Short Signatures, Provable Security, Generic Attacks and Computational Security of Multivariate Polynomial Schemes such as HFE, Quartz and Sflash

(draft) [⋆]

Nicolas T. Courtois
courtois@minrank.org
Axalto Cryptographic Research & Advanced Security,
36-38 rue de la Princesse, BP 45, F-78430 Louveciennes Cedex, France,

Abstract. The object of this paper is the concrete security of recent multivariate signature schemes. A major challenge is to reconcile some "tricky" ad-hoc constructions that allow to make short signatures, with regular provable security. The paper is composed of two parts.
In the first part of this paper we formalize and confront with the most recent attacks the security of several known multivariate trapdoor functions. For example the signature scheme Quartz is based on a trapdoor function G belonging to a family called HFEv-. It has two independent security parameters, and we claim that if d is big enough, no better method to compute an inverse of G than the exhaustive search is known. This will allow us to formulate our key assumption on which the provable security results can be build.
In the second part, we study the security concrete security of signature schemes under our assumption. We study some general constructions, that transform a trapdoor function into a short signature scheme, and in particular these designed to obtain short signatures. On the one hand, we present generic attacks on such constructions. On the other hand, we study the possibility to prove or justify the security with some well chosen assumptions.
Unfortunately for Quartz, our lower and upper security bounds do not coincide. Still the best attack known for Quartz is our generic attack using $\mathcal{O}(2^{80})$ computations with $\mathcal{O}(2^{80})$ of memory. We will also propose an alternative way of doing short signatures for which both bounds do coincide. Finally we also apply our results for Flash and Sflash.

Key Words: Generic signature schemes, short signatures, provable security, Hidden Field Equations (HFE), HFEv-, Quartz, Flash, Sflash, Nessie. MQ, algebraic attacks.

Note: The work described in this paper has been supported by the French Ministry of Research under RNRT Project "Turbo-signatures". Currently it is supported by another RNRT Project "X-CRYPT" and by the European NOE project ECRYPT.

[⋆] This paper can be seen as an updated and very much extended version with a lot of new material, of the paper "Generic Attacks and the Security of Quartz" published at PKC 2003, LNCS 2567, Springer, pages 267-278, and also presented in an even earlier version at the second Nessie workshop, September 13th 2001, Royal Holloway, University of London. June 15, 2005

1 Introduction

1.1 Provable Security

Provable security in concrete realistic setting is the new mainstream paradigm for "good" cryptography. It makes strong security definitions that capture all known attacks and allows to prove under well chosen assumptions concrete lower bounds on the security of cryptographic schemes. The ultimate goal is to make these bound coincide with the upper bounds obtained by cryptographic attacks on different levels of abstraction. Currently new cryptographic schemes based on multivariate polynomials have clearly much less provable security results than for number theory-based schemes. The ambition of this paper is to partially fill this gap and trace a roadmap for future research. A major challenge is to reconcile some "tricky" ad-hoc constructions that allow to make short signatures with provable security.

1.2 Short Signatures

The shortest signature scheme known in the "classical cryptography" is based on Weil pairing and achieves 160 bits with the security of 2^{80} [4]. Only recently schemes with signatures shorter than 160 bits have been proposed. These new schemes belong to the multivariate cryptography. Quartz, proposed for standardisation in the European Nessie project, achieves signatures of 128 bits with a claimed security level of 2^{80}. The McEliece-based signature scheme CFS gives signatures of about 80 bits [15] but has a substantially bigger public key than Quartz. Both Quartz and McEliece have features that make them a unique choice for some applications, while remain excluded from other applications. It seems that, see [15], the security of McEliece signatures can be proven in the random oracle model and the lower and upper bounds coincide with respect to two well known problems. In the present paper we will try to see what are the lower and upper bounds on the security of Quartz, based on some well-chosen (but plausible) assumptions.

1.3 Summary

The paper is organized as follows: First we overview the hardness properties that, given all the known attacks, are believed to hold for such multivariate trapdoor functions as HFEv-. In particular we formulate the Assumption 5.0.8 which is the basis of all our subsequent security considerations. In the section 9 we study generic attacks on such meta signature schemes. Then in the Section 10 and in Appendix B we study if the security of Quartz can be proved, or justified, based on our Assumption 5.0.8. Finally we give our conclusions.

Part I

Security of Multivariate Trapdoor Functions

2 Basic Notations

Let $GF(q)$ be a finite field. In this paper we usually have $q = 2$ (but not always). In general we study multivariate schemes over $GF(q)$, i.e. the input and the output values consists of several variables, usually n e.g. $x = (x_0, \ldots, x_{n-1})$ denotes a variable in $GF(q)^n$; In the present paper we always have $q = 2$ and $x = (x_0, \ldots, x_{n-1})$ can simply be viewed as a string of bits.

Quartz is based on a family of HFEv- multivariate schemes [31] as described first by Jacques Patarin in the extended version of [31]. A full description of Quartz requires several pages, and we refer to the specification submitted to the European Nessie call for cryptographic primitives [34, 35]. Quartz has the following parameters: the integers q, d, h, v, r. Two additional integers are defined with respect to the above parameters: $n \overset{def}{=} h + v$ and $m \overset{def}{=} h - r$. In Quartz we have $q = 2$, $d = 129$, $h = 103$, $v = 4$, $r = 3$, and therefore $n = 107$ and $m = 100$.

3 Multivariate Quadratic Trapdoor Functions

The public key of a multivariate scheme is usually a system of multivariate quadratic polynomials $G : GF(q)^n \rightarrow GF(q)^m$, in this paper $q = 2$.

$$\begin{cases} y_0 = G_0(x_0, \ldots, x_{n-1}) \\ \quad\vdots \\ y_m = G_{m-1}(x_0, \ldots, x_{n-1}) \end{cases}$$

$$G_i(x_0, \ldots, x_{n-1}) = \sum_{1 \le j < k \le h} \zeta_{i,j,k} x_j x_k + \sum_{1 \le j \le h} \nu_{i,j} x_j + \rho_i,$$

all the elements $\zeta_{i,j,k}$, $\nu_{i,j}$ and ρ_i being in $GF(q)$. The private key is some hidden algebraical or combinatorial structure that allows to inverse G.

Let G be a multivariate (trapdoor or not) function defined by some probability distribution $\mathcal{G}$. Usually $\mathcal{G}$ will have parameters (q, n, m), and possibly some other that we ignore for simplification. For example $\mathcal{G}(q, n)$ can be a randomly selected *basic* HFE scheme from [31]. The notation $G \leftarrow \mathcal{G}(q, n)$ means that we pick a trapdoor function $G : GF(q)^n \rightarrow GF(q)^n$ from such a distribution.
Similarly we denote by $G \leftarrow \text{HFEv}^-(2, 107, 100)$, a random Quartz public key, $G : GF(2)^{107} \rightarrow GF(2)^{100}$. We will not enter into details how the distribution $\text{HFEv}^-(2, 107, 100)$ is constructed and how the trapdoor works. We only need to know that it produces trapdoor functions with $n = 107$ variables and $m = 100$ equations over $GF(2)$.

4 Cracking Problems

Definition 4.0.1 (Adversary). A T-time adversary is a probabilistic Turing machine that stops in time $\leq T$ and outputs an answer.

The probability of success of the Adversary will be noted ε with $\varepsilon > 0$. We note that for both T and ε: may be variable and depend on $(q, n, m, etc..)$, for example $T = q \times n^{\mathcal{O}(1)}$, or they may be fixed, for example $\varepsilon = 2^{-64}$.

Definition 4.0.2 (The Cracking Problem).
Given $G \leftarrow \mathcal{G}$, with $G : GF(q)^n \rightarrow GF(q)^m$ and a random $y \leftarrow GF(q)^m$; Find with a non-negligible probability at least $\varepsilon(q, n, m)$, and in time bounded by $T(q, n, m)$ (at least) one solution $x \in GF(q)^n$ to

$$y = G(x).$$

We denote $\mathcal{MQ}(q, n, m)$[1] the probability distribution that consists of taking a random set of m quadratic equations over $GF(q)$ with n variables. The cracking problem for such a random set of quadratic equations is:

Definition 4.0.3 (MQ problem). Given $G \leftarrow \mathcal{MQ}(q, n, m)$ and a random y, find with a non-negligible probability $\geq \varepsilon$ and in time bounded by T (at least) one solution x to $y = G(x)$.

The MQ problem is not only worst-case difficult, as it is proven NP-complete in [23] for $GF(2)$, and in the extended version of [**?**] in general case. Moreover it seems very hard for all but a negligible number of functions and for all but a negligible number of outputs. Following [37]:

Critical Claim 4.0.4. For $q = 2$, when $m \approx n$ holds[2], and when $m \leq 100$ no method is known to solve a randomly chosen $G \leftarrow MQ(q, n, m)$ considerably faster that the exhaustive search in q^m.

In fact for $q = 2$ it should hold for bigger m than 100, and it also should hold for other small q (e.g. $q = 3$) but only for smaller values of m, see Section 6.

