Proposal of Enhancement for Quartz Digital Signature

E. R. Andrade and R. Terada

***Abstract*— Today, we see a large dependence on systems developed with cryptography. Especially in terms of public key cryptosystems, which are widely used on the Internet. However, public key cryptography was threatened and new sources began to be investigated when Shor in 1997 developed a polynomial time algorithm for factoring integers and to compute the discrete logarithm with a quantum computer. In this context, Patarin proposed Hidden Field Equations (HFE), a trapdoor based on $\mathcal{MQ}$ ($\mathcal{M}$ultivariate $\mathcal{Q}$uadratic) and IP (Isomorphism of Polynomials) problems. Such problems are not affected by the Shor algorithm, moreover $\mathcal{MQ}$ Problem was proved by Patarin and Goubin to be NP-complete. Despite the basic HFE has been broken, there are variants that are secure, obtained by a generic modification. The Quartz – digital signature scheme based on HFEv-, with special choice of parameters – is a good example of this resistance to algebraic attacks aimed at the recovery of the private key, because even today it remains secure. Furthermore, it also generates short signatures. However, Joux and Martinet, based on axioms of Birthday Paradox Attack, proved that Quartz is malleable, showing that if the adversary has a valid pair (message, signature), he can get a second signature with 2^{50} computations and 2^{50} calls to the signing oracle, so that the estimated current security standards are at least 2^{112}. Thus, based on Quartz, we present a new digital signature scheme, achieving the adaptive chosen message attacks that make calls to the random oracle, with a security level estimated at 2^{112}. Our cryptosystem also provides an efficiency gain in signature verification algorithm and vector initializations that will be used for signing and verification algorithms. Furthermore we provide an implementation of Original Quartz and Enhanced Quartz in the Java programming language.**

***Keywords*— Post-Quantum Cryptography, $\mathcal{MQ}$ Problem, Digital Signature, Quartz, MPKC.**

I. INTRODUÇÃO

PODEMOS perceber que atualmente, seja conscientemente ou não, uma dependência dos sistemas desenvolvidos sob a seara da criptografia foi instaurada em todos nós. Principalmente no tocante dos sistemas criptográficos de chave pública, que são vastamente utilizados na Internet, incluindo-se aí os esquemas de assinatura digital.

No entanto, desde quando Shor em 1997 desenvolveu um algoritmo de tempo polinomial para fatorar inteiros e para calcular o logaritmo discreto num computador quântico [49] – computador este, proposto por Deutsch em 1985 [15] – a criptografia de chave pública se viu ameaçada e começou a investigar novas fontes de problemas para seus sistemas. Este alarde ocorrera porque, basicamente, os criptossistemas de chave pública usados na atualidade têm sua segurança baseada na intratabilidade dos problemas da fatoração de inteiros, no caso de sistemas RSA, e do logaritmo discreto, em sistemas ElGamal ou de Curvas Elípticas, e tal descoberta tornaria estes sistemas inseguros quando possuíssemos computadores quânticos com a capacidade adequada para implementarmos o algoritmo de Shor.

Acreditamos que a possibilidade de evolução dos computadores quânticos não deveria ser encarada como único fator para a obsolescência dos criptossistemas de chave pública atuais. Pois além de existir incontáveis estudos acerca da segurança destes problemas ditos clássicos, a capacidade computacional aumenta significativamente a cada década, e isto, sem dúvida, torna padrões outrora considerados seguros em inseguros.

Uma interessante proposta para enfrentarmos estes desafios é utilização de sistemas MPKC (acrônimo da nomenclatura em inglês que significa Criptossistema de Chave Pública Multivariável), que se apoiam no Problema $\mathcal{MQ}$(*$\mathcal{M}$ultivariate $\mathcal{Q}$uadratic*) para o desenvolvimento ou aprimoramento de sistemas criptográficos de chave pública seguros.

O Quartz é um esquema de assinatura digital baseado no HFEv-, com escolha especial de parâmetros. Sua versão original proposta por Patarin, Courtois e Goubin em 2001 [44] foi atualizada pelos mesmos autores logo em seguida [14], sendo que desde então adotamos esta última como versão original. Este esquema de assinatura foi submetido e aceito no NESSIE (*New European Schemes for Signatures, Integrity and Encryption*), um projeto de pesquisa desenvolvido com a *Information Societies Technology* (IST) *Programme of the European Commission* para identificar sistemas criptográficos seguros que forneçam – em sentido amplo – confidencialidade e integridade dos dados, além de autenticidade das entidades [38]. De acordo com os relatórios públicos do NESSIE, o principal trunfo deste esquema são suas assinaturas curtas (apenas 128 bits) e a fundamentação em um problema intratável até mesmo em computadores quânticos (o problema $\mathcal{MQ}$) [37].

Contudo, o Quartz não foi selecionado para figurar no portfólio final desse projeto de pesquisa. Isto porque – em linhas gerais – o cálculo de suas chaves secretas foi considerado muito lento (comparando com os demais esquemas submetidos) [37]; por possuir algumas divergências nas especificações de sua implementação (quando confrontado com o requerido pelo NESSIE) [20]; e também por possuir

E. R. Andrade, Laboratório de Arquitetura de Redes de Computadores (LARC), Escola Politécnica, Universidade de São Paulo (Poli-USP), São Paulo, SP, Brasil, ewe@ime.usp.br

R. Terada, Departamento de Ciência da Computação (DCC), Instituto de Matemática e Estatística, Universidade de São Paulo (IME-USP), São Paulo, SP, Brasil, rt@ime.usp.br

uma arquitetura maleável que permite ao adversário obter uma segunda assinatura, caso ele possua um par (mensagem, assinatura) válido, com uma quantidade de cálculos muito menor do que o solicitado pelo projeto [33].

Desta forma, o objetivo de nosso trabalho foi analisar o esquema de assinatura digital Quartz, apresentando, ao final deste estudo, um novo protocolo de assinatura digital baseado nele, porém, com um nível de segurança ainda maior e um algoritmo de verificação mais eficiente. Além disto, foi desenvolvida uma implementação de referência, tanto do modelo original quanto de nosso modelo proposto, para então ser analisada a viabilidade de nosso modelo através da estimativa de segurança e apreciação dos tempos obtidos durante os testes realizados a partir de nossa implementação.

A. Contribuições e organização do trabalho

As principais contribuições deste trabalho são: a apresentação de um novo protocolo de assinatura digital baseado no Quartz, logo, com assinaturas curtas e fundamentado em um problema intratável até mesmo em computadores quânticos; obtenção de um criptossistema resistente a ataques adaptativos que realizem chamadas ao oráculo aleatório, com um nível de segurança estimado em 2^{112}, contra os 2^{50} do protocolo original; constatação de que nosso aprimoramento irá testar até 4.096 vezes menos hipóteses de utilização da chave pública durante a verificação de assinatura, quando comparado com o Quartz Original; implementação do Quartz Original e do Quartz Aprimorado em uma linguagem de programação portável.

Para isto, organizamos este trabalho da seguinte forma. Na seção II, apresentamos as principais notações e definições utilizadas durante o desenvolvimento deste trabalho. Na seção III, descrevemos sucintamente o Problema $\boldsymbol{\mathcal{MQ}}$ e seu uso na criptografia de chave pública, apresentando seu esquema genérico de funcionamento. Neste ponto, também elencamos e descrevemos sucintamente as principais características dos modificadores genéricos aplicáveis as funções $\boldsymbol{\mathcal{MQ}}$ básicas. Na seção IV até a XI colocamos as principais contribuições de nosso trabalho. Nelas, revisamos o HFE e o Quartz Original, levantamos alguns aspectos referente a segurança do modelo original, apresentamos nossa proposta de aprimoramento, analisamos as modificações propostas, estimamos o impacto destas modificações na segurança, e ainda, apresentamos os tempos obtidos durante os testes realizados a partir de nossa implementação de referência. Por fim, na seção XII, apresentamos as considerações finais de nossa pesquisa e propomos novas direções para trabalhos futuros.

II. PRINCIPAIS NOTAÇÕES UTILIZADAS

Com intuito de facilitar a leitura, destacamos na TABELA I os principais parâmetros, bem como algumas definições e terminologias pertinentes ao Quartz. Além disto, frisamos que utilizaremos $\mathbb{F}$ ou $\mathbb{F}_q$ para indicar um Corpo Finito de ordem q, onde q possua característica p, para algum p primo, e $k \in \mathbb{N}$, tal que $q = p^k$. Quando utilizarmos n estaremos tratando sobre o número de variáveis do sistema de equações, e v definirá quantas destas variáveis são do tipo vinagre. Por sua vez, quando empregarmos m estaremos indicando a quantidade de equações utilizadas em nosso sistema, sendo que r denotará quantas destas equações foram removidas, quando for o caso. E ainda, por definição, temos que h representará o grau da extensão do Copo Finito $\mathbb{F}$, ou seja $h \stackrel{\text{def}}{=} n - v$ e $\mathbb{E} = \mathbb{F}_{q^h}$.

TABELA I. DEFINIÇÕES, NOTAÇÃO E TERMINOLOGIA PERTINENTES AO QUARTZ.

Símbolo	Significado
$[\lambda]_{p\to q}$	Dada uma cadeia de bits $\lambda = (\lambda_0, \dots, \lambda_t)$ e dois inteiros p e q tais que $0 \le p \le q \le t$, temos que $[\lambda]_{p\to q} = (\lambda_p, \lambda_{p+1}, \dots, \lambda_{q-1}, \lambda_q)$
$\|\|$	Concatenação
$\lambda\|\|\mu$	Se $\lambda = (\lambda_0, \dots, \lambda_t)$ e $\mu = (\mu_0, \dots, \mu_t)$ são duas cadeias de bits, então $\lambda\|\|\mu = (\lambda_0, \dots, \lambda_t, \mu_0, \dots, \mu_t)$
$p(x_1, \dots, x_n)$	Um polinômio de grau d com n variáveis sobre $\mathbb{F}$
$\mathcal{P} = (p_1, \dots, p_m)$	Um sistema de m polinômios de grau d com n variáveis sobre $\mathbb{F}$

III. CRIPTOSSISTEMAS DE CHAVE PÚBLICA MULTIVARIÁVEL

Uma das motivações (não a única) para a nossa pesquisa é o risco do comprometimento dos atuais sistemas criptográficos de chave pública, que basicamente são fundamentados no problema da fatoração de inteiros e do logaritmo discreto, no caso de computadores quânticos com capacidade de processamento adequada serem desenvolvidos. Dentre as classes de criptossistemas pós-quânticos, os MPKCs se destacam por, principalmente, possibilitar a criação de esquemas de assinatura digital com tamanho de assinatura reduzida [13], que é um objetivo implícito de nosso trabalho.

