr[a \overset{R}{\leftarrow} \mathcal{K}(\mathcal{H}) : h_a(x) - h_a(y) = z] \leq \epsilon_1(n).$$

Definição 2.5. Dizemos que $\mathcal{H}$ é uma família ϵ_2- bissimétrica de funções de *hash* se para todo $x, y \in \mathcal{D}$ (aqui permite-se que $x = y$), $z \in \mathcal{R}$,

$$\Pr[a_1 \overset{R}{\leftarrow} \mathcal{K}(\mathcal{H}), a_2 \overset{R}{\leftarrow} \mathcal{K}(\mathcal{H}) : h_{a_1}(x) + h_{a_2}(y) = z] \leq \epsilon_2(n).$$

3. Formalização de PUFs

Neste trabalho, será considerada a formalização de PUFs apresentada no trabalho [Armknecht et al. 2009], a qual depende do conjunto de definições a seguir.

3.1. Funções ruidosas

Para três inteiros positivos l, m, δ com $0 \leq \delta \leq m$, uma (l, m, δ)-função ruidosa f^* é um algoritmo probabilístico que aceita entradas (desafios) $x \in \{0,1\}^l$ e gera saídas (respostas) $y \in \{0,1\}^m$ tais que a distância de Hamming entre duas saídas para a mesma entrada seja no máximo δ.

3.2. Funções Pseudoaleatórias (Fracas)

Considere uma família de funções $\mathcal{F}$ com entrada $\{0,1\}^l$ e saída $\{0,1\}^m$. Dizemos que $\mathcal{F}$ é (q_{prf}, ϵ_{prf})$-$*pseudoaleatória (PRF) em relação a uma distribuição* $\mathbb{D}$ sobre $\{0,1\}^m$, se a vantagem de distinguir entre as duas distribuições a seguir, para escolhas adaptativas distintas entre si de entradas $(x_1, \ldots, x_{q_{pfr}})$ é no máximo ϵ_{prf}:

- $y_i = f(x_i)$ onde $f \overset{R}{\leftarrow} \mathcal{F}$.
- $y_i \leftarrow \mathbb{D}$.

$\mathcal{F}$ é chamada de pseudoaleatória fraca (wPRF) em relação a uma distruibuição $\mathbb{D}$ se suas entradas não são escolhidas por um adversário, mas sim escolhidas aleatoriamente de $\{0,1\}^l$. Essas funções diferem da definição clássica [Katz and Lindell 2008] que considera a distribuição $\mathbb{D}$ como a distribuição uniforme $\mathbb{U}_m$.

3.3. Extrator Difuso

Um (m, n, δ, ϵ_{FE}) extrator difuso (do inglês, *fuzzy*) E é um par de procedimentos probabilísticos de geração Gen: $\{0,1\}^m \rightarrow \{0,1\}^n \times \{0,1\}^*$ e de reprodução Rep: $\{0,1\}^m \times \{0,1\}^* \rightarrow \{0,1\}^n$, onde o procedimento de geração produz informação auxiliar para que sua saída possa ser recuperada a partir do procedimento de reprodução.

A propriedade de correção garante que para $(z, \omega) = Gen(y)$ e $y' \in \{0,1\}^m$ com a distância de Hamming $dist(y, y') \leq \delta$, então $Rep(y', \omega) = z$. Se $dist(y, y') > \delta$, então nenhuma garantia é fornecida sobre a saída de Rep. A propriedade de segurança garante que, mesmo para adversários com acesso à informação auxiliar ω, a distância entre a distribuição observada de z e a distribuição uniforme é no máximo ϵ_{FE}, desprezível em função de um parâmetro de segurança.

3.4. PUF-wPRFs

Considere agora a família $\mathcal{V}$ de funções PUF-wPRF, construída a partir de uma família $\mathcal{P}$ de PUFs pseudoaleatórias fracas e um extrator difuso E. Uma família $\mathcal{V}$ é um conjunto de pares de procedimentos probabilísticos de geração e reprodução, com geração $Gen \circ \Pi$, para $\Pi \in \mathcal{P}$ com entrada $x \in \{0,1\}^l$ e saída $(z, \omega_x) \stackrel{def}{=} Gen(\Pi(x)) \in \{0,1\}^n \times \{0,1\}^*$; e reprodução $Rep \circ \Pi$ com entrada $(x, \omega_x) \in \{0,1\}^l \times \{0,1\}^*$ e saída $z = Rep(\Pi(x), \omega_x)$. Resta enunciar a propriedade de segurança dessa composição, derivada em [Armknecht et al. 2009].