We define the following (quite strong) property that should hold and seems to hold for all "interesting" MQ instances:

Definition 4.0.5 (Exhaustive security).
Let $G \leftarrow \mathcal{G}(q, n, m)$ be a function $G : GF(q)^n \rightarrow GF(q)^m$. Let $n \geq m$.We say that G achieves exhaustive security if for all Adversaries A running in T CPU clocks such that

[1] MQ means Multivariate Quadratic system of equations, see [37, 32].

[2] In fact only when $m \approx n$ the problem is still believed difficult. On one hand when $m >> n$ the problem becomes much easier, see [37]. On the other hand, when $n >> m$, several algorithms much faster than exhaustive search are presented in [13, 16].

$$\varepsilon = Pr[G \leftarrow \mathcal{G}(q,n,m), y \leftarrow GF(q)^m : \ G(A^{G,y}) = y]$$

we have the following inequality

$$\varepsilon \leq T/q^m.$$

The converse is obviously true for **any** (trapdoor or not) function:

Theorem 4.0.6 (Generic attack on a trapdoor function).
Let $G : GF(q)^n \rightarrow GF(q)^m$ be a function that is computed in an efficient way (we assume that the computing time is bounded by a small degree polynomial and neglect it). For all $0 \leq T \leq q^m$ there is an adversary that computes an inverse of G for a given $y \leftarrow GF(q)^m$ chosen uniformly at random, with success probability of:

$$\varepsilon \approx T/q^m.$$

Following the conclusions of the paper [37] we conjecture that:

Conjecture 4.0.7 (Exhaustive security of MQ instances).
Let $q = 2$ $m \approx n$, $n \geq m$ and $m \leq 100$. Then for any random $G \leftarrow MQ(q,n,m)$, G achieves the exhaustive security.

Example 4.0.8. If $q = 2$, $m = 80$ we expect to use not less (and not more) than 2^{40} computations in order to be able to compute a solution to MQ with probability at least 2^{-40}.

Remark: This conjecture is required in this paper. Again, it should hold even for bigger m and for other small q (e.g. $q = 3$) but only for smaller values of m, see Section 6. At any rate it holds for Quartz and with respect to all known attacks. In fact, it motivates the current research in multivariate cryptography:

Design Criterion 4.0.9 (Near-exhaustive security). A good multivariate cryptographic scheme should have a security close to the exhaustive search (q^m here).

The idea is that it should be considerably better than the square root ($q^{m/2}$ here) of the exhaustive search. Otherwise no one probably will bother with multivariate cryptography and use extensively studied group-based schemes such as RSA and Elliptic Curves. Their drawback is that on any group there are generic algorithms, precisely much faster than the exhaustive search (in the square root of the group size), such as Pollard's rho algorithm.

4.1 Distinguishability Problems

The strongest security claims made in cryptography are usually about indistinguishability with respect to some (ideal or real) random objects.

Definition 4.1.1 (the Distinguishability Problem). It is the problem of distinguishing $G \leftarrow \mathcal{G}$ from the random set of quadratic polynomials $G' \leftarrow MQ(q, n, m)$.

Definition 4.1.2 (Distinguishers). A T-time distinguisher is a T-time adversary that takes as an input a given $G \leftarrow \mathcal{G}(q, n, m)$ and outputs a yes or a no (0 or 1). The probability it outputs 1 on $G \leftarrow \mathcal{G}$ is denoted by

$$Pr[G \leftarrow \mathcal{G}(q, n, m) : A^G = 1]$$

Definition 4.1.3 ((T, ε)-Pseudo Random MQ). Let A be a T-time distinguisher. We define the distinguisher's advantage as:

$$Adv_{\mathcal{G}}^{PRMQ}(A) \overset{def}{=} \Big| Pr[G \leftarrow \mathcal{G} : A^G = 1] - Pr[G' \leftarrow \mathcal{MQ} : A^{G'} = 1] \Big|$$

We say that $\mathcal{G}$ is (T, ε)-indistinguishable from MQ (or a (T, ε)**-PRMQ**) if we have:

$$\underset{T\text{-time A}}{Max} \quad Adv_{\mathcal{G}}^{PRMQ}(A) \quad \leq \varepsilon$$

Definition 4.1.4 (PRMQ). We say that $\mathcal{G} : GF(q)^n \rightarrow GF(q)^m$ is a **PRMQ** if it is (T, ε)-PRMQ for all (T, ε) such that

$$\varepsilon > T/q^m.$$

Design Criterion 4.1.5 (PRMQ trapdoors). A good multivariate cryptographic trapdoor function should be a PRMQ.

We denote HFEv-(q, n, m) the trapdoor function generator used in Quartz, see [35] and [31] for the exact description. In the current state of knowledge it seems that:

Conjecture 4.1.6. The Quartz public key generator HFEv-$(2, 107, 100)$ is a PRMQ.

This conjecture holds with respect to all known attacks on HFE family of cryptosystems. We study this in more details in Section 8.

5 Hardness of HFEv-

We may now easily show that:

Theorem 5.0.7 (Exhaustive security of HFEv-).
Let $q = 2$, $m \approx n$, $n \geq m$ and $m \leq 100$. Let G randomly chosen as $G \leftarrow \mathrm{HFEv}^{-}(q, n, m)$. If the Conjectures 4.0.7 and 4.1.6 hold then G achieves the exhaustive security.

This result will be treated as an assumption in the remaining part of the paper. Since we don't really know if the Conjectures 4.0.7 and 4.1.6 are true, we would like to base the security on one single assumption. It can be rewritten as follows:

Critical Assumption 5.0.8 (Strong One-wayness of HFEv-).
Let G a random public key $G \leftarrow \mathrm{HFEv}^{-}(q, n, m)$ with $q = 2$, $m \approx n$, $n \geq m$ and $m \leq 100$. For any Adversary A running in T CPU clocks, given random $y \leftarrow GF(q^m)$, the probability that A outputs some $x = G^{-1}(y)$ with probability ε, satisfies:[3]:

$$\varepsilon \leq T/q^m.$$

[3] The assumption and all our results might be reformulated when T/q^m is replaced by any other function $E(T, q, n)$. However we would no longer achieve, neither the same bounds, nor such short signatures as in Quartz.

6 More on Hardness of MQ vs. Recent Algebraic Attacks

In this section we consider the validity of our Critical Claim 4.0.4 and our (related but stronger) Conjecture 4.0.7, beyond our assumptions: when q becomes bigger than 2, and m being potentially bigger than 100. This is an ongoing research topic.

In general, the maximum m such that these hold, will depend on q.

When $\mathbf{q = 2}$. For $q = 2$ we expect in fact that our Conjecture 4.0.7 holds for m bigger than 100, this bound is conservative and due to over-optimistic estimations of [37] on the behaviour of the XL algorithm.

Asymptotic aspects: Moreover, these over-optimistic estimations of [37] seems to suggest that XL would be subexponential for MQ over small finite fields and when $m \approx n$. This is **not at all** confirmed by the specialists of Gröbner bases. Apparently, applying the Buchberger algorithm to ideals generated in XL over $GF(2)$ [37, 12] has single exponential worst case complexity, see [19] or [2]. It is therefore possible that our Conjecture 4.0.7 holds in practice for bigger m. Yet probably not for any m [Jean-Charles Faugère, Magali Bardet, private communication, see also [20, 21]].

Bigger q. Recent advances (In [10], seriously revised and corrected in [9]) on improved versions on the XL algorithm from [37, 12, 14] and (very closely related) efficient methods for computing Gröbner bases [20, 21] indicate that our Conjecture 4.0.7 can only be true for very small q such as 2.

For example for $q = 2^7$, $n \geq m = 26$ according to [10] it is possible to solve the equations in 2^{58} by the XL' attack from [37], while $q^m = 2^{182}$. In [9] it is shown that the complexity of this XL' attack is in fact 2^{118}. Nevertheless another attack called XFL from [10] gives 2^{99} and all these figures remain much smaller than 2^{182}.

Another example is the "HFE Challenge 2", for which $q = 2^4$, $n \geq m = 32$ and the best attack known is again the XFL attack with complexity evaluated in details in [9] to be 2^{93}, less even than $q^{m/2} = 2^{128}$.

Is it really hard ? We face the problem of the hardness of MQ that is a fundamental problem of cryptography underlying the security of many cryptosystems including AES, see [17, 6, 5, 10]. More research on this topic is needed. Even though attacks on such problems have known huge progress in the recent years, see [37, 12, 14, 20, 21, 9, 10], it is possible to see that the efficiency of all these methods is limited by some algebraic invariant properties of the ideal generated by the system of polynomials. They are also limited by the speed of well known fundamental algorithms such as linear algebra.

7 Confronting the Assumption 5.0.8 with Recent Attacks for Sflash and Variants

Our Assumption 5.0.8 (Strong One Wayness) is based on two requirements: Indistinguishability (Conjecture 4.1.6) and the one-wayness of the underlying generic problem (Definition 4.0.5). We will see that the only the first requirement is satisfied for Sflash and version of it.