A. O problema $\boldsymbol{\mathcal{MQ}}$

Seja $\mathbb{F}$ um Corpo Finito de ordem q, $n \in \mathbb{N}$ o número de variáveis, $m \in \mathbb{N}$ o número de equações, $\mathcal{P} = (p_1, \dots, p_m)$ um sistema de m polinômios com grau d e n variáveis sobre $\mathbb{F}$. Temos que o Sistema de Equações Polinomiais Multivariáveis Simultâneas consiste em encontrar $x = (x_1, \dots, x_n) \in \mathbb{F}^n$ tal que $\mathcal{P}(x) = y$, sendo $y = (y_1, \dots, y_m) \in \mathbb{F}^m$, onde y é um vetor de dimensão m:

$$\mathcal{P} = \begin{cases} p_1(x_1, \dots, x_n) = y_1 \\ p_2(x_1, \dots, x_n) = y_2 \\ \vdots \\ p_m(x_1, \dots, x_n) = y_m \end{cases}$$

Assim, quando o grau de $\mathcal{P}$ é maior ou igual a 2, ou seja $d \ge 2$, chamamos, então, este sistema de equações polinomiais de Problema $\boldsymbol{\mathcal{MQ}}$.

O problema $\boldsymbol{\mathcal{MQ}}$ baseia-se no trabalho apresentado por Fraenkel e Yesha em 1979, onde os autores provaram que solucionar sistemas de equações polinomiais multivariáveis sobre $GF(2)$ é NP-difícil [27], sendo este resultado popularizado pelo livro de Garey e Johnson [28]. Entretanto, a demonstração de segurança desta primitiva sendo utilizada na

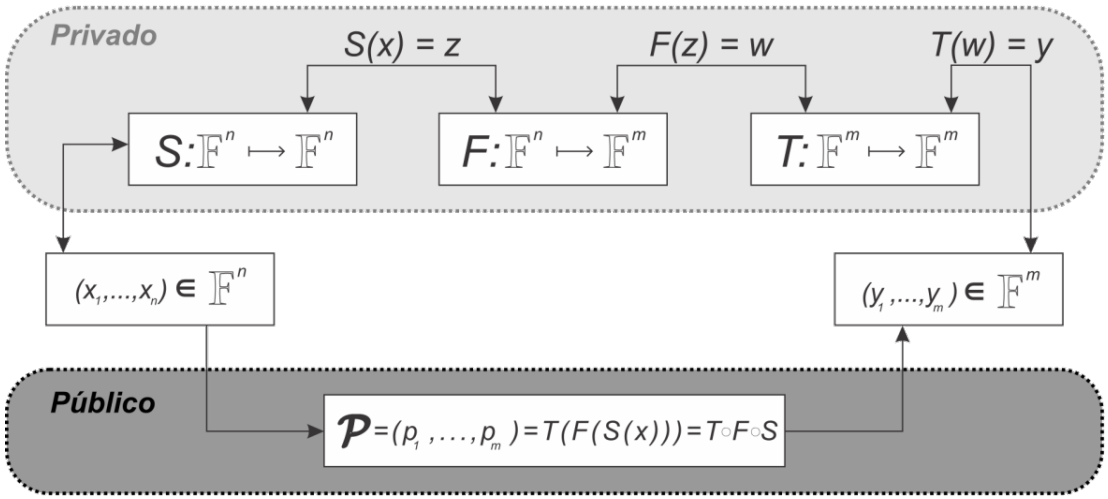

Figura 1. Modelo genérico de MPKC.

concepção de criptossistemas só foi apresentada em 1997. Neste novo trabalho, Patarin e Goubin demonstraram que a *trapdoor* criada a partir do problema $\mathcal{MQ}$ é NP-completo [45], não se conhecendo até hoje nenhum algoritmo, nem mesmo quântico, de tempo polinomial que possa resolver este problema [3], [39].

Precisamos destacar ainda, que ao estabelecer os parâmetros para instanciar uma função $\mathcal{MQ}$, devemos ter $n > m$ e $n \approx m$ para que o problema permaneça intratável. Isto porque, caso $n < m$, teríamos um sistema de equações superdefinido, onde existiriam mais equações do que variáveis, o que tornaria a sua resolução "fácil"' [10]. Por outro lado, caso $n \gg m$, teríamos um sistema de equações com muito mais variáveis do que equações, o que possibilitaria ao adversário utilizar o algoritmo proposto por Courtois *et al.* em 2002; sendo este algoritmo muito mais veloz do que a busca exaustiva [9].

B. Visão geral dos MPKCs

Sabemos que uma função construída a partir do problema $\mathcal{MQ}$ é uma função unidirecional [45], porém, para a construção de um criptossistema de chave pública é necessário que esta função seja também uma função alçapão (*trapdoor*). No entanto, Ding e Yang enfatizam, de forma simples e direta, que não é possível obter uma função deste tipo a partir de uma instância aleatória de função $\mathcal{MQ}$ [3]. Por isso, nos atuais MPKCs, a chave pública $\mathcal{P}$ é criada a partir da composição de duas transformações $S: \mathbb{F}^n \mapsto \mathbb{F}^n$, $T: \mathbb{F}^m \mapsto \mathbb{F}^m$ e um mapeamento central $F: \mathbb{F}^n \mapsto \mathbb{F}^m$, ou seja, $\mathcal{P} = T \circ F \circ S$, onde T, F e S são as chaves privadas [55]. Sendo este o modelo conceitual genérico adotado pelos criptossistemas MPKC, o qual pode ser vislumbrado na Figura 1.

O fato de $\mathcal{P} = T \circ F \circ S$, ou seja, $\mathcal{P}$ ser uma composição de outras funções, faz com que os atuais Criptossistemas de Chave Pública Multivariável não dependam exclusivamente do problema $\mathcal{MQ}$. Esta interdependência entre o problema $\mathcal{MQ}$ e IP (Isomorfismo de Polinômios) gera certa controvérsia, uma vez que mesmo acreditando-se que o problema IP seja difícil, nenhuma prova da intratabilidade deste problema foi formulada até o momento.

Contudo, ressaltamos que o modelo aqui exposto é apenas uma generalização das diversas $\mathcal{MQ}$-*trapdoors* existentes na atualidade, onde a principal diferença entre elas está, principalmente, no formato do mapeamento central F. Sendo que também existem algumas *trapdoors*, como é o caso da UOV (*Unbalanced Oil and Vinegar*), que utilizam apenas uma transformação afim para criar a chave pública $\mathcal{P}$, de modo que $\mathcal{P} = F \circ S$ [34]. Característica que utilizamos para propor o modelo Aprimorado do Quartz (veja Seção VII), pois apesar de fugir ligeiramente do modelo genérico, este tipo de composição é amplamente aceito, já que uma segunda transformação afim não adiciona segurança alguma ao sistema criptográfico [3], [36], uma vez que suas operações são apenas lineares; e a não realização desta transformação melhora a performance do processo de assinatura e geração de chaves.

C. Modificadores Genéricos

Com o passar dos anos, e também devido o aumento dos estudos acerca de MPKC, foi constatado que versões básicas das $\mathcal{MQ}$-*trapdoors* existentes na atualidade são inseguras [3],[55]. Contudo, para que todos estes criptossistemas não fossem descartados, modificadores genéricos (que, a grosso modo, são blocos construtores que alteram algumas estruturas das funções alçapão básicas) foram desenvolvidos para serem aplicados (pelo menos na teoria) em todas estas funções $\mathcal{MQ}$ básicas.

TABELA II. PRINCIPAIS MODIFICADORES GENÉRICOS E ALGUMAS DE SUAS CARACTERÍSTICAS.

Símbolo	Nome	Segurança	Ideia básica	Perda
-	Menos	seguro	descarta alguns polinômios	encriptação mais lenta
+	Mais	maioria sem efeito	adiciona polinômios	assinatura mais lenta
v	Vinagre	pouco mais seguro	variáveis extras são definidas	encriptação mais lenta
p	Pré-fixo ou Pós-fixo	em aberto	força algum $p_l = 0$	assinatura mais lenta
i	Pertubação Interna	em aberto	equivalente a p + v	tudo mais lento
f	Fixador	em aberto	usa algumas variáveis aleatórias	-
m	Mascaramento	em aberto	descarta algumas variáveis	-
s	Esparso	em aberto	usa polinômios esparsos	*speedup* mais lento

A TABELA II expressa de maneira sucinta – sem querer abordar todos os aspectos pertinentes a este tema –, os principais modificadores existentes, seguidos de suas características mais marcantes.

Vale ressaltar, também, que alguns destes modificadores mostraram-se mais eficientes para alguns esquemas do que para outros [55]. Além disto, assim como ocorrera nas versões básicas das $\mathcal{MQ}$-*trapdoors*, alguns modificadores genéricos como: Ramificação ($\perp$), Sub-Corpo ($/$) e Homogeneização(h), também foram considerados inseguros ou sem efeito [7], [24], [29], [41], [42], [54].

Desta forma, podemos pressupor que no momento da melhoria ou desenvolvimento de um criptossistema MPKC, há de se considerar a possibilidade de inserção do modificador genérico. Ponderando sobre qual é o ideal para aquele tipo de $\mathcal{MQ}$-*trapdoors*, considerando suas peculiaridades e contribuições para segurança. Mesmo não sendo esta uma tarefa trivial, que pode, inclusive, gerar vulnerabilidades em vez de melhorias.

IV. REVISANDO O QUARTZ

Nesta seção revisaremos o protocolo de assinatura digital Quartz, apresentado no NESSIE em 2001. Inicialmente, explanaremos sobre os aspectos gerais do HFE e exibiremos como é formada sua função de mapeamento central. Em seguida, explicaremos como funciona o algoritmo Quartz Original.