Proposição 3.1 (Pseudoaleatoriedade da composição PUF com extrator difuso E)**.** *Seja $\mathcal{P}$ uma família de PUFs $(q_{prf}, \epsilon_{prf})-$pseudoaleatória com respeito a uma distribuição $\mathbb{D}$ e E = (Gen, Rep) um extrator difuso. A vantagem de qualquer adversário que escolhe entradas adaptativas distintas entre si $(x_1, \ldots, x_{q_{prf}})$ e recebe como saídas $((z_1, \omega_1), \ldots, (z_{qprf}, \omega_{qprf}))$ para diferenciar entre as duas distribuições a seguir é no máximo $\epsilon_{prf} + q_{prf} \times \epsilon_{FE}$:*

- *(z_i, ω_i) = $Gen(\Pi(x_i))$, onde $\Pi \stackrel{R}{\leftarrow} \mathcal{P}$.*
- *(z_i, ω_i) onde $z_i \stackrel{R}{\leftarrow} \{0,1\}^n$, $(z_i', \omega_i) = Gen(\Pi(x_i))$ e $\Pi \stackrel{R}{\leftarrow} \mathcal{P}$.*

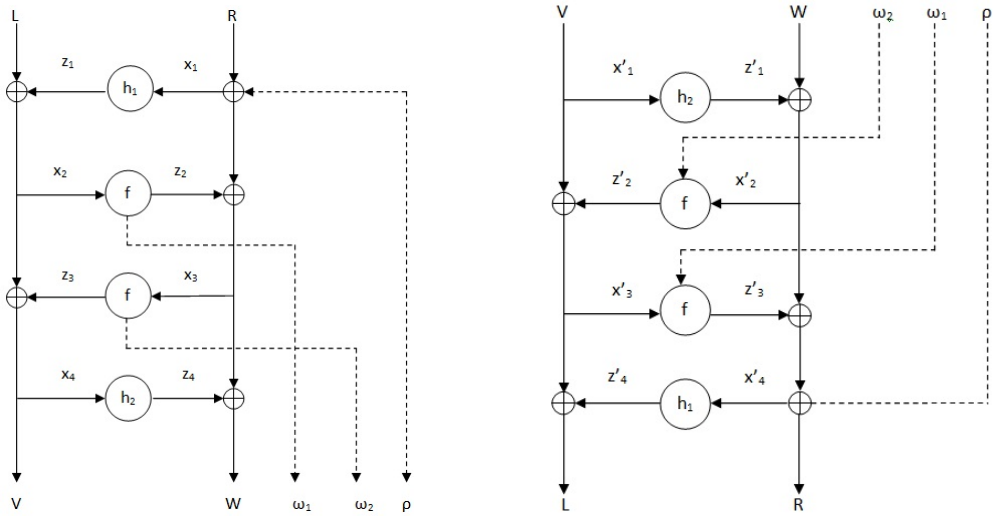
Vale ressaltar que as informações auxiliares do extrator difuso podem vazar algumas informações sobre a entrada, exigindo assim cuidado na escolha da PUF.

4. Proposta de cifra de bloco

Nesta seção, será apresentada a construção para cifração proposta. Seja $\mathcal{V}$ uma família de PUF-wPRFs com entrada e saída em $\{0,1\}^n$, $\mathcal{H}$ uma família de funções de *hash* ϵ_1-universal ϵ_2-bissimétrica com entrada e saída também de comprimento n e a entrada para a cifra em $\{0,1\}^{2n}$.

A cifra de bloco $\epsilon^{\mathcal{VH}}$ possui 4 rodadas e se baseia na construção Luby-Rackoff [Luby and Rackoff 1986] com modificações de [Ramzan 2001]. Na construção original, as funções de rodada consistiam em PRFs ou funções de *hash* universal. Na construção proposta, são realizadas duas invocações à uma PUF-wPRF $f \in \mathcal{V}$ e a duas funções de *hash* universal distintas $h_i \in \mathcal{H}$, $i = 1, 2$. Como pode ser visto na Figura 1(a), a primeira e a última funções de rodada são as funções h_i. O texto cifrado é (V, W, ω_1, ω_2 e ρ) e compreende os blocos propriamente ditos, as informações auxiliares ω_1, ω_2 produzidas pelo extrator difuso e o vetor de inicialização aleatório $\rho \in \{0,1\}^n$.