7.1 About Indistinguishability (Conjecture 4.1.6) of Sflash

The Results of Joux-Faugère Applied to Sflash

In [18] it is shown that Gröbner bases attacks, can distinguish Sflash from MQ for systems over $GF(2)$, and for the number of removed equations r being not too big (but may be > 2). Joux and Faugère fully develop and this attack in [26].

We will explain that for systems over $GF(2^k)$, $k > 1$, this attack does **not** extend well. We will explain this in the light of the paper [26] by Joux and Faugère.

For this, we will view the public key of Sflash (and other systems) as a function $G : GF(q^n) \to GF(q^n)$ (such univariate representation is used for example in [38, 26]). From [30] we know that for Sflash over $GF(q)$ with 0 equations removed, Then, if $Y = G(X)$, there exists the following equation in $GF(q^n)$ (that is a consequence of hidden algebraic structure of G, see [30]):

$$A(X,Y) = \sum_{ij} \alpha_{ij} X^{q^i} Y^{q^j} + \sum_i \beta_i X^{q^i} + \sum_i \gamma_i Y^{q^i} + \delta = 0$$

Now, if we consider Sflash with r last public multivariate equations removed and replaced by arbitrary r multivariate equations, and again we see it as a univariate function $G : GF(q^n) \to GF(q^n)$, we obtain that if $Y = G(X)$, we have:

$$\prod_{\Delta \in \{(0,\ldots,0)\} \times GF(q)^r} A(X, Y + \Delta) = 0$$

Then, when $q = 2$ and $r = 1, 2, 3$ the degree of this equation is not too high. this explains the nice results obtained in [18]. However when $q = 2^7$, even for $r = 1$ this equation does not give multivariate relations that will be detectable in practice. We confirmed this by computer simulations.

7.2 Our Simulations - Distinguishing Sflash from MQ

In the Table 1 we present results of our simulations, to see if Sflash can be distinguished from a random MQ system.

The value r is critical parameter. When it is zero, then Sflash becomes the Matsumoto-Imai cryptosystem also called C^*, easy to distinguish from random, by Patarin equations $A(X, Y)$ from [30] that we discussed above.

The object of our simulations is to see what is the maximum r for which we may distinguish the two cases by simulations that would take less than day on a PC. This allows to see if Sflash has or not a good security margin.

Conclusion: Our simulations show clearly that it is very hard to distinguish (by algebraic methods) Sflash from a random system of multivariate quadratic equations (MQ) even when a few equations are removed.

In practice, for $q = 2^7$ we have sometimes found some relations for $r = 1$, being probably of different origin than Joux attack above. For $r = 2$ we have found nothing, not even for $n = 5$ variables. This should mean that there is none either for any $r \geq 2$ and any $n \geq 5$). In Sflash we have $r = 11$ equations removed, and these simulations show that it has an excellent security margin with respect to the existence of detectable multivariate relations.

Table 1. Distinguishing Sflash from Random MQ over $GF(2^7)$ with XL-based method

n		5	5	5	5	5	5	5	5	5	5	5	5
m		5	5	5	5	5	5	4	4	4	4	4	4
D		3	4	5	6	7	3	3	4	9	15	3	3
D'		0	0	0	0	0	0	0	0	0	0	0	1
R		30	105	280	630	1260	30	25	84	3168	34272	25	30
T		56	126	252	462	792	56	35	126	2002	15504	35	56
Free	Sflash	29	89	205	405	725	29	24	78	1874	15280	24	29
	MQ	30	95	220	430	760	30	24	78	1874	15280	24	30

n		5	5	5	5	5	5
m		4	4	3	3	3	3
D		3	3	3	3	9	15
D'		0	1	2	2	8	14
R		25	30	24	39	3663	37332
T		35	56	56	56	2002	15504
Free	Sflash	24	29	24	36	1934	15388
	MQ	24	30	24	36	1934	15388

n		7	7	7	7	7	7	7	7	7	7	7	7
m		7	7	7	7	7	6	6	6	6	6	6	6
D		3	4	5	6	7	3	4	6	3	3	4	6
D'		0	0	0	0	0	0	0	0	0	2	1	5
R		56	252	840	2310	5544	48	216	1980	48	84	224	2772
T		120	330	792	1716	3432	120	330	1716	120	120	330	1716
Free	Sflash	56	231	672	1589	3304	48	201	1460	48	78	209	1652
	MQ	56	231	672	1589	3304	48	201	1460	48	78	209	1652

n		17	17	17	17	17	17
m		17	16	16	16	15	15
D		3	3	4	3	3	4
D'		0	0	0	1	1	3
R		306	288	2736	306	288	3705
T		1140	1140	5985	1140	1140	5985
Free	Sflash	305	288	2616	305	288	3330
	MQ	306	288	2616	306	288	3330

Legend:
n number of variables.
m number of equations.
D is the total degree of the XL equations
D' is the degree of the additional monomials in the x_i added to the system.
R number of equations generated (independent or not).
T number of monomials of degree $\leq D$.
Free number of linearly independent equations among the R equations.

7.3 Underlying One-way Problem for Flash and Sflash

As expected from the Joux attack on Sflash from [26], we discovered that in practice for Sflash, even when only 2 equations are removed, algebraic methods such as [26, 20, 21, 5, 18] cannot in practice distinguish Sflash from a random MQ. This is confirmed by all our computer simulations (in Table 1.)

Unfortunately recent attacks [10, 9] show that the security of the generic MQ problem is lower than expected, our Definition 4.0.5 and our Hardness Assumption 5.0.8 does not hold for these schemes, see Appendix B.3 and [10, 9].

Summary: Our Hardness of Assumption 5.0.8 does not hold for Flash and different versions of Flash and Sflash, not for structural reasons but because the underlying one-way problem does not achieve a security level close to exhaustive search (Definition 4.0.5) when $q = 2^7$.

Remark: For smaller q we would not have this problem. But then the attacks described in [18] would become more efficient, see our explanation based on the Joux-Faugère paper in Section 7.1.

8 Confronting the Assumption 5.0.8 with Recent Attacks on Quartz

8.1 Hardness of HFEv-

The main reason that such a (very, very) strong assumption can be made for HFEv-/Quartz is the following: The cryptosystems of the HFE family have two independent security parameters: the extension degree h and the degree of the hidden polynomial d. Quite often they also have additional security parameters (for example v and r here). This makes the study of their security much more complex than for cryptosystems that basically one security parameter such as RSA. However it also makes the multivariate cryptographic schemes much more flexible than the usual schemes. For example one parameter (in our case h) can usually be small to achieve cryptosystems that operate on small blocks (and allows e.g. short signatures), and the other parameters can be independently adjusted to achieve the desired security level. In other words, if it turns out that HFEv- does not satisfy our Assumption 5.0.8 above, it probably does when d is increased [4]. What is the value d that should be chosen, and whether Quartz is then practical or not[5], are two different questions that are out of the scope of this paper. In a way this paper studies the security that can be achieved given the first parameter h and (and also v and r), while assuming simply that d is big enough[6].

[4] This assuming that the generic problem MQ is as hard as expected, i.e. if for this system the Conjecture 4.0.7 holds. This is not always true see Section 7.3. Further research will determine if for equations over $GF(2)$ this conjecture is indeed true.

[5] The major drawback of Quartz is its slowness. This is currently the price to pay for such a short signature scheme and only two other short signature schemes known that give less than 160 bits of the Weil-pairing scheme [4]. The McEliece scheme from Asiacrypt 2001 is about as slow as Quartz, and has a much bigger public key of about 1Mbyte instead of 71 Kbytes, see [15]. There is also the degree 3 Dragon scheme (also based on HFE) which seems quite fast, but also has a very big public key, see [32, 27] and it is possible that a careful analysis of the security of this scheme will require that the parameter d would have to be revised, and it would end up being quite slow too.

[6] Finally, our Assumption 5.0.8 probably boils down to this.

8.2 Recent Attacks on HFEv-, Contributions and Inaccurate Claims of the Faugère-Joux Crypto 2003 Paper

The recent work of Joux [26] is definitely an important breakthrough, in allowing to understand the nature and the origin of the algebraic attacks on HFE and variants, previously discovered (experimentally) by Courtois *et al.* in [5, 18].

However, several claims and statements of this paper are inaccurate and misleading.