D. HFE (Hidden Field Equation)

Após ter quebrado a primeira *trapdoor* baseada em sistemas de equações multivariadas que se mostrou viável para os computadores da época, o MIA (*Matsumoto Imai Scheme A*) [41], Patarin – fundamentado nas ideias desta *trapdoor* considerada insegura – desenvolveu uma nova função alçapão denominada *Hidden Field Equations*, ou simplesmente HFE [42]. Esta nova *trapdoor* é uma generalização que modifica a função de mapeamento central F do MIA. Tal generalização tem como principal característica a troca dos monômios, empregados na *trapdoor* quebrada, por polinômios. Porém, apesar desta troca de monômios por polinômios, é mantido o conceito de utilizar uma extensão do corpo $\mathbb{F}_q$, comumente denotado por $\mathbb{E}$, tal que $\mathbb{E} = \mathbb{F}_{q^n}$ (onde q possua característica p, para algum p primo, e $k \in \mathbb{N}$, tal que $q = p^k$), juntamente com o corpo $\mathbb{F}_q$.

Deste modo, sejam: i e j números naturais; ξ_{ij}, ψ_i e μ elementos de $\mathbb{E}$; e θ_{ij}, σ_{ij} e γ_i números inteiros; a função do mapeamento central fica definida como [43]:

$$f(x) = \sum_{i,j}^{d} \xi_{ij} x^{q^{\theta_{ij}} + q^{\sigma_{ij}}} + \sum_{i}^{d} \psi_i x^{q^{\gamma_i}} + \mu \qquad (1)$$

$$onde \begin{cases} \xi_{ij} x^{q^{\theta_{ij}} + q^{\sigma_{ij}}} & são\ os\ termos\ quadráticos, \\ \psi_i x^{q^{\gamma_i}} & são\ os\ termos\ lineares, e \\ \mu & são\ os\ termos\ constantes \end{cases}$$

tal que $f(x)$ seja um polinômio em x sobre $\mathbb{E}_{q^n}$ com grau d, para $0 \leq \theta_{ij}, \sigma_{ij}, \gamma_i \leq d$.

Como $\mathbb{E}$ e $\mathbb{F}$ são isomórficos, podemos representar os elementos de $\mathbb{E} = \mathbb{F}_{q^n}$ numa n-tupla sobre $\mathbb{F}_q$, e a função (1) pode ser representada por polinômios com n variáveis $x_1, x_2, \ldots, x_n$ também sobre $\mathbb{F}_q$ [42].

E. Quartz Original

Como vimos anteriormente, o Quartz é um esquema de assinatura digital baseado no HFEv-. Neste esquema de assinatura, como os próprios modificadores genéricos já sugerem, algumas variáveis extra (chamadas de "variáveis vinagre") são adicionadas, e também, alguns "polinômios de perturbação" são inseridos no local dos polinômios removidos (em algumas fontes estes polinômios são chamados de polinômios secretos). Além disto, Patarin *et al.* destacam que os parâmetros escolhidos para o Quartz são cuidadosamente selecionados para melhorar sua segurança e impedir o funcionamento dos principais ataques conhecidos [44].

1) Parâmetros: Na versão original do Quartz temos definido que: $h = 103$, assim, a extensão do corpo utilizada pelo Quartz fica definida como $\mathbb{F}_{2^{103}} = \mathbb{E}$, mais precisamente, $\mathbb{E} = \mathbb{F}_2[X]/(X^{103} + X^9 + 1)$; $q = 2$; $d = 129$; $v = 4$; $r = 3$; $n = 107$ (pois $n \stackrel{\text{def}}{=} h + v$); $m = 100$ (pois $m \stackrel{\text{def}}{=} h - r$) [13], [14], [44]; e a função pública $\mathcal{P}$ – função *trapdoor* – é um mapeamento de 107 bits para 100 bits, ou seja $\mathbb{F}^{107} \mapsto \mathbb{F}^{100}$ [44].

2) Assinando Mensagens: Seja M uma mensagem representada por uma cadeia de bits, e S a assinatura obtida desta mensagem. Então, os procedimentos necessários à

obtenção de S devem ser realizados conforme segue:

1. Sejam M_0, M_1, M_2 e M_3 quatro cadeias de 160 bits definidas por: $M_0 = SHA\_1(M)$, $M_1 = SHA\_1(M_0||0)$, $M_2 = SHA\_1(M_0||1)$, $M_3 = SHA\_1(M_0||2)$.
2. Sejam H_1, H_2, H_3 e H_4 quatro cadeias de 100 bits definidas por: $H_1 = [M_1]_{0\to99}$, $H_2 = [M_1]_{100\to159}||[M_2]_{0\to39}$, $H_3 = [M_2]_{40\to139}$, $H_4 = [M_2]_{140\to159}||[M_3]_{0\to79}$.
3. Seja $\tilde{S}$ uma cadeia de 100 bits, tal que $\tilde{S}$ seja inicializada com 00 ... 0.
4. Para $i = 1$ até 4, faça:
   a. Calcule a cadeia de 100 bits $Y = H_i \oplus \tilde{S}$.
   b. Calcule a cadeia de 160 bits $W = SHA\_1(Y||\Delta)$.
   c. Obtenha a cadeia de 3 bits $R = [W]_{0\to2}$.
   d. Obtenha a cadeia de 4 bits $V = [W]_{3\to6}$.
   e. Calcule $B = \varphi(t^{-1}(Y||R))$.
   f. Considerando $F_V(Z) = B$ em Z sobre $\mathbb{E}$:
      i. Se a equação $F_V(Z) = B$ não tiver solução, troque W por $SHA\_1(W)$ e retornar ao passo 4c.
      ii. Neste passo a equação $F_V(Z) = B$ tem uma ou mais soluções em $\mathbb{E}$. Logo, temos que $A(1), A(2), \dots, A(\delta)$ são as soluções de $F_V(Z) = B$.
      iii. Se $F_V(Z) = B$ tiver apenas uma solução, defina $A = A(1)$. Caso contrário, aplique a função hash em cada uma das soluções, ou seja $I(j) = SHA\_1(A(j))$. Em seguida escolha o $A(j)$ que resulta no menor $I(j)$, considerando a ordenação *big-endian*.
   g. Calcule a cadeia de 107 bits $X = s^{-1}(\varphi^{-1}(A)||V)$.
   h. Defina um novo valor para $\tilde{S}$ como sendo $\tilde{S} = [X]_{0\to99}$.
   i. Obtenha a cadeia de 7 bits X_i definida por $X_i = [X]_{100\to106}$.
5. A assinatura S é a cadeia de 128 bits definida por $S = \tilde{S}||X_4||X_3||X_2||X_1$.

3) Verificando Assinatura: Dadas uma mensagem M, representada por uma cadeia de bits, e uma assinatura S, que neste caso é uma cadeia de 128 bits. Então, os procedimentos que seguem devem ser realizados para verificar se S é ou não uma assinatura válida para M.

1. Sejam M_0, M_1, M_2 e M_3 quatro cadeias de 160 bits definidas por: $M_0 = SHA\_1(M)$, $M_1 = SHA\_1(M_0||0)$, $M_2 = SHA\_1(M_0||1)$, $M_3 = SHA\_1(M_0||2)$.
2. Sejam H_1, H_2, H_3 e H_4 quatro cadeias de 100 bits definidas por: $H_1 = [M_1]_{0\to99}$, $H_2 = [M_1]_{100\to159}||[M_2]_{0\to39}$, $H_3 = [M_2]_{40\to139}$, $H_4 = [M_2]_{140\to159}||[M_3]_{0\to79}$.
3. Seja $\tilde{S}$ uma cadeia de 100 bits definida por $[S]_{0\to99}$.
4. Sejam X_4, X_3, X_2 e X_1 quatro cadeias de 7 bits definidas por: $X_4 = [S]_{100\to106}$, $X_3 = [S]_{107\to113}$, $X_4 = [S]_{114\to120}$, $X_4 = [S]_{121\to127}$.
5. Seja U uma cadeia de 100 bits, tal que U seja inicializada com $\tilde{S}$
6. Para $i = 4$ até 1, faça:
   a. Calcule a cadeia de 100 bits Y definida por $Y = G(U||X_i)$.
   b. Defina um novo valor para a cadeia de 100 bits U como sendo $U = Y \oplus H_i$.
7. Se U é igual à cadeia 00 ... 0, aceite a assinatura. Caso contrário, rejeite-a.

V. SOBRE A SEGURANÇA DO QUARTZ (HFEv-)

Existem diversos ataques capazes de recuperar a chave privada em criptossistemas desenvolvidos a partir da versão básica do HFE (ou seja, a *trapdoor* HFE sem modificadores aplicados a ela) [4], [11], [17], [19], [21], [22], [30], [32], [35]. Porém, até o momento, poucas criptoanálises foram capazes de sobrepujar a segurança adicionada pelos modificadores genéricos.

Ironicamente, uma das primeiras criptoanálises do HFE com modificadores que fora publicada afirmava decrementar a segurança desta *trapdoor* com, justamente, os modificadores "v" (vinagre), "-" (menos), ou os dois aplicados simultaneamente; além de também atacar sua versão básica [22]. Este trabalho foi desenvolvido por Faugère e Joux em 2003, e nele os autores afirmavam ser possível recuperar a chave privada do Quartz com um esforço muito menor do que 2^{80} triplo-DES através de um ataque algébrico viabilizado pelo algoritmo F5 [22], desenvolvido anteriormente por Faugère (frisando que o algoritmo F5 é baseado na teoria de bases de Gröbner [8]). Todavia, algum tempo após o desenvolvimento deste ataque, Courtois publicou uma versão estendida de [12] onde ele mostra que o ataque desenvolvido por Faugère e Joux possui informações imprecisas ou enganosas, que acabam tornando sua criptoanálise inválida para o HFEv- e Quartz [13]. Sobre este ataque, ainda vale ressaltar, que Wolf (em sua tese de doutorado) alertara sobre os argumentos utilizados por Faugère e Joux, afirmando que era necessário ter cautela antes de adotar tal ataque, uma vez ele ainda não havia sido analisado por outros pesquisadores [54].

Acreditamos que os estudos realizados por Courtois e Wolf ocorreram de maneira independente e em paralelo. Sendo assim, a suspeita de Wolf juntamente com as explicações de Courtois são suficientes para conjecturarmos que o Quartz (HFEv-) é resistente ao ataque algébrico que visa a recuperação da chave privada, desenvolvido por Faugère e Joux.

Posteriormente, visando o HFE com modificadores, alguns outros ataques capazes de recuperar a chave privada foram criados. Porém nenhum que atingisse o Quartz [3], [16], [36].