Figura 1. Proposta de cifração e decifração



(a) Proposta de cifração baseada na construção Luby-Rackoff

(b) Decifração da construção baseada na cifra Luby-Rackoff

4.1. Decifração

A decifração, que pode ser vista na Figura 1(b), é similar para o caso de uma cifra Luby-Rackoff tradicional, onde a informação auxiliar ω_i é utilizada junto com o procedimento de reprodução para reconstruir as saídas z_2 e z_3 das PUF-wPRFs da segunda e terceira funções de rodadas e o valor ρ para reverter a entrada utilizada na primeira função de rodada durante a cifração.

4.2. Segurança contra ataques de texto claro escolhido

A intuição para a demonstração a seguir é baseada na noção de segurança ROR-CPA, onde o jogo 1 consiste na randomização da saída da primeira ocorrência de f, o que equivale a randomizar R, e o jogo 2 consiste na randomização da entrada da primeira ocorrência de f, o que equivale a randomizar L. Como a vantagem de distinguir L de L' (lado esquerdo da mensagem escolhida aleatoriamente) é desprezível, a vantagem de distinguir R de R' (lado direito da mensagem escolhida aleatoriamente) também é desprezível, e os jogos 1 e 2 são indistinguíveis entre si, o adversário tem vantagem desprezível em distinguir a cifração de um texto claro de sua escolha da cifração de uma mensagem aleatória.

Teorema 4.1. *Seja $\varepsilon^{\mathcal{VH}}$ a cifra da Figura 1(a) usando a família $\mathcal{V}$ de PUF-wPRFs e a família $\mathcal{H}$ de funções de hash ϵ_1-universal ϵ_2-bissimétrica. Então a vantagem do adversário que realiza até q_{prf} consultas é no máximo:*

$$4\epsilon_{prf} + 2q_{prf} \times \epsilon_{FE} + 2q_{prf}^2 \times \epsilon_1(n).$$

Demonstração. Seja $(L^{(i)}, R^{(i)})$, $i = 1, \ldots, q_{prf}$; a sequência de escolhas adaptativas de textos claros e $x_j^{(i)}$, $z_j^{(i)}$ a entrada e saída respectivas para a função de rodada j dentre (h_1, f, f e h_2), e ρ é um valor uniformemente aleatório utilizado para tornar aleatório a parte R do texto claro. O teorema será provado pela definição de uma sequência de jogos e

estimativa da vantagem de se distinguir entre eles. O jogo real é portanto o cenário onde o adversário recebe a cifração do texto claro escolhido por ele.

No jogo 1, as saídas $z_2^{(i)}$ da segunda função de rodada f são trocadas por valores uniformemente aleatórios $\tilde{z}_2^{(i)} \xleftarrow{R} \{0,1\}^n$. Sob a hipótese de que os valores $x_2^{(i)}$ são distintos entre si, a vantagem de distinguir entre os dois casos de acordo com a Proposição 3.1 é no máximo $\epsilon_{prf} + q_{prf} \times \epsilon_{FE}$. Além disso, temos que considerar a probabilidade de colisão entre $R^{(i)} \oplus \rho$ para dois valores $R^{(i)}$ distintos e a probabilidade de colisão em h_1 para dois valores $x_1^{(i)}$ distintos, que totalizam no máximo $q_{prf}^2 \times \epsilon_1(n)$. Como consequência, a vantagem para distinguir entre o jogo real e o jogo 1 é limitada superiormente por $\epsilon_{prf} + q_{prf} \times \epsilon_{FE} + q_{prf}^2 \times \epsilon_1(n)$.

O jogo 2 é definido como o jogo 1, onde agora as entradas $x_2^{(i)}$ para a segunda função de rodada são trocadas por $\tilde{x}_2^{(i)} \xleftarrow{R} \{0,1\}^n$ selecionados aleatoriamente. Observe que os valores de $x_2^{(i)}$ são usados em dois diferentes contextos: i) para calcular o lado esquerdo do criptograma (pela adição módulo 2 com a saída da terceira função de rodada) e ii) como entrada para a segunda função de rodada. Em relação a i), observe que as saídas da terceira função de rodada são independentes dos valores de $x_2^{(i)}$, com os valores de $\tilde{z}_2^{(i)}$ (e portanto as entradas da terceira função de rodada) uniformemente escolhidos por definição, e que os valores $x_2^{(i)}$ são independentes do texto claro (por causa da escolha da função de *hash* universal). Assim, i) e ii) representam duas características independentes, possibilitando distinguir entre o jogo 1 e o jogo 2, e assim são examinadas separadamente.