1. Second paragraph, page 45: Relinearization is **not** a well suited technique for solving HFE in the Shamir-Kipnis paper [38]. A highly confusing paragraph. Relinearization is a method to handle the second step of the Shamir-Kipnis attack. It is highly inefficient and not necessary at all to break HFE by Shamir-Kipnis method. The second step can and should be replaced by a better method, one of the algorithms for the MinRank problem, as suggested by Courtois in [5] or [16]. This will make the attack much, much more efficient.
2. The paper clearly states in the conclusion that its main result is establishing that, when the degree d of the hidden polynomial is fixed, the security of HFE grows polynomially in the number of variables n. Here, Faugère and Joux, has deliberately and in full knowledge, chosen to attribute to themselves a result, that exists in not less than 5 previously published works [31, 38, 16, 5, 18]. As we will explain now, all these already give a method that allows to break HFE in polynomial time (when d is fixed):

   a. First of all, this result was already known to Patarin, the inventor of HFE. It is shown in Section 4 of the published version of [31] and in Section 7.2. of the extended version of [31] that can be found at `http:/www.hfe.info`. Fearing that the word "polynomial complexity" would make people believe that HFE is insecure, the author does not use this word in [31], but the attack is clearly and fully described in [31]. In fact, HFE has very little to fear from this attack. It is easy to see that, though this attack is polynomial in n when d is fixed, it will be exponential in d, which makes it highly impractical even for quite small values of d, e.g. 17.

   b. The possibility of break HFE-type systems efficiently have been undoubtedly discovered by Courtois, in [5, 16], written in January

1999[7] and submitted to Crypto, at the same time as the Shamir-Kipnis paper [38].

c. This precise Shamir-Kipnis paper [38], is also a (very different) method that can allow to break some instances of HFE efficiently, but only when combined with improvements of Courtois (submitted to the same conference), and published finally only in 2001, see [5, 16].

   Important: Both the above attacks [b] and [c] are still polynomial in n when d is fixed, but now they are exponential in $\log d$, or $\log^2 d$, and not in d. This the reason why several HFE systems can be broken in practice, and all the work of Faugère [22, 5] is exploiting and improving on this initial discovery of Courtois [5] without due acknowledgment. In particular, in [22] Faugère is not the first to break HFE Challenge 1. It was first done in [16, 5], with a complexity that is lower than exhaustive search, and the attack [22] is mainly a matter of optimisation of this earlier Courtois attack.

d. Daum and Felke were first to discover that HFE with "variations", such as HFE- or HFEv, are also susceptible to be broken by algebraic attacks. They presented their results at Yacc 2002 conference (without formal proceedings) and later it was published by Courtois Daum and Felke at PKC 2003 [18]. Thus again, Faugère and Joux are not the first to discover that practical attacks on these "variations" are possible. The paper of Faugère and Joux [26] was however the first to propose a systematic construction of algebraic attacks for arbitrary HFEv- systems, and to evaluate their complexity.

   Note: Unfortunately the complete evaluation of the complexity of the attack for HFEv- and Quartz is not given in [26].

[7] Moreover, this attack did not came out of nowhere, and extends an even earlier algebraic attack of Patarin, that at Crypto 95 breaks the Matsumoto-Imai cryptosystem, which is in fact a special case of HFE, see [29, 30].

Part II

Security of Generic (Short) Signature Schemes

9 Feistel-Patarin Construction

9.1 Generic Threats

The classical way to compute digital signatures with a trapdoor function $G : GF(q)^n \rightarrow GF(q)^m$ is to compute a hash H, and the signature is given as:

$$\sigma = G^{-1}(H)$$

As a direct consequence of 4.0.6, such a signature can be forged in the square root of exhaustive search. This attack is generic: it does not depend on G. We produce a lists of $q^{m/2}$ $G(\sigma_i)$ and a list of $q^{m/2}$ hashes of different messages (or different versions of the same message). Then we expect to be able to produce at least one valid pair (message, signature), which is an existential forgery.

Remark: This generic attack is not an issue for most well-known signature schemes such as RSA, DSA, McEliece etc. It is because for all these functions there already exists an attack in the square root of exhaustive search (or less) and the parameters have already been chosen sufficiently large to avoid it. The situation is somewhat different for multivariate quadratic schemes such as HFE. For several such schemes, there is no attack known noticeably smaller than the exhaustive search. Therefore if the signature is computed as $\sigma = G^{-1}(H)$ for a scheme such as HFE or HFEv-, $q^{m/2}$ should be at least 2^{80} and it implies that the signatures should have at least 160 bits. For this reason, when it comes to shorter signatures, multivariate quadratic schemes usually compute a signature in different, somewhat strange way.

9.2 Removing Existential Forgeries

We assume $m = n$ (this condition will be relaxed later). How to compute a signature using a trapdoor function ? On one hand, the owner of the private key should use the computation of $G^{-1}()$ at least once, so that he will be the only person to be able to compute a signature. On the other hand, the verification should be in the implicit form Verif(σ, H) in order to avoid meet-in-the middle attacks. On the Figure 1 we show an example of such a construction derived from the Feistel scheme, in which the hash is divided in two pieces and the signature is computed as (cf. Fig. 1):

$$H(M) = (H_1(M), H_2(M))$$

$$\sigma = H_1 \oplus G^{-1}(H_2 \oplus G^{-1}(H_1))$$

Now, the meet-in-the middle attack fails: we can still produce two lists of candidate messages and candidate signatures, but we are unable

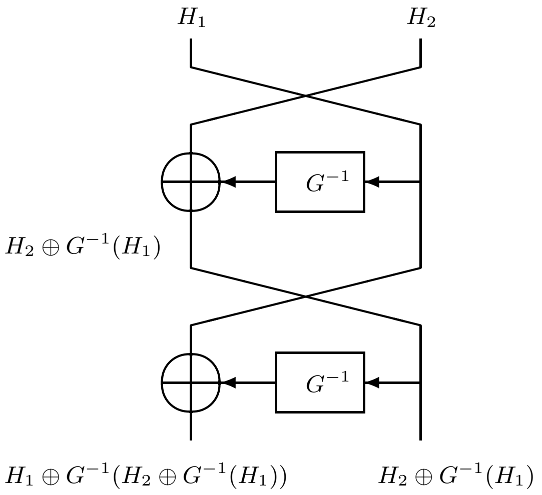

Fig. 1. 2-round Feistel applied to signature generation

to detect which signature correspond to which message by just sorting two lists. It is necessary to run the verification algorithm on each pair (message, signature) and it gives a complexity of q^m.

Security: Clearly, the Feistel-based (meta) signature scheme described above is better than the (ordinary) signature scheme, but still not perfect. In [32] Patarin explains that that a signature can still be forged in $q^{\frac{2m}{3}}$ instead of $q^{\frac{m}{2}}$ previously. For this, we precompute $q^{\frac{2m}{3}}$ values $f(X)$ for some $q^{\frac{2m}{3}}$ values for X. It allows to compute an inverse of G with probability $q^{-\frac{m}{3}}$. Thus one can compute two consecutive inverses with probability $q^{-\frac{2m}{3}}$. Then, given $q^{\frac{2m}{3}}$ messages we are able to forge a signature of (about) one of them. In general we have:

Theorem 9.2.1 (Generic attack on signature schemes).
Let $G : GF(q)^n \rightarrow GF(q)^m$. Any deterministic signature scheme that combines K inverses and message hash values can be broken in $q^{\frac{K}{K+1}m}$.

Proof. We precompute $q^{\frac{K}{K+1}m}$ values $f(X)$ for some $q^{\frac{K}{K+1}m}$ values for X. It allows to compute an inverse of G with probability $q^{-\frac{1}{K+1}m}$ and thus to compute iteratively K inverses with probability $q^{-\frac{K}{K+1}m}$. Thus with $q^{\frac{K}{K+1}m}$ messages we are able to forge a signature. □

Remark: For a function $G : GF(q)^n \rightarrow GF(q)^m$ the complexity is $q^{\frac{K}{K+1}m}$ with $q^{\frac{K}{K+1}m}$ of memory. It is in fact the best known attack against Quartz and gives 2^{80} computations with 2^{80} of memory.

It is obvious that when K grows, the complexity of the attacks tends to q^m of the exhaustive search. Unfortunately using the Feistel structure cannot easily be extended to more rounds: we will not have enough information to verify the signature anymore. Still the formula it gives can be generalized and in many different ways:

9.3 Extending to any Number of Inverses

For example we may consider the following (cf. Fig. 2):

$$H(M) = (H_1(M), H_2(M), \ldots, H_K(M))$$
$$\sigma = G^{-1}(H_K \oplus \ldots \oplus G^{-1}(H_3 \oplus G^{-1}(H_2 \oplus G^{-1}(H_1)) \ldots)$$

We call it the Feistel-Patarin signature scheme, though it has little to do (now) with the original Feistel scheme.

```
$\sigma \leftarrow 0$
for $i = 1$ to $K$ do
    {
        $\sigma \leftarrow \sigma \oplus H_i(M)$
        $\sigma \leftarrow G^{-1}(\sigma)$
    }
return $\sigma$
```

Fig. 2. Basic Feistel-Patarin scheme with K inverses

9.4 Extending to $m \neq n$

We need to generalize the above construction to trapdoor functions with input and output spaces of different sizes $G : GF(q)^n \rightarrow GF(q)^m$, and $m \neq n$. In this paper we limit to $m \leq n$, see [16] for the case $m > n$. The adaptation consists of cutting off and publishing the additional $(m - n)$ symbols obtained in every round, as they are a necessary ingredient in the signature verification process. It makes signatures somewhat longer. More precisely we do the following (Fig. 3):

```
$\sigma \leftarrow 0$
for $i = 1$ to $K$ do
{
        $\sigma \leftarrow \sigma \oplus H_i(M)$
        $U \in G^{-1}(\sigma)$
        $\sigma \leftarrow U_{1\rightarrow m}$
        $X_{i1}||\ldots||X_{i(n-m)} \leftarrow U_{(m+1)\rightarrow n}$
}
return $\sigma||X_{11}||\ldots||X_{K(n-m)}$
```

Fig. 3. Generalized Feistel-Patarin with $m \leq n$

This is precisely used in Quartz with $K = 4$, as represented on Fig. 4.