Lembramos que tentar recuperar a chave privada não é o único objetivo de um adversário que ataca esquemas de assinatura digital. Uma abordagem possível é tentar gerar uma assinatura válida (porém falsificada) sem poder determinar ou modificar a mensagem cuja assinatura foi forjada. Este é o nível de sucesso mais baixo para um adversário, porém, se alcançado, é suficiente para considerarmos um esquema de

assinatura inseguro.

Seguindo esta abordagem, Joux e Martinet – baseados em axiomas do Ataque pelo Paradoxo de Aniversário – provaram que o Quartz é maleável, demonstrando que caso o adversário possua um par (mensagem, assinatura) válido, ele conseguirá obter uma segunda assinatura com $2^{m/2}$ computações e $2^{m/2}$ chamadas ao oráculo de assinatura, com um método que consiste em encontrar a segunda pré-imagem sem se preocupar com a inversão da função pública G [33]. Assim, sob este cenário, estima-se que um adversário do Quartz consegue forjar uma segunda assinatura com somente 2^{50} computações e 2^{50} chamadas ao oráculo aleatório, logo muito inferior aos padrões segurança atuais que são de, no mínimo, 2^{112} [1].

Apesar disto, recentemente Sakumoto *et al.* apresentaram um trabalho sugerindo que os modelos usuais de prova de segurança não devem ser diretamente aplicados a *trapdoors* como o HFE e UOV [48]. Além disto, os autores desenvolveram um novo modelo de prova de segurança para esquemas de assinatura digital baseado em HFE e UOV, frisando que nenhum modelo deste tipo existia até o momento da publicação desse artigo. Neste novo modelo, juntamente com pequenas modificações também propostas neste trabalho, as assinaturas passam a ser uniformemente distribuídas [48], e com esta nova distribuição, Sakumoto *et al.* afirmam que os esquemas de assinatura baseados em HFE e UOV podem atingir um nível de segurança resistente a falsificação existencial através de um ataque adaptativo de mensagem escolhida, e fornecem um teorema para calcular a segurança provável destes esquemas modificados [48].

Assim, fundamentados nas criptoanálises aqui citadas e principalmente na nova prova de segurança e modificações desenvolvidas pro Sakumoto *et al.* em 2011, seguiremos propondo nosso aprimoramento para o Quartz nas seções subsequentes.

VI. A QUESTÃO DO SHA-1

Sabemos que algoritmos de assinatura e verificação são mais rápidos quando aplicados sobre o resultado de uma função hash (y) do que quando aplicado diretamente sobre sua entrada (x), isto porque y é relativamente muito mais curto que x [50]. Motivado por este atributo das funções hash, o Quartz emprega o SHA-1 em diversos pontos do seu algoritmo de assinatura e, consequentemente, no de verificação também. Mais precisamente, o uso do SHA-1 pode ser vislumbrado nos passos 1, 4.b, 4.f.i e 4.f.iii do algoritmo de assinatura, e no passo 1 do algoritmo de verificação.

Todavia, a cada ano que passa a resistência a colisões do SHA-1 tem sido consideravelmente reduzida. Por exemplo, em 2005, Wang *et al.* publicaram um algoritmo para colisões do SHA-1 reduzido a 58 iterações com a complexidade 2^{33}, sendo este o principal marco no declínio do SHA-1 [51].

Além disto, Joux publicou em 2004 um trabalho averiguando a segurança em funções hash iteradas (situação presente no passo 1 dos algoritmos de assinatura e verificação). Neste trabalho o autor demonstra que concatenando-se os resultados de funções hash a resistência a colisões é de apenas $\mathcal{O}(n2^{n/2})$ e não $\mathcal{O}(2^n)$ como esperava-se.

Desta forma, acreditamos ser latente a necessidade de atualização do Quartz (e qualquer outro esquema de assinatura digital que utilize o SHA-1) quanto ao emprego desta função de hash em suas rotinas.

VII. QUARTZ APRIMORADO

Agora que já estudamos os detalhes pertinentes ao Quartz, podemos iniciar a explicação acerca do nosso modelo aprimorado. Iniciaremos esta seção abordando quais foram os parâmetros escolhidos para o Quartz Aprimorado, pois a escolha de parâmetros além de definir o *trade-off* entre segurança e performance também garante a resistência contra as criptoanálises conhecidas, quando feita corretamente.

Em seguida, descreveremos os algoritmos de Geração de Chaves, Assinatura e Verificação. Nestes algoritmos será possível notar três grandes mudanças. A primeira delas está no fato de utilizarmos somente uma transformação afim no processo de assinatura das mensagens. Outra mudança está na substituição do SHA-1 pelo SHA-3 no momento de inicializar os vetores de bits que serão utilizados no processo de assinatura e verificação de assinaturas, bem como no momento de gerar as variáveis R e V do algoritmo de assinatura. A terceira grande mudança esta no fato de concatenarmos um *salt* Γ à mensagem M antes de empregarmos a função de hash nesta mensagem.

A. Parâmetros

Em nossa versão aprimorada do Quartz temos definido que: $h = 229$, assim, a extensão do corpo utilizada pelo Quartz Aprimorado fica definida como $\mathbb{F}_{2^{229}} = \mathbb{E}$, mais precisamente, $\mathbb{E} = \mathbb{F}_2[X]/(X^{229} + X^9 + X^6 + X^5 + X^2 + X + 1)$; $q = 2$; $d = 129$; $v = 2$; $r = 5$; $n = 231$ (pois $n \stackrel{\text{def}}{=} h + v$); $m = 224$ (pois $m \stackrel{\text{def}}{=} h - r$); e a função pública $\mathcal{P}$ – função *trapdoor* – é um mapeamento de 231 bits para 224 bits, ou seja $\mathbb{F}^{231} \mapsto \mathbb{F}^{224}$.

Temos definido, ainda, um parâmetro adicional g, onde g expressa o tamanho do *salt* aleatório Γ que será concatenado à mensagem antes dela servir como entrada para a função hash, ou seja, $g = |\Gamma|$. Lembramos que Sakumoto *et al.*, em seu novo modelo de prova, propuseram a utilização deste *salt* aleatório para uniformizar as assinaturas em esquemas de assinatura digital baseados no HFE [48], sendo estimado um tamanho aproximado de $\log(q_{assina}(q_{hash} + q_{assina}))$ bits para que o esquema de assinatura seja considerado seguro [48]. Sabemos que q_{hash} e q_{assina} correspondem, respectivamente, à quantidade de consultas aos oráculos de hash e assinatura; e que em provas de esquemas de assinatura digital normalmente são considerados $q_{assina} = 2^{30}$ e $q_{hash} = 2^{60}$ [2]. Assim, segue que $g = 96$.

B. Assinando Mensagens

Seja M uma mensagem representada por uma cadeia de bits, e S a assinatura obtida desta mensagem. Então, em nosso esquema aprimorado, os procedimentos necessários a obtenção de S devem ser realizados conforme segue:

1. Seja Γ uma cadeia de 96 bits, tal que $\Gamma \in_R \{0,1\}^{96}$.
2. Seja M_0 uma cadeia de 512 bits definida por $M_0 = SHA\_3(M|\Gamma)$.
3. Sejam H_1 e H_2 duas cadeias de 224 bits definidas por: $H_1 = [M_0]_{0\to223}$, $H_2 = [M_0]_{224\to447}$.
4. Seja $\tilde{S}$ uma cadeia de 224 bits, tal que $\tilde{S}$ seja inicializada com 00 ... 0.
5. Para $i = 1$ até 2, faça:
   a. Calcule a cadeia de 224 bits $Y = H_i \oplus \tilde{S}$.
   b. Calcule a cadeia de 512 bits $W = SHA\_3(Y||\Delta)$.
   c. Obtenha a cadeia de 5 bits $R = [W]_{0\to4}$.
   d. Obtenha a cadeia de 2 bits $V = [W]_{5\to6}$.
   e. Considerando $F_V(Z) = B$ em Z sobre $\mathbb{E}$:
      i. Se a equação $F_V(Z) = B$ não tiver solução, troque W por $SHA\_3(W)$ e retornar ao passo 5c.
      ii. Neste passo a equação $F_V(Z) = B$ tem uma ou mais soluções em $\mathbb{E}$. Logo, temos que $A(1), A(2), \dots, A(\delta)$ são as soluções de $F_V(Z) = B$.
      iii. Se $F_V(Z) = B$ tiver apenas uma solução, defina $A = A(1)$. Caso contrário, aplique a função hash em cada uma das soluções, ou seja $I(j) = SHA\_3(A(j))$. Em seguida escolha o $A(j)$ que resulta no menor $I(j)$, considerando a ordenação *big-endian*.
   f. Calcule a cadeia de 231 bits $X = s^{-1}(\varphi^{-1}(A)||V)$.
   g. Defina um novo valor para $\tilde{S}$ como sendo $\tilde{S} = [X]_{0\to223}$.
   h. Obtenha a cadeia de 7 bits X_i definida por $X_i = [X]_{224\to230}$.
6. A assinatura S é a cadeia de 334 bits definida por $S = \tilde{S}||X_2||X_1||\Gamma$.

C. Assinando Mensagens

Dadas uma mensagem M, representada por uma cadeia de bits, e uma assinatura S, que neste caso é uma cadeia de 334 bits. Então, no Quartz Aprimorado, os procedimentos que seguem devem ser realizados para verificar se S é ou não uma assinatura válida para M.

1. Seja $\tilde{S}$ uma cadeia de 224 bits definida por $\tilde{S} = [S]_{0\to223}$.
2. Sejam X_2 e X_1 duas cadeias de 7 bits definidas por: $X_2 = [S]_{224\to230}$, $X_1 = [S]_{231\to237}$.
3. Seja Γ uma cadeia de 96 bits definida por $\Gamma = [S]_{238\to334}$.
4. Seja M_0 uma cadeia de 512 bits definida por $M_0 = SHA\_3(M|\Gamma)$.
5. Sejam H_1 e H_2 duas cadeias de 224 bits definidas por: $H_1 = [M_0]_{0\to223}$, $H_2 = [M_0]_{224\to447}$.
6. Seja U uma cadeia de 224 bits, tal que U seja inicializada com $\tilde{S}$.
7. Para $i = 2$ até 1, faça:
   a. Calcule a cadeia de 224 bits Y definida por $Y = G(U||X_i)$.
   b. Defina um novo valor para a cadeia de 224 bits U como sendo $U = Y \oplus H_i$.
8. Se U é igual à cadeia 00 ... 0, aceite a assinatura. Caso contrário, rejeite-a.