A vantagem de distinguir entre o jogo 1 e o 2 baseado em i) é equivalente a decidir se os valores $L^{(i)} \oplus z_1 \oplus z_3$ são uniformemente aleatórios ou pertencem às saídas da terceira função de rodada. Com os mesmos argumentos acima utilizados, a vantagem é limitada superiormente por $\epsilon_{prf} + q_{prf} \times \epsilon_{FE} + q_{prf}^2 \times \epsilon_1(n)$.

A vantagem de distinguir entre o jogo 1 e o jogo 2 baseado em ii) é no máximo a vantagem de distinguir $(\Pi(x_1), \ldots, \Pi(x_{qprf}))$ de $(\Pi(\tilde{x}_1), \ldots, \Pi(\tilde{x}_{qprf}))$ onde Π denota a PUF usada na segunda função de rodada. Pela definição de PUFs, a vantagem de distinguir entre $(\Pi(x_1), \ldots, \Pi(x_{qprf}))$ e $y_1, \ldots, y_{qprf}$, onde $y_i \leftarrow \mathbb{D}$ para $\mathbb{D}$ uma distribuição apropriada é no máximo ϵ_{prf} [Armknecht et al. 2009]. Na verdade, o mesmo vale para $(\Pi(\tilde{x}_1), \ldots, \Pi(\tilde{x}_{qprf}))$ (o fato de que os valores $\tilde{x}_i^{(1)}$ são desconhecidos não podem aumentar a vantagem). Assim, por desigualdade triangular, segue-se que a vantagem em relação a ii) é no máximo $2\epsilon_{prf}$. No total a vantagem de distinguir entre o jogo 1 e o jogo 2 é menor ou igual a $3\epsilon_{prf} + q_{prf} \times \epsilon_{FE} + q_{prf}^2 \times \epsilon_1(n)$.

Finalmente, observamos que é indistinguível se $x_2^{(i)}$ ou $L^{(i)}$ é aleatório e também se $z_2^{(i)}$ ou $R^{(i)}$ é aleatório. Assim, o jogo 2 é indistinguível de um jogo aleatório onde os textos claros são aleatórios. Por transitividade, a vantagem de um adversário real ou aleatório é no máximo $4\epsilon_{prf} + 2q_{prf} \times \epsilon_{FE} + 2q_{prf}^2 \times \epsilon_1(n)$. □

4.3. Segurança contra ataques de criptograma escolhido

A intuição para a demonstração a seguir é baseada na noção de segurança ROR-CCA1 e segue a demonstração do Teorema 4.1, com a única diferença que aqui o adversário pode fazer consultas ao oráculo de decifração $\Delta^{\mathcal{VH}}$.

Teorema 4.2. *Seja $\varepsilon^{\mathcal{VH}}$ a função de cifração (Figura 1(a)) e $\Delta^{\mathcal{VH}}$ a função decifração (Figura 1(b)) usando a família $\mathcal{V}$ de PUF-wPRFs e a família $\mathcal{H}$ de funções de hash ϵ_1- universal ϵ_2- bissimétrica. Então a vantagem do adversário que realiza até q_{prf} consultas é no máximo:*

$$8\epsilon_{prf} + 4q_{prf} \times \epsilon_{FE} + 4q_{prf}^2 \times \epsilon_1(n) + q_{prf}^2 \times \epsilon_2(n).$$

Demonstração. Para o caso de cifração temos que a vantagem de um adversário distinguir entre o jogo real ou aleatório é no máximo $4\epsilon_{prf} + 2q_{prf} \times \epsilon_{FE} + 2q_{prf}^2 \times \epsilon_1(n)$, de acordo com o Teorema 4.1. Agora será considerado o caso em que o adversário pode fazer consultas de decifração. Seja $(V^{(i)}, W^{(i)})$, $i = 1, \ldots, q_{prf}$; uma sequência de escolhas adaptativas de criptogramas, onde $x'^{(i)}_j$, $z'^{(i)}_j$ são entrada e saída respectivas para a função da rodada j dentre (h_2, f, f, h_1) e o valor ρ reverte a entrada utilizada na primeira função de rodada durante a cifração. O teorema será provado pela estimativa da vantagem de se distinguir entre uma sequência de jogos, na verdade os mesmos utilizados no Teorema 4.1, com a diferença que agora o adversário possui acesso ao oráculo de decifração $\Delta^{\mathcal{HF}}$.