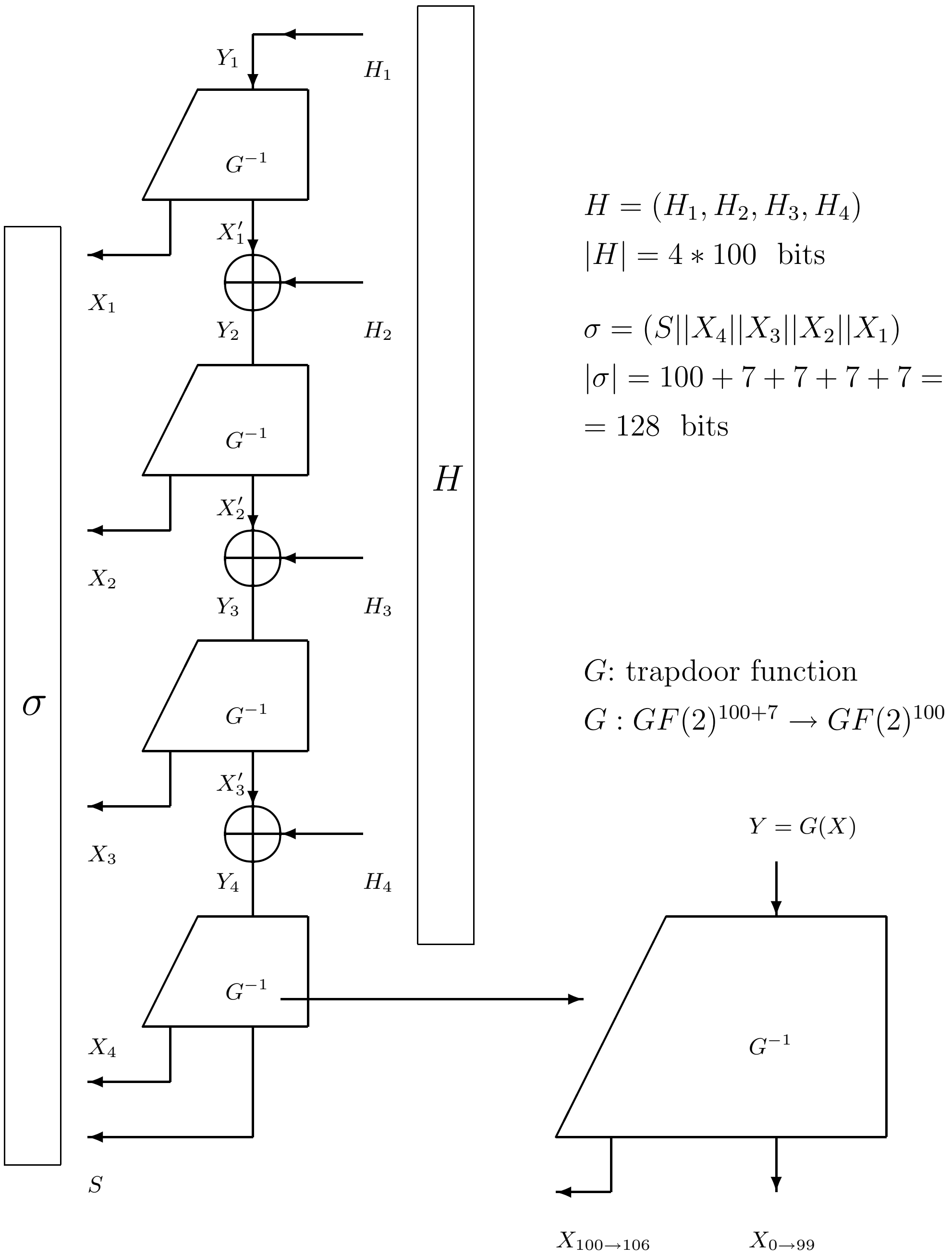

Fig. 4. Signature generation in Quartz

9.5 The Signature Length

The signature length in the generalized Feistel-Patarin construction is:

$$|\sigma| = (\ m + K(n-m)\ ) \cdot \log_2(q) \ \text{ bits}$$

Given the Theorem 9.2.1, the signature length at a given security level 2^{SF} is:

$$|\sigma| = \left\lceil \frac{K+1}{K} \cdot SF \right\rceil + K(n-m)\log_2(q) \ \text{ bits}$$

We note that the length decreases for small K, and then it increases. Therefore at some point the signature is the shortest. In Quartz the minimum is $|\sigma| = 128$ bits, achieved for both[8] $K = 3$ and $K = 4$.

10 Is it possible to Prove the Security of Quartz ?

We established lower bounds for the attacks on signature schemes such as Quartz. The question is now whether these lower bounds are also upper bounds, and more importantly, can the perfect correspondence between the generic attack 4.0.6 and the Assumption 5.0.8, be extended to Quartz ? This problem is studied in this section, and in more details in Appendix B. Though several open problems remain, we will prove several interesting results about the security of Quartz, Flash and Sflash.

We restrict to no-message attacks. The goal is to prove that an adversary cannot compute a valid pair (message, signature) given the public key. We assume the random oracle model with Q being the number of queries. It is a very powerful tool for security proofs. It allows an immediate reduction for a signature scheme of the form $G^{-1}(h)$: if the adversary computes a valid pair (message, signature) for some h taken out of the Q oracle answers, then we can transform this adversary into a machine that computes $G^{-1}(h)$ for one out of Q given values. This remains true not only for Q random values, but also for with probability very close to 1 to any set of chosen Q values. Indeed our reduction must behave exactly in the same way as with truly random values, provided that this set cannot be distinguished from random (which is in most cases easily achieved).

Thus we get machine that given some Q random (or chosen, but randomly looking values), outputs $G^{-1}(h)$ for one of them. By injecting a chosen value into a list of length Q, we get a a machine able to invert G with probability at least $1/Q$. However from the Assumption 5.0.8 we know that this success probability cannot be bigger than T/q^m, and thus $1/Q \leq T/q^m$. Reading the oracle takes time and thus we have $Q \leq T$. Combining these two equations gives: $T \geq Q \geq q^m/T$ and thus $T \geq q^{m/2}$. A signature cannot be forged in less than $q^{m/2}$ for no-message attacks.

[8] The reasons for which the $K = 4$ has been chosen, and not $K = 3$, is that the authors wanted some internal value, namely $h = m + r$ to be a prime, with $m = \left\lceil \frac{K+1}{K} \cdot 80 \right\rceil$ and $r = 3$, see [35, 34].

We note that one may think that this argument should also be applied to Sflash-v2 [7, 8, 10, 11]. If it satisfies our (quite strong) Assumption 5.0.8, then a signature cannot be forged in less than 2^{91}, again only for no-message attacks. Unfortunately recent attacks show that these schemes do **not** satisfy the assumption, see Appendix B.3 and [10, 9].

It also applies for the first G^{-1} in Quartz: its entry is given entirely by the oracle, see Fig. 4. A Quartz signature cannot be forged in less than 2^{50}, quite disappointing compared to the claimed security level of 2^{80}.

Now let us assume that in a signature scheme, several inverses G^{-1}, say K inverses, are computed for several H_i, such that all the H_i are independent parts of an output of one single application of a hash function. Then an adversary that can do an existential forgery, is able, with probability $1/Q$, to solve the inversion problem simultaneously for K independent instances H_i. Our Assumption 5.0.8 says that the probability of finding an inverse is at most T/q^m. It is therefore legitimate to think that the probability to compute K inverses in parallel will [9] be at most $(T/q^m)^K$. The adversary does with probability $1/Q$ something that can only be done with probability $(T/q^m)^K$. Thus, we get $1/T \geq 1/Q \geq (T/q^m)^K$. This gives $T \geq q^{\frac{K}{K+1}m}$. We obtain a lower bound that is the exact converse of our generic attack 4.0.6 !

From this one might think that it is possible to prove the security of Quartz and obtain an upper and a lower bound that coincide, thus achieving the exact security level of 2^{80} for no-message attacks. Unfortunately our argument does not apply to Quartz, the entries of the three other G^{-1} functions are not given by the random oracle, see Fig. 4. We are here at the heart of the problem of short signatures. Is it possible to have a signature scheme for which a lower bound on an attack can be proven that is more than $q^{\text{signature size}/2}$? The answer is yes and in Section B.4 we show a very surprising way to compute signatures, in which the signer does compute G^{-1} for two independent values H_1 and H_2, but in which the signature length is only the size of one $G^{-1}(H_i)$.