VIII. ANÁLISE DA PROPOSTA

Para a análise de nosso protocolo Quartz Aprimorado vamos inicialmente explicar porque foram escolhidos os parâmetros elencados na Seção VII-A, para isto, descreveremos os benefícios de adotar tais parâmetros, justificando o impacto que tal escolha gera na segurança, abordando também a possível perda ou ganho de eficiência que tal modificação pode gerar, quando comparada ao esquema original. Explanaremos também, sobre a substituição da função de hash SHA-1 (utilizada no modelo original) pela função de hash SHA-3 (escolhida para nosso aprimoramento). E ainda, justificaremos porque é interessante adotarmos as modificações propostas por Sakumoto *et al.* em seu novo modelo de prova para esquemas de assinaturas baseados em HFE [48], mesmo sabendo que tais modificações acarretam um aumento de 96 bits no tamanho final da assinatura.

Como acreditamos ser dedutível que uma segunda rodada de operações lineares não adicione segurança alguma a um esquema baseado na intratabilidade de equações multivariáveis quadráticas, não nos preocuparemos em dar maiores detalhes sobre a não utilização de duas transformações afim em nosso aprimoramento. Sendo que o leitor pode consultar maiores detalhes sobre o tema em [3], [6], [16] e [34].

A. Escolha de Parâmetros

Sabemos que no Quartz, caso o adversário possua um par (mensagem, assinatura) válido, é possível que este adversário obtenha uma segunda assinatura válida com $2^{m/2}$ computações e $2^{m/}$ chamadas ao oráculo aleatório [33]. Como nosso aprimoramento não modifica a estrutura geral do Quartz, apenas a adapta; podemos deduzir que tal ataque também é válido para ele. Desta forma, temos que $m \geq 224$ para obtermos um nível de segurança de no mínimo 2^{112} (padrão mínimo exigido para sistemas criptográficos atuais [1]).

Todavia, ao definirmos nossos parâmetros não nos preocupamos somente com o tamanho de m. Isto porque ao estabelecermos parâmetros para instanciar uma função $\boldsymbol{\mathcal{MQ}}$, devemos ter $n > m$ e $n \approx m$ para que o problema permaneça intratável. Também existe o fato de Ding e Schmidt em 2005 terem demonstrado ser possível quebrar o HFEv quando $v = 1$ [18], assim, tomamos o cuidado de escolher um $v > 1$, porém não excessivamente maior. Além disto, escolhemos cuidadosamente o valor de h para que o mesmo fosse primo (fato que também ocorre no modelo original [14], [44]); isto porque até hoje não foi apresentada nenhuma criptoanálise que atinja MPKCs que utilizem extensões de corpos com característica igual a um número primo [13], [23], [36], [54].

Desta forma, lembramos que os parâmetros de nosso

modelo aprimorado são: $m = 224$, $n = 231$, $h = 229$, $v = 2$, $r = 5$, $q = 2$, $d = 129$ e $g = 96$.

Com estes parâmetros, constatamos dois inconvenientes em nosso aprimoramento. O primeiro deles está no aumento das chaves de nosso criptossistema, pois a chave privada aumenta de 3 Kbytes no Quartz Original para 8 Kbytes em nosso protocolo, sendo que a chave pública salta de 71 Kbytes para 739 Kbytes. O outro inconveniente de nossa escolha de parâmetros está na perda de eficiência dos algoritmos de Geração de Chaves e Assinatura (perda que pode ser constatada nas Tabelas IV e V da Seção XI). Em linhas gerais, acreditamos que tal ineficiência se deu devido ao aumento significativo na quantidade de objetos e instâncias a serem manipuladas por nossa implementação, sendo que melhorias como paralelismo ou instruções de máquina possam facilmente ser incorporadas a futuras implementações que visem à melhoria deste quesito.

Apesar desta perda de eficiência ocorrer, acreditamos que ela não seja tão grave, uma vez que a geração é efetuada apenas uma vez para cada usuário, dentro de um prazo longo de validade das chaves. Além disto, em determinados cenários, a ineficiência do processo de assinatura não é grave se supormos que o ato de assinar é realizado apenas uma vez para cada mensagem (ou cada Certificado Digital), enquanto a verificação é efetuada por muitos receptores da mensagem (ou do Certificado Digital).

Contudo, aliado ao expressivo ganho no nível de segurança (para maiores detalhes consulte a seção IX), a escolha de parâmetros do nosso esquema aprimorado proporcionou, também, uma melhoria no algoritmo de Verificação de Assinatura. Isto porque testaremos até 4.096 vezes menos hipóteses de utilização da chave pública no momento da resolução da função G, enquanto estamos verificando a validade de uma assinatura. Para sermos mais específicos, consideremos as seguintes características: (a) o mapeamento central F_V de $\mathbb{E}$ para $\mathbb{E}$ utilizado tanto na assinatura de mensagens quanto na composição da chave pública terá seu número de possibilidades para V variando conforme a quantidade de variáveis vinagre utilizadas pelo criptossistema, pois $(F_V)_{V\in\{0,1\}^v}$; (b) estas possibilidades representam um total de 2^v chaves públicas a serem geradas pelo algoritmo de Geração de Chaves; (c) como o Quartz (tanto o original quanto nosso modelo aprimorado) realiza operações iteradas para assinar as mensagens, e antecipadamente não podemos definir quais das 2^v possibilidades de variáveis vinagres foram utilizadas no processo de assinatura, temos que, seja K o número de iterações do algoritmo, então todas as $(2^v)^K$ combinações devem ser testadas para que o algoritmo de verificação negue uma assinatura falsa. Deste modo $(2^4)^4 = 2^{16}$ possibilidades devem ser testadas durante a resolução de G no Quartz Original e $(2^2)^2 = 2^4$ no Quartz Aprimorado, o que representa um intervalo de possibilidades $2^{12} = 4.096$ vezes menor em nosso aprimoramento.

B. Substituição do SHA-1 pelo SHA-3

Sabemos que algoritmos de assinatura (Assina) e verificação (Verifica) são mais rápidos quando aplicados sobre o resultado de uma função hash (y) do que quando aplicado diretamente sobre sua entrada (x), isto porque y é relativamente mais curto que x [50]. Logo, caso a função hash $H()$ não seja resistente a colisões, um adversário poderia obter uma mesma assinatura para duas mensagens distintas. Ou seja, dado um par de legíveis x_1 e x_2, onde $x_1 \neq x_2$ acarrete $H(x_1) = H(x_2)$, então $\sigma = \text{Assina}_{sk}\big(H(x_1)\big) = \text{Assina}_{sk}\big(H(x_2)\big)$. Caso isto ocorra, teríamos ainda que $\text{Verifica}_{pk}\left(M, \text{Assina}_{sk}\big(H(x_1)\big)\right) = \text{Verifica}_{pk}\left(M, \text{Assina}_{sk}\big(H(x_2)\big)\right) = 1$, ou seja, o algoritmo de verificação aceitaria as assinaturas, ferindo, também, os princípios de autenticidade e irretratabilidade, essenciais a esquemas de assinaturas digital.

Para ilustrar o quão prejudicial pode ser utilizar uma função hash inadequada, pensemos em nosso aprimoramento. Suponha que em vez do SHA-3 de 512 bits (que tem sua segurança estimada em 2^{256} [31]) utilizássemos o SHA-1. Desta forma, um adversário de nosso criptossistema poderia forjar uma segunda assinatura com aproximadamente 2^{33} operações [51] sem atacar diretamente o nosso protocolo.

Portanto, acreditamos que substituir a função SHA-1 pelo SHA-3 seja imprescindível para mantermos nosso esquema aprimorado dentro da segurança estimada. Sendo que, conforme pode ser vislumbrado nas TABELA IV e TABELA V, a adesão ao SHA-3 além de ajudar na segurança do Quartz Aprimorado também proporciona um ganho de eficiência no momento de inicializarmos os vetores. Tal melhoria ocorre em virtude da não realização de operações iteradas e também por não concatenar suas saídas para obter vetores do tamanho estabelecido pelo algoritmo de assinatura.

C. Modificação proposta por Sakumoto et al. em 2011

Inicialmente, no artigo que submetia o Quartz ao NESSIE [14], [44], seus autores apenas justificavam a dificuldade de quebrar seu protocolo, não fornecendo nenhuma prova matemática devido a falta de modelo apropriado para demonstrar a segurança de criptossistemas baseados em equações multivariáveis quadráticas. Alguns anos depois, o primeiro modelo de prova para MPKCs foi formalizado [13], neste trabalho, Courtois apresentava um modelo de prova que estimava a segurança de criptossistemas baseados nesta primitiva quando colocados sob ataques em que o adversário possua apenas a chave pública da vítima [13]. Apesar de ser uma prova matemática, este modelo é considerado "fraco", visto que em cenários reais o adversário facilmente pode obter (através de interceptações, por exemplo) muito mais informações do que somente a chave pública da vítima.

Felizmente, há pouco tempo Sakumoto *et al.* desenvolveram um novo modelo de prova, demonstrando que criptossistemas baseados nas *trapdoors* HFE e UOV podem ser existencialmente seguros contra ataques adaptativos de mensagem escolhida [48]. Este novo modelo de prova é considerado "forte", pois dá amplos poderes ao adversário e exige o seu nível de sucesso mais baixo. No entanto, para utilizarmos este modelo na demonstração de segurança de um criptossistema baseado no HFE, como é o caso do Quartz

Aprimorado, necessitamos concatenar um *salt* a mensagem antes de empregarmos a função de hash nesta mesma mensagem afim de obtermos assinaturas uniformemente distribuídas [48]. Esta modificação acarreta um aumento no tamanho final da assinatura. Em nosso modelo aprimorado este aumento é de 96 bits, porém, mesmo com este aumento no tamanho final da assinatura, consideramos viável a adesão desta modificação já que nosso esquema aprimorado poderá ser provado como sendo "fortemente infalsificável" em vez de somente "infalsificável", como ocorre no Quartz Original.

IX. ESTIMATIVA DE SEGURANÇA

Nesta seção, iremos inicialmente olhar para a probabilidade de recuperar a chave privada do Quartz (HFEv-) através do melhor ataque conhecido na atualidade. Em seguida, como a modificação proposta por Sakumoto *et al.* foi aderida por nosso aprimoramento, buscaremos determinar a segurança exata de nosso criptossistema de acordo com o novo modelo de prova proposto pelos mesmos autores. Por fim, visando demonstrar que o ataque de Joux e Martinet é computacionalmente inviável em nosso aprimoramento, calcularemos quantas computações e quantas chamadas ao oráculo aleatório serão necessárias para que um adversário derive uma segunda assinatura caso ele possua um par (mensagem, assinatura) válido.