Temos que o jogo 1 consiste na randomização da saída da primeira ocorrência de f, ou seja, as saídas $z'^{(i)}_2$ da segunda função de rodada f são trocadas por alguns valores uniformemente aleatórios $\tilde{z'}^{(i)}_2 \xleftarrow{R} \{0,1\}^n$ o que equivale a randomizar W. Já o jogo 2 consiste na randomização da entrada da primeira ocorrência de f, ou seja, as entradas $x'^{(i)}_2$ para a segunda função de rodada são trocadas por $\tilde{x'}^{(i)}_2 \xleftarrow{R} \{0,1\}^n$, o que equivale a randomizar V.

Como a vantagem de distinguir V de V' (lado esquerdo do criptograma) aleatório é desprezível, a vantagem de distinguir W de W' (lado direito do criptograma) aleatório também é desprezível, e os jogos 1 e 2 são indistinguíveis entre si, assim o adversário tem vantagem desprezível em distinguir a decifração do criptograma normal ou de um criptograma aleatório.

Assim, utilizando o mesmo argumento do Teorema 4.1, temos que a vantagem do adversário real ou aleatório fazendo até q_{prf} consultas é no máximo $4\epsilon_{prf} + 2q_{prf} \times \epsilon_{FE} + 2q_{prf}^2 \times \epsilon_1(n)$. Como agora o adversário tem acesso à $\Delta^{\mathcal{FH}}$, temos que analisar o evento onde $1 \leq i, j \leq q$ tal que $h_1(R^{(i)}) \oplus L^{(i)} = W^{(j)} \oplus h_2(V^{(j)})$. De acordo com [Ramzan 2001], temos que:

$$\begin{aligned}
&\Pr[\exists\, 1 \leq i, j \leq q_{prf} : h_1(R^{(i)}) \oplus L^{(i)} = W^{(j)} \oplus h_2(V^{(j)})] \\
&\quad \leq \sum_{1 \leq i,j \leq q_{prf}} \Pr[h_1(R^{(i)}) \oplus L^{(i)} = W^{(j)} \oplus h_2(V^{(j)})] \\
&\quad \leq \sum_{1 \leq i,j \leq q_{prf}} \Pr[h_1(R^{(i)}) \oplus h_2(V^{(j)}) = W^{(j)} \oplus L^{(i)}] \\
&\quad \leq q_{prf}^2 \times \epsilon_2(n).
\end{aligned}$$

Assim, a vantagem do adversário distinguir entre o jogo real ou aleatório baseada na noção de segurança ROR-CCA1 é no máximo: $8\epsilon_{prf} + 4q_{prf} \times \epsilon_{FE} + 4q_{prf}^2 \times \epsilon_1(n) + q_{prf}^2 \times \epsilon_2(n)$. □

Desta forma, a segurança da construção de cifração fica reduzida às propriedades de segurança da composição PUF com extrator difuso, tanto para ataques de texto claro escolhido quanto ataques de criptograma escolhido.

4.4. Comparação com trabalhos relacionados

A cifra proposta em [Armknecht et al. 2009] é uma combinação da construção Luby-Rackoff com PUF-wPRF e possui 3 funções de rodadas, onde cada rodada é uma PUF-wPRF diferente e possui o valor aleatório ρ utilizado para tornar aleatória a entrada da primeira função de rodada.

Diferentemente da cifra mencionada acima, a construção proposta utiliza 4 funções de rodadas, onde apenas a segunda e a terceira funções de rodadas utilizam PUF-wPRFs, reduzindo assim o tamanho do criptograma, uma vez que cada chamada a uma PUF-wPRF gera informação auxiliar (ω_i) que precisa ser fornecida durante a decifração. Adicionalmente, a cifra Luby-Rackoff com 4 rodadas possui a vantagem de fornecer segurança contra ataques de criptograma escolhido [Ramzan 2001].

A estrutura da cifra proposta com 4 rodadas é praticamente a mesma apresentada em [Ramzan 2001], com algumas diferenças. A segunda e terceira funções de rodadas em [Ramzan 2001] são PRFs, diferentemente da construção proposta que utiliza PUF-wPRFs. A principal vantagem de se utilizar PUF-wPRFs é não ser necessário armazenar uma chave passível de captura, visto que a chave está implicitamente embutida na configuração da PUF. É importante observar que essa substituição nem sempre é trivial e exige tanto adaptações na construção, para refletir a pseudoaleatoriedade fraca de uma PUF-wPRF [Armknecht et al. 2009], quanto modificações nos argumentos formais de segurança. A primeira e quarta funções de rodadas consistem em uma função de *hash* universal em ambos os casos.