[9] It is not a consequence of our Assumption 5.0.8 and requires an additional assumption (cf. Assumption B.2.1 in Appendix B). It does not contradict any known attacks for Quartz. We need to assume that K is small. We may deduce this result, assuming that the only way to compute K inverses is to use the best algorithm to compute one inverse K times. For example the owner of the private key can compute 1 inverse with probability 1, and K inverses with probability at most $1^K = 1$, even here there is no contradiction. For the algorithms that does not contain the private key, we expect it to be true, because it seems that the only thing they can do to find solutions is to guess them, and our assumption is obviously true for any algorithm that is just guessing.

11 Conclusion

Quartz is based on a trapdoor function G that belongs to a family called HFEv-. It has two independent security parameters and if d is sufficiently large, there is no better method known to compute inverses of G, than simple guessing. We formalized this, and we studied what kind of security can be achieved by a general class of (short) signature schemes, under this assumption and in the random oracle model. On the one side we studied generic attacks on such signature schemes and gave exact lower security bounds. On the other side, we studied the security reductions that could give security upper bounds under some well chosen assumptions. Unfortunately for Quartz, our lower and upper bounds do not coincide. We also proposed a new method for computing short signatures for which the two bounds do coincide, however it is less general than the scheme of Quartz.

In practice it seems that Quartz, though lacking a tight security reduction, is still a correct way to achieve short signatures. The best attack known for Quartz is our generic attack using $\mathcal{O}(2^{80})$ computations with $\mathcal{O}(2^{80})$ of memory. In practice memory is very expensive and the fastest "memoryless" attack known requires as much as 2^{100} computations.

The methodology used in this paper is meant to allow formal security treatment of short signature schemes such as Quartz. It does not apply well for Flash and Sflash: these schemes do not satisfy our assumptions, and they are not meant to provide very short signatures.

References

1. Jee Hea An, Yevgeniy Dodis, Tal Rabin: *On the security of joint signatures and encryption,* Eurocrypt'02, LNCS 2332, Springer.
2. B. Barkee, D. C. Can, J. Ecks, T. Moriarty, R. F. Ree: *Why You Cannot Even Hope to use Gröbner Bases in Public Key Cryptography: An Open Letter to a Scientist Who Failed and a Challenge to Those Who Have Not Yet Failed,* in Journal of Symbolic Computation 18, 1994, S. 497-501
3. Nessie Security Report v1.0., pages 184-185, 2002, available from `www.cryptonessie.org`.
4. Dan Boneh, H. Shacham, and B. Lynn: *Short signatures from the Weil pairing,* Asiacrypt 2001, LNCS 2139, Springer, pp. 514-532.
5. Nicolas Courtois: *The security of Hidden Field Equations (HFE)*; Cryptographers' Track Rsa Conf. 2001, LNCS 2020, Springer, pp. 266-281.
6. Nicolas Courtois: *General Principles of Algebraic Attacks and New Design Criteria for Components of Symmetric Ciphers*, in AES 4 Conference, Bonn May 10-12 2004, LNCS 3373, pp. 67-83, Springer.
7. Nicolas Courtois, Jacques Patarin, Louis Goubin: *Flash, a fast multivariate signature algorithm*; Cryptographers' Track Rsa Conference 2001, LNCS 2020, pp. 298-307, Springer. Contains the first version of Sflash algorithm (Sflash-v1) as initially submitted to Nessie.
8. Nicolas Courtois, Louis Goubin and Jacques Patarin: Second updated version of Sflash specification (Sflash-v2). Available at `http://www.cryptosystem.net/sflash/`
9. Jiun-Ming Chen, Nicolas Courtois and Bo-Yin Yang: *On Asymptotic Security Estimates in XL and Gröbner Bases-Related Algebraic Cryptanalysis,* ICICS'04, LNCS 3269, pp. 401-413, Springer, 2004.
10. Nicolas Courtois: *Algebraic Attacks over $GF(2^k)$, Application to HFE Challenge 2 and Sflash-v2.* In PKC 2004, LNCS 2947, pp. 201-217, Springer, 2004.
11. Nicolas Courtois, Louis Goubin and Jacques Patarin: SFLASHv3, a fast asymmetric signature scheme. New, third version of Sflash specification (Sflash-v3), proposed after this paper was written. Available on `eprint.iacr.org/2003/211/`.
12. Nicolas Courtois and Jacques Patarin, *About the XL Algorithm over $GF(2)$,* Cryptographers' Track RSA 2003, LNCS 2612, pages 141-157, Springer 2003.
13. N. Courtois, L. Goubin, W. Meier, J.-D. Tacier: *Solving Underdefined Systems of Multivariate Quadratic Equations,* PKC 2002, LNCS 2274, Springer, pp. 211-227.
14. Nicolas Courtois: *Higher Order Correlation Attacks, XL algorithm and Cryptanalysis of Toyocrypt,* ICISC 2002, LNCS 2587, pp. 182-199, Springer. An updated version is available at `http://eprint.iacr.org/2002/087/`.
15. Nicolas Courtois, Matthieu Finiasz and Nicolas Sendrier: *How to achieve a McEliece-based Digital Signature Scheme*; Asiacrypt 2001, LNCS2248, Springer, pp. 157-174.
16. Nicolas Courtois: *La sécurité des primitives cryptographiques basées sur les problèmes algébriques multivariables MQ, IP, MinRank, et HFE*, PhD thesis, Paris 6 University, 2001, in French.
17. Nicolas Courtois and Josef Pieprzyk, *Cryptanalysis of Block Ciphers with Overdefined Systems of Equations,* Asiacrypt 2002, LNCS 2501, pp.267-287, Springer, a preprint with a different version of the attack is available at `http://eprint.iacr.org/2002/044/`.

18. Nicolas Courtois, Magnus Daum and Patrick Felke: *On the Security of HFE, HFEv- and Quartz,* PKC 2003, LNCS 2567, Springer, pp. 337-350. The extended version can be found at `http://eprint.iacr.org/2002/138/`.
19. Magnus Daum: *Das Kryptosystem HFE und quadratische Gleichungssysteme über endlichen Körpern,* Diplomarbeit, Universität Dortmund, 2001. Available from `daum@itsc.ruhr-uni-bochum.de`
20. Jean-Charles Faugère: *A new efficient algorithm for computing Gröbner bases (F_4),* Journal of Pure and Applied Algebra 139 (1999) pp. 61-88. See `www.elsevier.com/locate/jpaa`
21. Jean-Charles Faugère: *A new efficient algorithm for computing Gröbner bases without reduction to zero (F5),* Workshop on Applications of Commutative Algebra, Catania, Italy, 3-6 April 2002, ACM Press.
22. Jean-Charles Faugère: Report on a successful attack of HFE Challenge 1 with Gröbner bases algorithm F5/2, announcement that appeared in `sci.crypt` newsgroup on the internet in April 19th 2002.
23. Michael Garey, David Johnson: *Computers and Intractability, a guide to the theory of NP-completeness*, Freeman, p. 251.
24. Henri Gilbert, Marine Minier: *Cryptanalysis of SFLASH.* Eurocrypt 2002, LNCS 2232, Springer, pp. 288-298, 2002.
25. S. Goldwasser, S. Micali and R. Rivest: *A Secure Digital Signature Scheme,* Siam Journal on Computing, Vol. 17, 2 (1988), pp. 281-308.
26. Antoine Joux, Jean-Charles Faugère: *Algebraic Cryptanalysis of Hidden Field Equation (HFE) Cryptosystems Using Gröbner Bases*, Crypto 2003, LNCS 2729, pp. 44-60, Springer.
27. Neal Koblitz: *Algebraic aspects of cryptography,* Springer, ACM3, 1998, Chapter 4: "Hidden Monomial Cryptosystems", pp. 80-102.
28. Gwenaëlle Martinet, Antoine Joux: *Some Weaknesses in Quartz Sign. Scheme,* preprint.
29. Tsutomu Matsumoto, Hideki Imai: *Public Quadratic Polynomial-tuples for efficient signature-verification and message-encryption*; EUROCRYPT'88, Springer 1998, pp. 419-453.
30. Jacques Patarin: *Cryptanalysis of the Matsumoto and Imai Public Key Scheme of Eurocrypt'88,* Crypto'95, Springer, LNCS 963, pp. 248-261, 1995.
31. Jacques Patarin: *Hidden Fields Equations (HFE) and Isomorphisms of Polynomials (IP): two new families of Asymmetric Algorithms,* Eurocrypt'96, pp. 33-48.
32. Jacques Patarin: *La Cryptographie Multivariable*; Mémoire d'habilitation à diriger des recherches de l'Université Paris 7, 1999.
33. Jacques Patarin, Nicolas Courtois, Louis Goubin: *C*-+ and HM - Variations around two schemes of T. Matsumoto and H. Imai,* Asiacrypt 1998, pp. 35-49. *Unbalanced Oil and Vinegar Signature Schemes;* Eurocrypt 1999, Springer.
34. Jacques Patarin, Louis Goubin, Nicolas Courtois: *Quartz,* 128*-bit long digital signatures*; Cryptographers' Track Rsa Conference 2001, LNCS 2020, pp.282-297, Springer. See [35] for the updated Quartz specification.
35. Jacques Patarin, Louis Goubin, Nicolas Courtois: *Quartz,* 128*-bit long digital signatures*; An updated version of Quartz specification. Available at `http://www.minrank.org/quartz-b.pdf` and at `http://www.cryptonessie.org`. The official web page of Quartz is `http://www.minrank.org/quartz/`.
36. D. Pointcheval, J. Stern: *Security arguments for Digital signatures and Blind Signatures,* Journal of Cryptology, Vol.13(3), Summer 2000, pp. 361-396. *Efficient signature schemes based on birational permutations,*

37. Nicolas Courtois, Adi Shamir, Jacques Patarin, Alexander Klimov, *Efficient Algorithms for solving Overdefined Systems of Multivariate Polynomial Equations*, Eurocrypt'2000, LNCS 1807, Springer, pp. 392-407. An extended version is available from `http://www.hfe.info`
38. Adi Shamir, Aviad Kipnis: *Cryptanalysis of the HFE Public Key Cryptosystem*; Crypto'99.