A. Melhor ataque ao Quartz (HFEv-)

Discutimos anteriormente na Seção V que até o momento não foi desenvolvido nenhum ataque ao Quartz capaz de recuperar a chave privada (inverter a função G) com um esforço menor do que o Ataque por Força Bruta, ou seja, a *trapdoor* HFEv- permanece segura [3], [16], [36].

Além disto, em 2010, utilizando Unidades de Processamento Gráfico (GPUs – *Graphics Processing Unit*) Bouillaguet *et al.* apresentaram um algoritmo para resolução de equações polinomiais em $\mathbb{F}_2$ que pode ser empregado para solucionar qualquer instância de problema $\boldsymbol{MQ}$ onde, evidentemente, $q = 2$. Neste algoritmo, em vez de utilizarem bases de Gröbner (principalmente devido a necessidade exponencial de memória para sua implementação) Bouillaguet *et al.* resolveram utilizar uma abordagem diferente, baseando-se no algoritmo padrão de Busca Exaustiva [5]. Com este novo método os autores deste trabalho demonstraram ser possível encontrar todos os zeros de um polinômio de grau d com n variáveis, em corpos $\mathbb{F}_2$, efetuando apenas $d.2^n$ operações binárias [5].

Como este é o algoritmo para resolução de equações polinomiais através da Busca Exaustiva mais rápido da atualidade, então, podemos conjecturar que um adversário conseguirá inverter a função pública G com uma probabilidade estimada em $\epsilon' \leq 1/(d.2^n)$. Desta forma, com os parâmetros adotados em nossa proposta de aprimoramento, temos que $\epsilon' \leq 1/(129.2^{231}) \approx 2^{-238}$.

B. Estimativa de segurança segundo Sakumoto et al.

No apêndice A de [48], onde consta a demonstração dos Teoremas formulados por Sakumoto *et al.*, os autores mostram que a probabilidade do algoritmo de inversão, simulado pelo oráculo aleatório, encontrar a inversa da assinatura S utilizando chamadas a este oráculo aleatório e a chave pública G é de aproximadamente $\epsilon(1 - (q_{hash} + q_{assina})q_{assina}2^{-g})/(q_{hash} + q_{assina} + 1)$. Onde q_{hash} e q_{assina} correspondem, respectivamente, a quantidade de consultas aos oráculos de hash e assinatura. E g corresponde ao tamanho do *salt* aleatório inserido na inicialização de vetores do algoritmo proposto, lembrando que g = $\log(q_{assina}(q_{hash} + q_{assina}))$-bits.

Como, em provas de esquemas de assinatura digital normalmente são considerados $q_{assina} = 2^{30}$ e $q_{hash} = 2^{60}$ [2]. Desta forma, obtemos $\epsilon = \epsilon'(q_{hash} + q_{assina} + 1)/(1 - (q_{hash} + q_{assina})q_{assina}2^{-g}) \approx \epsilon'.2^{60}$ [48].

Assim, utilizando este Teorema que compõe o novo modelo de prova de segurança proposto por Sakumoto *et al.* e o ϵ' calculado anteriormente, temos que a probabilidade de um adversário recuperar a chave privada do Quartz Aprimorado utilizando um oráculo aleatório é de $\epsilon \approx 2^{-178}$.

C. Estimativa de esforço para o ataque de Joux e Martinet

Joux e Martinet em 2003 desenvolveram um poderoso ataque ao Quartz; neste trabalho, os autores – baseados em axiomas do Ataque pelo Paradoxo de Aniversário – provaram que este criptossistema é maleável, demonstrando que caso o adversário possua um par (mensagem, assinatura) válido, ele conseguirá obter uma segunda assinatura com $2^{m/2}$ computações e $2^{m/2}$ chamadas ao oráculo de assinatura, com um método que consiste em encontrar a segunda pré-imagem sem se preocupar com a inversão da função pública G [33].

Desta forma, com os parâmetros adotados em nossa proposta de aprimoramento, temos que através deste ataque o adversário terá que realizar $2^{224/2} = 2^{112}$ computações e $2^{224/2} = 2^{112}$ chamadas ao oráculo de assinatura.

Portanto, para recuperar a chave privada do Quartz Aprimorado será necessário um esforço maior do que 2^{178}, e para derivar uma segunda assinatura através do ataque de Joux e Martinet o adversário terá que efetuar mais de 2^{112} operações e chamadas ao oráculo de assinatura.

X. COMPARANDO O QUARTZ APRIMORADO COM OUTROS PROTOCOLOS

Na TABELA III, listamos o tamanho das assinaturas de alguns outros esquemas baseados no problema $\boldsymbol{MQ}$. Além disto, para também haver uma comparação com esquemas de assinatura mais convencionais e já padronizados, incluímos o comprimento das assinaturas do ECDSA e do RSA. Frisamos que as informações listados na referida tabela tem como referência um nível de segurança de aproximadamente 2^{100}, sendo tal segurança calculada de acordo com suas fontes, que também encontram-se listadas na TABELA III.

TABELA III. TAMANHO DAS ASSINATURAS DE ALGUNS CRIPTOSSISTEMAS.

	Criptossistema	q	d	m	n	Tamanho da Assinatura (em bits)	Referência
Pós-Quântico	CyclicUOV	256	256	77	77	624	[46]
	Rainbow	16	30	58	58	352	[47]
	NC-Rainbow	256	17	26	26	672	[56]
	CyclicRainbow	256	17	26	26	344	[46]
	Quartz Aprimorado	2	129	224	231	334	Nosso
Quântico	ECDSA					400	[40]
	RSA					2048	[1]

Compreendemos que o tamanho da assinatura não é a única métrica a ser avaliada no momento da adesão de um esquema de assinatura digital. Porém, entendemos que este é um quesito importante para a economia de tráfego de rede, além de ser um ponto extremamente forte de nosso aprimoramento.

XI. IMPLEMENTAÇÃO DE REFERÊNCIA E TESTES

Nossa implementação tem como objetivo comparar o tempo gasto pelo Quartz Original e pelo Quartz Aprimorado durante a inicialização de vetores, geração de chaves, assinatura de mensagens, verificação de assinatura verdadeira e a verificação de uma assinatura falsa.

Para não realizarmos muito re-trabalho, baseamos nossa implementação no Projeto QuartzLight, feito em Java e que tem como autor Christopher Wolf [53]. Este projeto implementa uma versão mais "frágil" do Quartz [52], porém, mesmo com esta vulnerabilidade, é interessante para nós devido seus diversos métodos e classes que podem ser re-utilizados para fatorarmos polinômios sobre Corpos Finitos e executarmos operações sobre $GF(2^n)$ e matrizes binárias. Além disto, utilizamos implementações em Python das funções de hash SHA-1 [25] e SHA-3 [26], também para não temos uma carga de re-trabalho excessiva.

Lembrando que o código fonte desta implementação está disponível em nossa página pessoal (http://www.ime.usp.br/~ewe/QuartzAprimorado/) para que o mesmo possa ser baixado, alterado e/ou executado conforme a necessidade e interesse de cada usuário.

A. Testes realizados

Utilizando nossa implementação do Quartz em sua versão original e também em sua versão aprimorada – proposta por este trabalho –, realizamos os testes necessários para a coleta dos tempos de execução dos algoritmos de geração de chaves, inicialização de vetores, assinatura e verificação (tanto de assinaturas legítimas quanto de assinaturas falsas).

Para isso, utilizamos dois computadores distintos. Sendo eles:

Brucutu: processador Intel Xeon E5645 de 2,4 GHz × 24, com 128 GB de memória RAM, utilizando o Sistema Operacional Linux Debian 7.0 (wheezy), OpenJDK 1.6.0_27 IcedTea e Python 2.7.3;

PC: processador Intel Core i7-2670QM de 2,2 GHz, com 8 GB de memória RAM, utilizando o Sistema Operacional Linux Ubuntu 12.10 (quantal), Java 1.7.0_25 da Oracle e Python 2.7.3.

Nesses computadores, rodamos os testes em apenas uma linha de execução (*thread*) e efetuamos um total de 1000 repetições para cada um dos quesitos analisados, coletando o tempo de barreira (horário em que a função encerrou seu processamento menos o horário inicial da chamada da função) em cada uma destas repetições.

B. Tempos obtidos nos testes

Apresentamos nas TABELA IV e TABELA V o intervalo, a média e o desvio padrão de tempo gasto pelo Quartz Original e pelo Quartz Aprimorado.

Acreditamos que a discussão levantada na Seção VIII – durante a análise da proposta de nosso aprimoramento – seja suficiente para justificar os tempos obtidos em nossos testes. Ou seja, a substituição do SHA-1 pelo SHA-3 além de ajudar na segurança do Quartz Aprimorado também proporcionou um ganho de eficiência de aproximadamente 75% no momento da Inicialização dos Vetores; sendo que tal melhoria ocorreu principalmente devido a não realização de operações iteradas e também por não concatenar suas saídas para obter vetores do tamanho estabelecido pelo algoritmo de assinatura.

Além disso, observamos que os algoritmos de Geração de Chaves e Assinatura têm sua eficiência prejudicada devido a nossa escolha de parâmetros (cerca de 360% acrescido no tempo de Geração de Chaves e 240% na Assinatura), isto porque tal escolha ocasionou um aumento significativo na quantidade de objetos e instâncias a serem manipuladas por nossa implementação.

TABELA IV. TEMPOS OBTIDOS DURANTE A REALIZAÇÃO DOS TESTES NO BRUCUTU.