5. Proposta de autenticação

Em aplicações de cifração de disco e outros dispositivos, a confidencialidade não é o único serviço de segurança desejável, visto que a simples cifração não impede um adversário ativo de manipular texto cifrado com o objetivo de causar alterações maliciosas no texto claro resultante da decifração ou corrupção do conteúdo armazenado sem possibilidade de detecção posterior.

Para que essa detecção seja possível, utiliza-se tipicamente construções para Código de Autenticação de Mensagem (*Message Authentication Code* – MAC), o que permite alcançar IND-CCA 2. A combinação de um esquema de cifração com IND-CPA com um autenticador inforjável contra ataques de mensagem escolhida, a partir de construções genéricas (por exemplo, [Bellare and Namprempre 2000]), permite atingir a noção de segurança IND-CCA2.

É importante observar que cifração autenticada de discos, quando realizada na granularidade natural de bloco, apresenta limitações intrínsecas do ponto de vista de autenticação, visto que um atacante ativo pode sempre substituir um bloco cifrado por um bloco autêntico anteriormente escrito sem ser detectado. A proposta de autenticação aqui apresentada se concentra apenas em detectar tentativas maliciosas de se alterar blocos cifrados do disco que não reutilizem blocos autênticos anteriores.

Em [Black et al. 1999], é proposta uma constução de MAC seguro utilizando funções de *hash* universal e PRFs. O MAC proposto a seguir é baseado nessa construção com a diferença de que, ao invés de utilizar PRFs, emprega-se uma PUF-wPRF. Vale ressaltar que, como a saída da cifra proposta possui comprimento fixo (saída da quarta função de rodada), ou seja, a entrada do MAC tem comprimento fixo, a construção do MAC se torna muito mais simples.

Definição 5.1. Seja uma família de funções de *hash* universal $\mathcal{H} = \{h : \{0,1\}^m \rightarrow \{0,1\}^m\}$, uma família de PUF-wPRF $\mathcal{F} = \{\{0,1\}^m \rightarrow \{0,1\}^m \times \{0,1\}^*\}$ e um vetor de inicialização $\sigma \in \{0,1\}^m$. O Código de Autenticação de Mensagem FMAC($\mathcal{H}$, $\mathcal{F}$) = *TAG*, onde a *TAG* é composta pelo autenticador t e pela informação auxiliar ω conforme a Figura 2, é definido pelos algoritmos de geração de parâmetros, cálculo do MAC e verificação como a seguir, respectivamente:

- Gen: escolher $\sigma \xleftarrow{R} \{0,1\}^m$ uniformente aleatório e $h \leftarrow \mathcal{H}$;
- Mac: ao receber como entrada $\sigma \in \{0,1\}^m$, $h \in \mathcal{H}$ e a mensagem M, calcular:

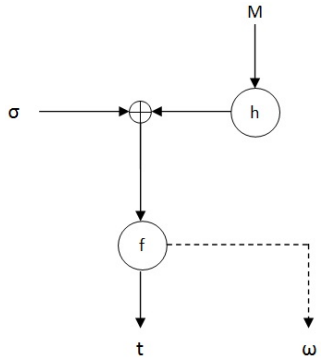
$$t = Mac(M) = f(h(M) \oplus \sigma).$$

- *Vrfy*: ao receber a mensagem M, $h \in \mathcal{H}$ e o autenticador t, produzir a saída é 1 se e somente se:

$$t \stackrel{?}{=} f(h(M)) \oplus \sigma).$$

Fornecer um argumento formal para a segurança da construção de MAC envolve demonstrar que a mesma é inforjavél contra ataque adaptativo de mensagem escolhida.

Figura 2. MAC utilizando função de *hash* universal e PUF-wPRF.



Teorema 5.1. *Sejam $\mathcal{H} = \{h : \{0,1\}^* \rightarrow \{0,1\}^m\}$ uma família de funções de hash $\epsilon-$universal e $\mathcal{F} = \{\{0,1\}^m \rightarrow \{0,1\}^m \times \{0,1\}^*\}$ uma família de PUF-wPRFs. A vantagem de um adversário $\mathcal{A}$ que realiza até q_{prf} consultas é no máximo:*

$$\Pr[\textit{Mac-forge}_{\mathcal{A}}(m) = 1] \leq q_{prf} \times \lambda(n) + \frac{q_{prf}^2}{2^m},$$

com $\lambda(n)$ uma função desprezível do parâmetro de segurança n.