A About Security Proofs for Short Signature Schemes

A.1 What is a secure signature scheme

According to the most common requirements, initially proposed by Goldwasser, Micali and Rivest, a secure signature scheme should be existentially unforgeable against chosen-message attacks, see [25, 36, 3]. There are two interpretations of non-forgeable:

Unforgeability: The adversary should be able to compute a valid pair (message, signature) for a new message that has not been signed by the signature oracle (that is simulating the legitimate signer). It does not matter if he is able to produce another signature for the same message and it is not required at all that two message should not have the same signature.

Strong Unforgeability: The adversary should be able to compute a valid pair (message, signature), even for an old message that has already been legitimately signed by the signature oracle. In this case, one needs to prevent also computing a different signature for the same message (a.k.a. duplicate signatures, see [3]). However, it is still not a problem at all, should two messages have the same signature.

Discussion. There is some controversy whether unforgeability under Chosen-Message Attacks (CMA), the first version, would imply signature non-repudiation, but it certainly does imply message non-repudiation which is sufficient in practice. Most people seems to believe that unforgeability is enough, for digital signatures and that strong unforgeability and strong is too strong. We may define strong non-repudiation as strong unforgeability under CMA. Strong unforgeability may be required for some stronger primitives, for example in "signcryption" [1]. Indeed if the message have already been signed by the legitimate signer, all signatures are perfectly valid, there is no doubt about the fact that the the holder of the private key did indeed sign this message M, he cannot deny any responsibility.

In practice, some problems may however arise if for example, in a simplified payment system, where the user is only required to sign the time and amount of the transaction, the vendor may claim that the user ordered several identical items on the same day.

A.2 Quartz and Strong Unforgeability

In this paper, all the security proofs are done under No-Message Attacks (NMA) in which the adversary is only given the public key. In this case there is no difference between unforgeability and strong unforgeability.

In this paper we prove that Quartz is unforgeable against CMA with a concrete bound of $q^{m/2}$ which is insufficient in practice. It seems also that Quartz is unforgeable against CMA with exactly the same concrete lower bound $q^{m \cdot \frac{K}{K+1}}$ but this not certain and has not been proven so far.

In addition it seems that Quartz could be unforgeable against CMA with the (not satisfactory) concrete security bound $q^{m/2}$. However Quartz is NOT unforgeable against CMA in practice, and this bound can be met, as shown in [28]. Given a valid pair (message,signature), it is possible to compute a second signature, within $2^{k/2}$ computations, by a very simple method, that consists of finding a second pre-image for the last inverse G^{-1} in the signature computation process, see [28].

Below we prove our result on unforgeability of Quartz under NMA, and also propose a new, different way of doing short signatures for which a much better lower bound can be proven.

B Black-box Reductions For No-Message Attacks

In this section we study the security of some signature schemes based, as in Quartz on computation of one or several inverses $y \mapsto G^{-1}(y)$, provided that the trapdoor function G is difficult to inverse, as specified by the Strong One-wayness Assumption 5.0.8. We limit ourselves to no-message attacks, i.e. for the usual case with the adversary trying to forge a valid pair (message, signature) given (only) the public key (and no signature oracles). We need to build a black-box reduction, from the forger, to a machine that solves some difficult problem. It is done for a scheme that is similar to Quartz, but not identical: the signature is longer and it does not involve chaining of the G^{-1} as in Quartz. Later we will see if the result can be extended to Quartz.

B.1 The Black-Box Reduction

First of all, it is easy to prove the security of the usual signature scheme $\sigma = G^{-1}(H(M))$ in the random oracle model. The point is that the value $H(M)$ on which the trapdoor function G is inverted, is produced by the hash function. It does not only mean that it is completely random, and therefore G is inverted on random y, but also that the value may in fact be **chosen** by the random oracle, and as long its probability distribution is indistinguishable from random source, the adversary cannot say the difference. Therefore the adversary may as well be used to compute inverses of some **chosen** values: we have a black-box reduction that is achieved by replacing the random oracle by a given source (if it is random-looking). Similar black-box reduction works for signature schemes that compute several inverses and we have the following:

Theorem B.1.1 (Security reduction for signatures scheme that compute inverses of hashed values). Let G be a trapdoor one-way function. Assume that a signature scheme satisfies:

- it computes $G^{-1}(H_i)$ for some K values $H_i \in GF(q)^m$,
- all the H_i are independent parts of a (single) output of a hash function:
$$H(M) = (H_1, H_2, \dots, H_K)\,.$$
- all the K values $G^{-1}(H_1), \dots, G^{-1}(H_K)$ can be completely recovered in the signature verification,

If an attacker having (only) the access to the public key is able to compute a valid pair (message, signature) with probability ε, in random oracle model, and with Q queries to the hashing oracle.
Then it can be then transformed into a machine than given a random K-tuple $(y_1, y_2, \dots, y_K)$ will with probability ε/Q output all the K inverses: $(G^{-1}(y_1), G^{-1}(y_2), \dots, G^{-1}(y_K))$.

Proof. In the adversary's interaction with the hash oracle, the oracle gives a random and independent K-tuple $(H_1^{(i)}, H_2^{(i)}, \dots, H_K^{(i)})$ each time it is called for $i = 1 \dots Q$. We replace a randomly chosen K-tuple by $(y_1, y_2, \dots, y_K)$. Since both lists are random and independent, even a computationally unbounded adversary cannot distinguish between two situations. Consequently the result will be the same: he will, with probability ε output, a valid pair (message, signature), and with probability ε/Q it will contain the inverse of the chosen K-tuple.

B.2 Security Arguments

It turns out that our Strong One-wayness Assumption 5.0.8 is not enough.

Assumption B.2.1 (Super-Strong One-wayness of HFEv-).
Let K be a small integer. Let G a random public key $G \leftarrow \mathrm{HFEv}^-(q, n, m)$ with $m \approx n$, $n \geq m$ and $m \leq 100$. For any Adversary A running in T CPU clocks, given random K-tuple $(y_1, \dots, y_K) \leftarrow GF(q^m)^k$, the probability that A computes all the inverses $x_i = G^{-1}(y_i)$ with probability ε' is upper-bounded by:
$$\varepsilon' \leq (T/q^m)^K\,.$$

It is not a consequence of the Assumption 5.0.8. There may be algorithms that compute all K inverses $x_i = G^{-1}(y_i)$ faster than applying K times, the best algorithm to compute one inverse. Cf. also the footnote 10, p. 26.

Theorem B.2.2. Let K be a small integer. Let G be a trapdoor one-way function $G : GF(q)^n \rightarrow GF(q)^m$ with $m \leq n$. Assume that G satisfies the Assumption B.2.1. Assume that in a signature scheme, given a valid (pair message, signature), K inverses $G^{-1}(H_i)$ can be computed, with $H(M) = (H_1, H_2, \ldots, H_K)$. We assume that H is a random oracle. Let A be an Adversary, having (only) the access to the public key, running in time T, and able to compute a valid pair (message, signature) with a success probability ε. Then:

$$T \geq \varepsilon^{\frac{1}{K+1}} \cdot q^{m \cdot \frac{K}{K+1}}.$$

Proof. From Theorem B.1.1, we obtain a machine that inverts G in parallel for K random values $y_1, \ldots, y_K$ and with probability $\varepsilon' = \varepsilon/Q$. Following our Assumption, the attacker is able to do with probability ε/Q something that can only be done with a probability at most $(T/q^m)^K$. Thus:

$$\varepsilon/Q \leq (T/q^m)^K.$$

Finally, since the oracle queries take time, we have $Q \leq T$ and obtain:

$$\varepsilon/T \leq \varepsilon/Q \leq (T/q^m)^K, \text{ from this we get } T^{K+1} \geq \varepsilon(q^m)^K, \text{ and finally}$$

$$T \geq \varepsilon^{\frac{1}{K+1}} \cdot q^{m \cdot \frac{K}{K+1}}.$$

□

Corollary B.2.3. Let G be a trapdoor one-way function $G : GF(q)^n \rightarrow GF(q)^m$ with $m \leq n$. Assume that the only method known for to obtain $G^{-1}(y_i)$ for some values y_i is to guess them and apply G. Assume that in a signature scheme, given a valid pair (message, signature), K inverses $G^{-1}(H_i)$ are be computed, with $H(M) = (H_1, H_2, \ldots, H_K)$. Let A be an Adversary having (only) the access to the public key and running in time T that is able to compute a valid pair (message, signature). Then, under the random oracle assumption,

$$T \geq q^{m \cdot \frac{K}{K+1}}.$$

We obtained an exact, tight converse of the Theorem 9.2.1. It gives the exact security level of the described signature schemes (but not for Quartz).