			Quartz Original	Quartz Aprimorado
Inicialização dos Vetores	**SHA-1**	**Média (ms)**	158	-
		Desvio Padrão (ms)	16	-
		Intervalo (ms)	121 - 236	-
	SHA-3	**Média (ms)**	-	40
		Desvio Padrão (ms)	-	7
		Intervalo (ms)	-	34 - 57
Geração de Chaves		**Média (s)**	16,9	75,1
		Desvio Padrão (s)	0,2	0,4
		Intervalo (s)	16,5 - 17,7	74,2 - 77,8
Assinatura		**Média (s)**	5,2	19,1
		Desvio Padrão (s)	0,7	0,2
		Intervalo (s)	4,4 - 27,2	18,9 - 20,0
Verificação de Assinatura		**Média (ms)**	3814	18
		Desvio Padrão (ms)	233	3
		Intervalo (ms)	4 - 3927	17 - 40
Verificação de Assinatura Falsa		**Média (ms)**	60074	180
		Desvio Padrão (ms)	959	14
		Intervalo (ms)	52067 - 62258	159 - 194

TABELA V. TEMPOS OBTIDOS DURANTE A REALIZAÇÃO DOS TESTES NO PC.

			Quartz Original	Quartz Aprimorado
Inicialização dos Vetores	**SHA-1**	**Média (ms)**	62	-
		Desvio Padrão (ms)	15	-
		Intervalo (ms)	47 - 130	-
	SHA-3	**Média (ms)**	-	15
		Desvio Padrão (ms)	-	4
		Intervalo (ms)	-	12 - 44
Geração de Chaves		**Média (s)**	18,5	87,0
		Desvio Padrão (s)	1,7	10,4
		Intervalo (s)	15,6 - 26,8	72,2 - 108,3
Assinatura		**Média (s)**	5,4	16,6
		Desvio Padrão (s)	5,4	2,9
		Intervalo (s)	4,3 - 169,2	16,5 - 25,6
Verificação de Assinatura		**Média (ms)**	164	35
		Desvio Padrão (ms)	89	4
		Intervalo (ms)	136 - 2447	33 - 53
Verificação de Assinatura Falsa		**Média (ms)**	43248	99
		Desvio Padrão (ms)	3488	14
		Intervalo (ms)	36197 - 54591	96 - 145

Contudo, a Verificação de Assinatura de nosso aprimoramento mostrou-se significativamente melhor do que a do modelo original, sendo possível observar um ganho de $99{,}5\,\%$ (no Brucutu) e cerca de $78\,\%$ (no PC) durante a verificação de assinaturas legítimas e aproximadamente $99{,}7\,\%$ quando trata-se de assinaturas falsas; lembrando que no Quartz Aprimorado, durante a resolução da função G, são testadas até 4.096 vezes menos hipóteses de utilização da chave pública do que no Quartz Original.

Frisa-se que os tempos expostos acima foram obtidos a partir de nossa implementação de referência: feita em Java, sem paralelismo ou qualquer conjunto de instruções avançadas. Ou seja, todas as alterações no desempenho ocorreram em virtude dos parâmetros escolhidos e do novo design proposto em nosso aprimoramento.

XII. CONSIDERAÇÕES FINAIS

Nossa principal contribuição nesta pesquisa foi à apresentação de um novo protocolo de assinatura digital, baseado em sistemas polinomiais multivariados quadráticos. Como resultado, obtivemos um criptossistema com um nível de segurança estimado em 2^{112}, contra os 2^{50} do protocolo original. Nossa proposta apresenta, ainda, um ganho de eficiência na inicialização dos vetores que serão utilizados pelo algoritmo de assinatura; para ser mais específico, através de nossos testes – realizados a partir de nossa implementação de referência – constatamos um ganho de aproximadamente 75%. Além disto, mostramos que no Quartz Aprimorado, durante a resolução da função G, testaremos até 4.096 vezes menos hipóteses de utilização da chave pública, quando comparado com o Quartz Original.

Todavia, observamos que devido os parâmetros escolhidos para nosso criptossistema, houve um aumento significativo no tamanho das chaves, fato que também acarretou uma perda de eficiência nos algoritmos de geração de chaves e assinatura.

Em virtude do tamanho das chaves de nosso aprimoramento, acreditamos que uma possível extensão para nosso trabalho seria pesquisar uma maneira de reluzi-las. Outra possível extensão de nosso trabalho seria buscar uma prova de segurança mais eficiente (*tight*) no modelo do

oráculo aleatório para protocolos baseados na *trapdoor* HFE. Isto porque, conforme vimos durante a estimativa de segurança do Quartz Aprimorado, com o modelo de prova proposto por Sakumoto *et al.* [48] perdemos em média 60 bits de segurança.

AGRADECIMENTOS

Agradecemos à CAPES pelo apoio financeiro concedido.

REFERÊNCIAS

[1] Elaine Barker and Allen Roginsky. NIST Special Publication 800-131a - Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths. Technical report, National Institute of Standards and Technology, NIST, U.S. Department of Commerce, Washington DC. http://csrc.nist.gov/publications/nistpubs/800-131A/sp800-131A.pdf, 2011. Último acesso em 09/07/2013.

[2] Mihir Bellare and Phillip Rogaway. The exact security of digital signatures: How to sign with RSA and Rabin. In U. Maurer, editor, *Advances in Cryptology - EUROCRYPT 96 Proceedings*, volume 1070 of *Lecture Notes in Computer Science*, pages 399–416. Springer-Verlag, 1996.

[3] Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen, editors. *Post-Quantum Cryptography*. Springer, 2009.

[4] Luk Bettale, Jean-Charles Faugère, and Ludovic Perret. Cryptanalysis of HFE, multi-HFE and variants for odd and even characteristic. *Designs, Codes and Cryptography*, pages 1–52, 2012.

[5] Charles Bouillaguet, Hsieh-Chung Chen, Chen-Mou Cheng, Tung Chou, Ruben Niederhagen, Adi Shamir, and Bo-Yin Yang. Fast Exhaustive Search for Polynomial Systems in $\mathbb{F}_2$. In Stefan Mangard and François-Xavier Standaert, editors, *Cryptographic Hardware and Embedded Systems, CHES 2010*, volume 6225 of *Lecture Notes in Computer Science*, pages 203–218. Springer Berlin Heidelberg, 2010.

[6] Charles Bouillaguet, Jean-Charles Faugère, Pierre-Alain Fouque, and Ludovic Perret. Practical Cryptanalysis of the Identification Scheme Based on the Isomorphism of Polynomial with One Secret Problem. In Dario Catalano, Nelly Fazio, Rosario Gennaro, and Antonio Nicolosi, editors, *Public Key Cryptography - PKC 2011*, volume 6571 of *Lecture Notes in Computer Science*, pages 473–493. Springer Berlin Heidelberg, 2011.

[7] An Braeken, Christopher Wolf, and Bart Preneel. A study of the security of Unbalanced Oil and Vinegar signature schemes. Cryptology ePrint Archive, Report 2004/222. http://eprint.iacr.org/2004/222, 2004. Último acesso em 11/06/2013.

[8] Jean charles Faugère. A new efficient algorithm for computing Gröbner Bases without reduction to zero (F5). In *ISSAC 02: Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation*, pages 75–83, 2002.

[9] Nicolas Courtois, Louis Goubin, Willi Meier, and Jean-Daniel Tacier. Solving underdefined systems of multivariate quadratic equations. In David Naccache and Pascal Paillier, editors, *Public Key Cryptography*, volume 2274 of *Lecture Notes in Computer Science*, pages 211–227. Springer Berlin Heidelberg, 2002.

[10] Nicolas Courtois, Alexander Klimov, Jacques Patarin, and Adi Shamir. Efficient algorithms for solving overdefined systems of multivariate polynomial equations. In Bart Preneel, editor, Advances in *Cryptology - EUROCRYPT 2000*, volume 1807 of *Lecture Notes in Computer Science*, pages 392–407. Springer Berlin Heidelberg, 2000.

[11] Nicolas T. Courtois. The security of Hidden Field Equations (HFE). In David Naccache, editor, *Topics in Cryptology - CT-RSA 2001*, volume 2020 of *Lecture Notes in Computer Science*, pages 266–281. Springer Berlin Heidelberg, 2001.

[12] Nicolas T. Courtois. Generic Attacks and the Security of Quartz. In YvoG. Desmedt, editor, *Public Key Cryptography - PKC 2003*, volume 2567 of *Lecture Notes in Computer Science*, pages 351–364. Springer Berlin Heidelberg, 2002.

[13] Nicolas T. Courtois. Short signatures, provable security, generic attacks and computational security of multivariate polynomial schemes such as HFE, QUARTZ and SFLASH. Cryptology ePrint Archive, Report 2004/143. http://eprint.iacr.org/2004/143, 2004. Versão extendida e revista do artigo *Generic Attacks and the Security of Quartz* publicado no PKC 2003. Último acesso em 12/06/2013.

[14] Nicolas T. Courtois, Louis Goubin, and Jacques Patarin. Quartz, na asymmetric signature scheme for short signatures on PC. Primitive specification and supporting documentation (second revised version), 2001.

[15] David Deutsch. Quantum theory, the Church-Turing principle and the universal quantum computer. *Proceedings of the Royal Society of London Ser. A*, A400:97–117, 1985.

[16] Jintai Ding, Jason E. Gower, and Dieter Schmidt. *Multivariate public key cryptosystems*, volume 25 of *Advances in information security*. Springer, 2006.

[17] Jintai Ding and TimothyJ. Hodges. Inverting HFE systems is quasi-polynomial for all fields. In Phillip Rogaway, editor, *Advances in Cryptology - CRYPTO 2011*, volume 6841 of *Lecture Notes in Computer Science*, pages 724–742. Springer Berlin Heidelberg, 2011.

[18] Jintai Ding and Dieter Schmidt. Cryptanalysis of HFEv and Internal Perturbation of HFE. In Serge Vaudenay, editor, *Public Key Cryptography - PKC 2005*, volume 3386 of *Lecture Notes in Computer Science*, pages 288–301. Springer Berlin Heidelberg, 2005.

[19] Jintai Ding, Dieter Schmidt, and Fabian Werner. Algebraic attack on HFE revisited. In Tzong-Chen Wu, Chin-Laung Lei, Vincent Rijmen, and Der-Tsai Lee, editors, *Information Security*, volume 5222 of *Lecture Notes in Computer Science*, pages 215–227. Springer Berlin Heidelberg, 2008.

[20] Emmanuelle Dottax and École Normale Supérieure. Tweak reviews: ES-IGN, RSA-PSS, QUARTZ and SFLASH. NES/DOC/ENS/WP1/018/1. Technical report, Commission of the European Communities, out 2002. Último acesso em 04/07/2013.

[21] Jean-Charles Faugère. Algebraic cryptanalysis of HFE using gröbner bases, February 2003.