Demonstração. [Katz and Lindell 2008] apresentam uma construção segura de MAC a partir de uma PRF F_k proposta por [Goldreich et al. 1985]. A redução de segurança é

obtida ao se considerar primeiramente um MAC calculado com uma função verdadeiramente aleatória f' e se observar que o adversário não ganha vantagem além da desprezível de forjar em tempo polinomial um MAC válido para uma mensagem de sua escolha ao se substituir f' por um MAC calculado com F_k. Neste caso, a probabilidade do adversário construir um MAC válido continua desprezível.

Conforme a formulação de PUF-wPRFs apresentada em [Armknecht et al. 2009], uma wPRF possui o mesmo comportamento de uma PRF se suas entradas são escolhidas de maneira uniformemente aleatória. Ou seja, a vantagem de se distinguir entre uma PRF e uma wPRF em tempo polinomial é limitada por uma função desprezível $\lambda(n)$. Assim, por transitividade, uma wPRF é indistinguível de uma função ideal em tempo polinomial, desde que suas entradas sejam uniformemente aleatórias. Como o adversário pode realizar até q_{prf} consultas, a vantagem do adversário é de $q_{prf} \times \lambda(n)$, uma função desprezível.

Para que as entradas da wPRF sejam uniformemente aleatórias, temos que considerar a probabilidade de colisão de $h(M) \oplus \sigma$ que é no máximo $\frac{q^2}{2^m}$ [Katz and Lindell 2008, Lema A.9]. Assim temos que a probabilidade de um adversário construir um MAC válido construído a partir de uma PUF-wPRF é no máximo $q_{prf} \times \lambda(n) + \frac{q^2}{2^m}$. □

Fica portanto demonstrada a segurança da construção proposta para autenticação, a partir da propriedades de segurança fornecidas pela formalização da PUF.

6. Conclusão e Trabalhos futuros

A utilização de PUFs como primitiva criptográfica é um campo promissor de pesquisa e promete fornecer soluções elegantes e inovadoras para diversos problemas de ordem criptográfica, como autenticação de transações e identificação de dispositivos.

Neste artigo, foram propostas construções para cifração e autenticação baseadas na combinação de PUFs e funções de *hash* universal. O objetivo das construções é fornecer serviços de segurança resistentes ao vazamento de *bits* da chave, uma vez que não há necessidade de armazenamento de material criptográfico em memória, cenário ideal para aplicações que exijam garantias de sigilo e autenticidade de discos ou dispositivos móveis A cifra de bloco proposta aprimora a melhor construção presente na literatura, fornecendo tanto segurança contra ataques de criptograma escolhido quanto redução do fator de expansão provocado pelo processo de cifração. A construção de autenticação, por sua vez, é bastante simples e envolve apenas uma chamada à PUF. Ambas as construções foram analisadas do ponto de vista de segurança utilizando noções de segurança padronizadas na literatura.

Como possíveis trabalhos futuros, diversas alternativas são possíveis para se reduzir o tamanho do criptograma resultante da construção de cifração apresentada. Em primeiro lugar, adotar extratores determinísticos no lugar de extratores difuso pode tornar as informações auxiliares ω_i constantes para todas as cifrações. Outra possibilidade é relaxar os requisitos de segurança dos vetores de inicialização para que os mesmos sejam determinísticos para um mesmo bloco sendo cifrado. Desta forma, a PUF pode ser utilizada para derivar os vetores de inicialização a partir de metadados que identificam o bloco fisicamente. A integração entre as construções de cifra de bloco e MAC propostas

permite ainda unificar os dois vetores de inicialização. Por fim, aspectos de implementação, como medidas de desempenho e comparações com cifras convencionais são também outras alternativas.

A redução da sobrecarga de armazenamento é fundamental para que as construções propostas sejam viáveis na prática. Entretanto, todos esses trabalhos possuem impacto substancial de segurança e um exame cuidadoso das implicações se faz necessário.

Agradecimentos

Os autores agradecem à CAPES pelo auxílio financeiro, e à *Intel Corporation* pelo financiamento do projeto "*Physical Unclonable Functions for SoC Devices*" (INTEL/ENE - 2012/05156), o qual este trabalho faz parte.

Referências

Armknecht, F., Maes, R., Sadeghi, A.-R., Sunar, B., and Tuyls, P. (2009). Memory leakage-resilient encryption based on physically unclonable functions. In *15th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2009)*, pages 685–702. Springer.