B.3 Applications, $K = 1$, consequences on Flash and Sflash

All well known signature schemes constructed as $\sigma = G^{-1}(H)$ satisfy the assumption of the signature scheme we used in Theorems B.1.1 and B.2.2. However that trapdoor functions they use, for example RSA encryption, admit attacks asymptotically much faster than the exhaustive search, and therefore **do not** satisfy our very strong hardness Assumption 5.0.8 used in Theorems B.2.2.

Currently the only convincing trapdoor functions that seem to satisfy our Strong One-wayness Assumption 5.0.8 are multivariate cryptographic trapdoor functions of HFE family [31, 5] and only for small[10] q, see Section 6.

For the signature schemes Flash and Sflash [11], submitted to European call for cryptographic primitives [7, 8, 10, 11], things are less simple.

These schemes use a trapdoor function $G : GF(q)^{37} \rightarrow GF(q)^{26}$ with respectively $q = 256$ for Flash and $q = 128$ for Sflash-v2. Thus by the Theorem B.2.2 and the converse given by the Theorem 9.2.1 we have:

Corollary B.3.1 (Exact security of Flash and Sflash). If the trapdoor function used in Flash/Sflash satisfied the Strong One-wayness Assumption 5.0.8, the security of these respective signature schemes [7, 8, 10, 11] against no-message attacks would be exactly:

$$q^{m/2} = 2^{104} \text{ for Flash} \quad \text{ and } 2^{91} \text{ for Sflash-v2.}$$

Unfortunately, recent results [10, 11, 9] show that neither Flash, nor Sflash-v1, nor Sflash-v2 do satisfy the assumption. It is not true for the most recent Sflash-v3 either, see [10]. This however does not mean that Sflash-v3 is insecure, but the theory of this paper is more adapted to Quartz (designed to make very short signatures) than to Sflash (designed to make very fast signatures but as short as in Quartz).

B.4 Applications with $K \geq 2$, Differential Signature scheme

It is trivial to construct a signature scheme for which the Theorem B.2.2 applies, for any small K, and based on any trapdoor function. For this we need just to compute in parallel K signatures $\sigma_i = G^{-1}(H_i)$ and the signature will be given by a K-tuple $\left(G^{-1}(H_1(M)), \ldots, G^{-1}(H_K(M))\right)$. Unfortunately such a signature will be K times as long as when $K = 1$.

[10] Small means $q \leq 16$ or less. It is however not clear if $q = 2$ is the best choice: there is a tradeoff between the hardness of the basic one-way problem MQ, see Conjecture 4.0.7 and section 6, and the structural attacks from [5, 18, 26] mentioned in Section 8.2

[11] We refer here to the updated version of Sflash-v2 that is very similar to Flash, see [8, 10, 11].

Differential Signatures It is highly non-trivial to construct a signature scheme for which the Theorem B.2.2 applies, but with shorter signatures. For $K = 2$, it is possible to have a complete signature compression. For example we may compute the signature as follows:

$$\boxed{\sigma = G^{-1}(H_2) - G^{-1}(H_1)}$$

This surprising signature scheme has a non-trivial verification procedure. The signature compression is based on the fact that the two values $x = G^{-1}(H_1)$ and $x + \sigma = G^{-1}(H_2)$ can be completely recovered given their difference σ, as the equation $G(x+\sigma) - G(x) = H_2 - H_1$ becomes linear in the x_i after σ, H_1 and H_2 are known (the degree 2 terms will just cancel out). This scheme is called the "Differential Signature" scheme. It is not completely generic: it works only when G is a Multivariate Quadratic (MQ) function and only for $K = 2$.

Example: For example we may use the differential signature scheme with the following set of parameters for HFEv-:

$$\begin{cases} q = 2 \\ h = 127 \\ r = 7 \\ v = 1 \\ d = 257 \\ n = 128 \\ m = 120 \end{cases}$$

For these parameter values[12], we have:

Corollary B.4.1. Let $\varepsilon = 1$, $m = 120$, $n \geq m$, $K = 2$. Then the exact security of the differential signature scheme for no-message attacks is

$$Security = 2^{\frac{2}{3}m} = 2^{80}.$$

This follows immediately from Theorems B.2.2 and 9.2.1. The differential signature scheme allows, unlike Quartz, to have digital signatures of 128 bits with proven exact security of 2^{80} (for no-message attacks).

B.5 Application to Quartz and Similar Schemes

Unfortunately, in the construction used in Quartz as represented on Fig. 4 and more generally for generalized Feistel-Patarin with $K > 1$, only one of the values for which we have to compute the inverse G^{-1} is given by the hashing oracle. Our reduction, the Theorem B.1.1, cannot be applied for $K = 4$. We may however apply the Theorem B.1.1 and thus B.2.2

[12] We note that $h = 127$ is a prime as in Quartz (though yet no attacks are known when it isn't).

with $K = 1$, only considering the first inverse, that is indeed given by the random oracle. This shows that the security of this scheme is at least $q^{m/2}$, which is not very satisfactory, knowing that the the Theorem 9.2.1 gives an attack in $q^{m\frac{K}{K+1}}$. Concretely, our lower bound for the security of Quartz under the Assumption 5.0.8 is 2^{50}, and the upper bound is 2^{80}. They do unfortunately not coincide.

C Chosen-Message Security

The chosen-message security of schemes such as Quartz remains an open problem. In the extended version of this paper, available from the author, it is shown that, for some special trapdoor function, and without an additional assumption, they are not in general secure against such attacks. However, it is conjectured that if the signature is computed in deterministic way, with random coins being derived from the message by a pseudo-random generator using an additional secret key, such schemes should be secure also against this (the most general) class of attacks. Such a solution is used in Quartz.

C.1 Chosen-Message Security

In this section we will again consider the security of the generalized Feistel-Patarin signature meta-scheme used in Quartz, but against the most general, adaptively chosen-message attacks. We are in fact studying a very large class of known signature schemes including many versions of RSA: the "usual" way of computing digital signatures with a trapdoor function is just a special case of Generalized Feistel-Patarin signature scheme with $K = 1$. In the section B we showed that (modulo appropriate assumptions) such schemes can be (to some extend) proven secure against passive attacks with an adversary that only has access to the public key. We contend that in general they are not necessarily secure against known, or chosen message attacks.

The problem will arise for trapdoor functions, such as Rabin or multivariate schemes, that are not-bijective. Let Y be an uniform random variable in $GF(q)^m$. Let $G : GF(q^n) \rightarrow GF(q^m)$ with $n \geq m$. On average we have about q^{n-m} solutions, but the actual number varies. The problem depends very much of how in details, the meta-scheme deals with this. There are in fact two problems with this:

1. For example, if the root is chosen in a deterministic way, then let $E(y)$ be some (internal) values that decide which root is chosen. If E

is (partially or totally) known to the attacker, he may choose y, obtain $x = G^{-1}(y)$ from the signature oracle, and compute $E(y)$. Thus he will obtain triples $x, E(y), y$ that leak information about some internal values in the hidden structure of G. Suche triples might eventually help to recover this hidden structure and therefore break the trapdoor function G.

2. Making H non-deterministic does not help either, for example let G be a Rabin trapdoor function. If the adversary obtains two different signatures for the same message, with good probability he will be able to factor the modulus.

Clearly, we showed that without further precisions, the generalized Feistel-Patarin scheme is **not secure** in general, against chosen-message attacks. For Rabin trapdoor function, the problem arises when it is randomized and it seems that making the inverse deterministic solves the problem. However following point 1, for multivariate schemes, the deterministic solution is bad too.

C.2 Avoiding Chosen-Message Attacks

The suggested solution to the dilemma, already used in Quartz, is deterministic, in order to have always the same signature for a given message. It chooses one of the existing[13] solutions in a deterministic way, according to some pseudo-random function H. However, the difference is that H is (in a sense) secret. More precisely Quartz uses a cryptographic pseudorandom function with two parameters $H(y, \Delta)$, with a **secret** quantity Δ of 80 bits.

[13] In Quartz we have on average $q^{n-m} = 2^7 = 128$ solutions.