[22] Jean-Charles Faugère and Antoine Joux. Algebraic cryptanalysis of Hidden Field Equation (HFE) cryptosystems using gröbner bases. In Dan Boneh, editor, *Advances in Cryptology - CRYPTO 2003*, volume 2729 of *Lecture Notes in Computer Science*, pages 44–60. Springer Berlin Heidelberg, 2003.

[23] Adam Thomas Feldmann. A survey of attacks on Multivariate Cryptosystems. Master's thesis, University of Waterloo, Ontario, Canada, 2005.

[24] Patrick Felke. On the Affine Transformations of HFE-Cryptosystems and Systems with Branches. Cryptology ePrint Archive, Report 2004/367. http://eprint.iacr.org/2004/367, 2004. Último acesso em 03/07/2013.

[25] Python Software Foundation. 14.1. hashlib – Secure hashes and message digests. https://docs.python.org/2/library/hashlib.html, 2015. Último acesso em 31/01/2015.

[26] Python Software Foundation. pysha3 0.3. https://pypi.python.org/pypi/pysha3/, 2015. Último acesso em 31/01/2015.

[27] A.S. Fraenkel and Y. Yesha. Complexity of problems in games, graphs and algebraic equations. *Discrete Applied Mathematics*, 1(1–2):15–30, September 1979.

[28] Michael R. A. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. A Series of Books in the Mathematical Sciences. W. H. Freeman, 1979.

[29] Henri Gilbert and Marine Minier. Cryptanalysis of SFLASH. In LarsR. Knudsen, editor, *Advances in Cryptology - EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 288–298. Springer Berlin Heidelberg, 2002.

[30] Louis Granboulan, Antoine Joux, and Jacques Stern. Inverting HFE is Quasipolynomial. In Cynthia Dwork, editor, *Advances in Cryptology - CRYPTO 2006*, volume 4117 of *Lecture Notes in Computer Science*, pages 345–356. Springer Berlin Heidelberg, 2006.

[31] Shu jen Chang, Ray Perlner, William E. Burr, Meltem Sönmez Turan, John M. Kelsey, Souradyuti Paul, and Lawrence E. Bassham. NIST Interagency or Internal Reports 7896: Third-Round Report of the SHA-3 Cryptographic Hash Algorithm Competition. Technical report, National Institute of Standards and Technology, NIST, U.S. Department of Commerce, Washington DC. http://nvlpubs.nist.gov/nistpubs/ir/2012/NIST.IR.7896.pdf, 2012. Último acesso em 09/07/2013.

[32] Xin Jiang, Jintai Ding, and Lei Hu. Kipnis-shamir attack on HFE revisited. In Dingyi Pei, Moti Yung, Dongdai Lin, and Chuankun Wu, editors, *Information Security and Cryptology*, volume 4990 of *Lecture*

Notes in Computer Science, pages 399–411. Springer Berlin Heidelberg, 2008.

[33] Antoine Joux and Gwenaëlle Martinet. Some weaknesses in Quartz Signature Scheme. NES/DOC/ENS/WP5/026/1. Technical report, Commission of the European Communities, jan 2003. Último acesso em 12/06/2013.

[34] Aviad Kipnis, Jacques Patarin, and Louis Goubin. Unbalanced Oil and Vinegar signature schemes. In Jacques Stern, editor, *Advances in Cryptology - EUROCRYPT 99*, volume 1592 of *Lecture Notes in Computer Science*, pages 206–222. Springer Berlin Heidelberg, 1999.

[35] Aviad Kipnis and Adi Shamir. Cryptanalysis of the HFE Public Key Cryptosystem by Relinearization. In *CRYPTO 99: Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology*, pages 19–30, London, UK, 1999. Springer-Verlag.

[36] Dongdai Lin, Jean-Charles Faugère, Ludovic Perret, and Tianze Wang. On enumeration of polynomial equivalence classes and their application to MPKC. Cryptology ePrint Archive, Report 2011/055. http://eprint.iacr.org/2011/055, 2011. Último acesso em 29/06/2013.

[37] Gwenaëlle Martinet and École Normale Supérieure. QUARTZ, FLASH and SFLASH. NES/DOC/ENS/WP3/006/2. Technical report, Commission of the European Communities, mar 2001. Último acesso em 14/06/2013.

[38] NESSIE. Final report of European project IST-1999-12324: New European Schemes for Signatures, Integrity, and Encryption (NESSIE), (Abril de 2004). https://www.cosic.esat.kuleuven.be/nessie/Bookv015.pdf. Technical report, Commission of the European Communities, Abril 2004. Último acesso em 10/06/2013.

[39] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

[40] NIST. FIPS 186-3: Digital Signature Standard (DSS). Technical report, National Institute of Standards and Technology, NIST, U.S. Department of Commerce, Washington DC. http://csrc.nist.gov/publications/fips/fips186-3/fips_186-3.pdf, 2009. Último acesso em 16/07/2013.

[41] Jacques Patarin. Cryptanalysis of the Matsumoto and Imai Public Key Scheme of Eurocrypt'88. In Don Coppersmith, editor, *Advances in Cryptology - CRYPT0 95*, volume 963 of *Lecture Notes in Computer Science*, chapter 20, pages 248–261. Springer Berlin / Heidelberg, Berlin, Heidelberg, July 1995.

[42] Jacques Patarin. Hidden Field Equations (HFE) and Isomorphisms of Polynomials (IP): Two new families of asymmetric algorithms. In Ueli Maurer, editor, *Advances in Cryptology - EUROCRYPT 96*, volume 1070 of *Lecture Notes in Computer Science*, pages 33–48. Springer-Verlag, 12–16 May 1996.

[43] Jacques Patarin. Hidden Field Equations (HFE) and Isomorphisms of Polynomials (IP): Two new families of asymmetric algorithms - Extended Version, 1996.

[44] Jacques Patarin, Nicolas T. Courtois, and Louis Goubin. QUARTZ, 128-bit Long Digital Signatures. In David Naccache, editor, *Topics in Cryptology - CT-RSA 2001*, volume 2020 of *Lecture Notes in Computer Science*, pages 282–297. Springer Berlin Heidelberg, 2001.

[45] Jacques Patarin and Louis Goubin. Trapdoor one-way permutations and Multivariate Polynomials - Extended Version. In *Proc. of ICICS 97*, *LNCS 1334*, pages 356–368. Springer, 1997.

[46] Albrecht Petzoldt, Stanislav Bulygin, and Johannes Buchmann. CyclicRainbow – A Multivariate Signature Scheme with a Partially Cyclic Public Key. In Guang Gong and KishanChand Gupta, editors, *Progress in Cryptology - INDOCRYPT 2010*, volume 6498 of *Lecture Notes in Computer Science*, pages 33–48. Springer Berlin Heidelberg, 2010.

[47] Albrecht Petzoldt, Stanislav Bulygin, and Johannes Buchmann. Selecting parameters for the Rainbow Signature Scheme. In Nicolas Sendrier, editor, *Post-Quantum Cryptography*, volume 6061 of *Lecture Notes in Computer Science*, pages 218–240. Springer Berlin Heidelberg, 2010.

[48] Koichi Sakumoto, Taizo Shirai, and Harunaga Hiwatari. On provable security of UOV and HFE Signature Schemes against Chosen-Message Attack. In Bo-Yin Yang, editor, *Post-Quantum Cryptography*, volume 7071 of *Lecture Notes in Computer Science*, pages 68–82. Springer Berlin Heidelberg, 2011.

[49] Peter W. Shor. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997.

[50] Routo Terada. *Segurança de dados: Criptografia em redes de computador*. Blucher, 2ª revisada e ampliada edition, 2008.

[51] Xiaoyun Wang, Yiqun Lisa Yin, and Hongbo Yu. Finding collisions in the full SHA-1. In *Advances in Cryptology - CRYPTO 2005*: 25th Annual International Cryptology Conference, Santa Barbara, California, USA, August 14-18, 2005, Proceedings, volume 3621 of *Lecture Notes in Computer Science*, pages 17–36. Springer, 2005.

[52] Christopher Wolf. Implementing QUARTZ in java. Draft for the 3rd NESSIE Workshop. http://www.christopher-wolf.de/ql/quartzJava.pdf, 2002. Último acesso em 05/07/2013.

[53] Christopher Wolf. QuartzLight in Java. http://www.christopher-wolf.de/ql/, 2002. Último acesso em 17/07/2013.

[54] Christopher Wolf. *Multivariate Quadratic Polynomials in Public Key Cryptography*. PhD thesis, Katholieke Universiteit Leuven – Faculteit Ingenieurswetenschappen - Departement Elektrotechniek (ESAT), 2005.

[55] Christopher Wolf and Bart Preneel. Taxonomy of Public Key Schemes based on the problem of Multivariate Quadratic equations. Cryptology ePrint Archive, Report 2005/077. http://eprint.iacr.org/2005/077, 2005. Último acesso em 28/06/2013.

[56] Takanori Yasuda, Kouichi Sakurai, and Tsuyoshi Takagi. Reducing the Key Size of Rainbow using non-commutative rings. In Orr Dunkelman, editor, *Topics in Cryptology – CT-RSA 2012*, volume 7178 of *Lecture Notes in Computer Science*, pages 68–83. Springer Berlin Heidelberg, 2012.

Ewerton R. Andrade possui graduação em Sistemas de Informação pelo Centro Universitário Luterano de Ji-Paraná (2009), também é graduado em Matemática pela Universidade Federal de Rondônia (2011), mestrado em Ciência da Computação pelo Instituto de Matemática e Estatística da Universidade de São Paulo (2013), e atualmente é doutorando em Engenharia de Computação pela Escola Politécnica da Universidade de São Paulo. Tem experiência na área de Ciência da Computação, com ênfase em Segurança de Dados / Criptografia, atuando principalmente nos seguintes temas: segurança de dados, criptografia, assinaturas digitais baseadas em polinômios multivariados quadráticos, funções de derivação de chaves e esquemas de hash de senhas.

Routo Terada possui graduação em Engenharia Elétrica Eletrônica pela Universidade de São Paulo (1970), mestrado em Matemática Aplicada pela Universidade de São Paulo (1975) e doutorado em Computer Science - University of Wisconsin - Madison (1979). Atualmente é professor titular da Universidade de São Paulo, avaliador de artigos do International Journal of Information Security e do Journal of the Brazilian Computer Society. Tem experiência na área de Ciência da Computação, com ênfase em Criptografia, atuando principalmente nos seguintes temas: segurança de dados, criptografia, algoritmos, criptossistemas.