Bellare, M., Desai, A., Jokipii, E., and Rogaway, P. (1997). A concrete security treatment of symmetric encryption. In *38th Annual Symposium on Foundations of Computer Science (FOCS 1997)*, pages 394–403. IEEE.

Bellare, M. and Namprempre, C. (2000). Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In *6th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2000)*, pages 531–545. Springer.

Bernstein, D. J. (2005). Cache-timing attacks on AES. `http://cr.yp.to/antiforgery/cachetiming-20050414.pdf`.

Black, J., Halevi, S., Krawczyk, H., Krovetz, T., and Rogaway, P. (1999). UMAC: fast and secure message authentication. In *Advances in Cryptology (CRYPTO 1999)*, pages 216–233. Springer.

Bonneau, J. and Mironov, I. (2006). Cache-collision timing attacks against AES. In *Cryptographic Hardware and Embedded Systems (CHES 2006)*, pages 201–215. Springer.

Frikken, K. B., Blanton, M., and Atallah, M. J. (2009). Robust authentication using physically unclonable functions. In *Information Security*, pages 262–277. Springer.

Gassend, B., Clarke, D., Van Dijk, M., and Devadas, S. (2002). Silicon physical random functions. In *9th ACM Conference on Computer and Communications Security (CCS 2002)*, pages 148–160. ACM.

Goldreich, O., Goldwasser, S., and Micali, S. (1985). On the cryptographic applications of random functions. In *Advances in Cryptology (CRYPTO 1985)*, pages 276–288. Springer.

Goldwasser, S. and Micali, S. (1984). Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270 – 299.

Guajardo, J., Kumar, S. S., Schrijen, G.-J., and Tuyls, P. (2007). Physical unclonable functions and public-key crypto for FPGA IP protection. In *International Conference on Field Programmable Logic and Applications (FPL 2007)*, pages 189–195. IEEE.

Halderman, J. A., Schoen, S. D., Heninger, N., Clarkson, W., Paul, W., Calandrino, J. A., Feldman, A. J., Appelbaum, J., and Felten, E. W. (2009). Lest we remember: cold-boot attacks on encryption keys. *Communications of the ACM*, 52(5):91–98.

Heninger, N. and Shacham, H. (2009). Reconstructing RSA Private Keys from Random Key Bits. In *29th Annual International Cryptology Conference (CRYPTO 2009)*, pages 1–17. Springer.

Katz, J. and Lindell, Y. (2008). *Introduction to modern cryptography*. Chapman & Hall.

Kocher, P. C., Jaffe, J., and Jun, B. (1999). Differential Power Analysis. In Wiener, M. J., editor, *19th Annual International Cryptology Conference (CRYPTO 1999)*, pages 388–397. Springer.

Luby, M. and Rackoff, C. (1986). Pseudo-random permutation generators and cryptographic composition. In *18th ACM Symposium on Theory of Computing (STOC 1986)*, pages 356–363, New York, USA. ACM.

Maes, R., Herrewege, A. V., and Verbauwhede, I. (2012). PUFKY: A Fully Functional PUF-Based Cryptographic Key Generator. In *14th International Workshop on Cryptographic Hardware and Embedded Systems (CHES 2012)*, pages 302–319. Springer.

Maes, R. and Verbauwhede, I. (2010). Physically unclonable functions: A study on the state of the art and future research directions. In *Towards Hardware-Intrinsic Security*, pages 3–37. Springer.

Odonnell, C. W., Suh, G. E., and Devadas, S. (2004). PUF-based random number generation. MIT CSAIL CSG Technical Memo 481 (`http://csg.csail.mit.edu/pubs/memos/Memo-481/Memo-481.pdf`).

Pappu, R., Recht, B., Taylor, J., and Gershenfeld, N. (2002). Physical One-Way Functions. *Science*, 97:2026–2030.

Patsakis, C. (2013). RSA private key reconstruction from random bits using SAT solvers. Cryptology ePrint Archive, Report 2013/026. `http://eprint.iacr.org/`.

Ramzan, Z. A. (2001). *A study of Luby-Rackoff ciphers*. PhD thesis, Massachusetts Institute of Technology.

Suh, G. E. and Devadas, S. (2007). Physical unclonable functions for device authentication and secret key generation. In *44th annual Design Automation Conference (DAC 2007)*, pages 9–14. ACM.

Tuyls, P., Schrijen, G., Škorić, B., van Geloven, J., Verhaegh, N., and Wolters, R. (2006). Read-proof hardware from protective coatings. *In 8th International Workshop on Cryptographic Hardware and Embedded Systems (CHES 2006)*, pages 369–383.