Gui: Revisiting Multivariate Digital Signature Schemes based on HFEv-

No Author Given

No Institute Given

Abstract. The QUARTZ digital signature scheme (Patarin, Courtois and Goubin, 2001) is one of the best known multivariate PKCs, based on an adaptation of "Hidden Field Equations with vinegar-minus" for very short signatures. Designed for a 80-bit security level, QUARTZ has no known flaws and is currently estimated to have a security level $\sim 2^{92}$.

QUARTZ was never widely used, probably due to its slow signing speed. The authors of QUARTZ had chosen ultra-safe parameters in 2001, based on what they knew about HFEv-. In this paper, we show how to choose parameters to speed up such schemes at 80- and 128-bit security levels given the new research on HFEv- security levels since then.

We show that reducing the degree of the central HFE polynomial, when combined with an appropriate increase in the number of Vinegar variables and minus equations, does not decrease the security of the scheme compared to the original QUARTZ design. This is backed up both with theory and with experiments. We achieve a speed-up of the signature generation process by two orders of magnitude. We call our new design Gui and show that the performance of Gui is comparable to that of standard signature schemes, including signatures on elliptic curves.

Keywords: Multivariate Cryptography, QUARTZ Signature Scheme, HFEv-, Direct Algebraic Attacks

1 Introduction

Cryptographic techniques are an essential tool to guarantee the security of communication in modern society. Today, the security of nearly all of the cryptographic schemes used in practice is based on number theoretic problems such as factoring large integers and solving discrete logarithms. The best known schemes in this area are RSA [25], DSA [16] and ECC. However, schemes like these will become insecure as soon as large enough quantum computers arrive. The reason for this is Shor's algorithm [26], which solves number theoretic problems like integer factorization and discrete logarithms in polynomial time on a quantum computer. Therefore, one needs alternatives to those classical public key schemes, based on hard mathematical problems not affected by quantum computer attacks.

Besides lattice, code and hash based cryptosystems, multivariate cryptography is one of the main candidates for this [1]. Multivariate schemes are in general very fast and require only modest computational resources, which makes them attractive for the use on low cost devices like smart cards and RFID chips [4,5]. Additionally, at least in the area of digital signatures, there exists a large number of practical multivariate schemes [10,17].

In 2001, Patarin and Courtois proposed a multivariate signature scheme called QUARTZ [21], which is based on the concept of HFEv-. While QUARTZ produces very short signatures (128 bit), the signature generation process is very slow (at the time about 11 seconds per signature [5]). The main reason for this is the use of a high degree HFE polynomial (for QUARTZ this degree is given by $D = 129$), which makes the inversion of the central map very costly.

At the time of the design of the QUARTZ scheme, very little was known about the complexity of algebraic attacks against the HFE family of systems, in particular, the HFEv- schemes. Therefore, the authors of QUARTZ could not base their parameter choice on theoretical foundations. Recently, there has been a fundamental breakthrough in terms of understanding the behavior of algebraic attacks on the HFE family of systems [8,9,11], which enables us to substantially improve the original design of QUARTZ without reducing its security. In this paper, we propose to choose new parameter sets for more efficient HFEv- based signature schemes for the low (80-bit) and high (112+-bit) security levels. We achieve this by reducing the degree of the central HFEv- polynomial while increasing the number of vinegar variables and minus equations.

Under state-of-the-art theoretical and experimental analysis, this adaptation should not cost us in terms of security, compared to conservative choices like the original QUARTZ design. Referring to a 3-legged Chinese utensil [28] dating back to earthenware pottery from the 4000-year-old Longshan culture, we call our new scheme Gui. We show that our new design speeds up the signature generation process by two degrees of magnitude compared to QUARTZ, and has comparable performance to standard signature schemes like RSA and ECDSA.

The rest of this paper is organized as follows. In Section 2 we give an introduction into the area of multivariate cryptography and in particular BigField signature schemes. Section 3 introduces the HFEv- signature scheme and the changes made to this scheme by Patarin and Courtois when defining QUARTZ. Furthermore, in this section, we give a short overview on the security and efficiency of QUARTZ. Section 4 presents the results of our experiments with direct attacks against low degree versions of HFEv-. Based on these results, we propose in Section 5 our new multivariate signature scheme Gui. Section 6 gives details on the implementation and compares the efficiency of Gui with that of some standard signature schemes. Finally, Section 7 concludes the paper.

2 Multivariate Cryptography

The basic objects of multivariate cryptography are systems of multivariate quadratic polynomials. The security of multivariate schemes is based on the

MQ Problem: Given m multivariate quadratic polynomials $p^{(1)}(\mathbf{x}), \ldots, p^{(m)}(\mathbf{x})$ in n variables $x_1, \ldots, x_n$, find a vector $\bar{\mathbf{x}} = (\bar{x}_1, \ldots, \bar{x}_n)$ such that $p^{(1)}(\bar{\mathbf{x}}) = \ldots = p^{(m)}(\bar{\mathbf{x}}) = 0$.
The MQ problem (for $m \approx n$) is proven to be NP-hard even for quadratic polynomials over the field GF(2) [14].

To build a public key cryptosystem based on the MQ problem, one starts with an easily invertible quadratic map $\mathcal{F} : \mathbb{F}^n \to \mathbb{F}^m$ (central map). To hide the structure of $\mathcal{F}$ in the public key, one composes it with two invertible affine (or linear) maps $\mathcal{S} : \mathbb{F}^m \to \mathbb{F}^m$ and $\mathcal{T} : \mathbb{F}^n \to \mathbb{F}^n$. The *public key* is therefore given by $\mathcal{P} = \mathcal{S} \circ \mathcal{F} \circ \mathcal{T}$. The *private key* consists of $\mathcal{S}$, $\mathcal{F}$ and $\mathcal{T}$ and therefore allows to invert the public key.

Note: Due to the above construction, the security of multivariate schemes is not only based on the MQ-Problem but also on the EIP-Problem ("Extended Isomorphisms of Polynomials") of finding the composition of $\mathcal{P}$.

In this paper we concentrate on multivariate signature schemes of the BigField family. For this type of multivariate schemes, the map $\mathcal{F}$ is a specially chosen easily invertible map over a degree n extension field $\mathbb{E}$ of $\mathbb{F}$. One uses an isomorphism

$\Phi : \mathbb{F}^n \rightarrow \mathbb{E}$ to transform $\mathcal{F}$ into a quadratic map

$$\bar{\mathcal{F}} = \Phi^{-1} \circ \mathcal{F} \circ \Phi \tag{1}$$

from $\mathbb{F}^n$ to itself. The public key of the scheme is therefore given by

$$\mathcal{P} = \mathcal{S} \circ \bar{\mathcal{F}} \circ \mathcal{T} = \mathcal{S} \circ \Phi^{-1} \circ \mathcal{F} \circ \Phi \circ \mathcal{T} : \mathbb{F}^n \rightarrow \mathbb{F}^n. \tag{2}$$

The standard signature generation and verification process of a multivariate BigField scheme works as shown in Figure 1.

Signature Generation

$$\begin{tikzcd}
& X \in \mathbb{E} \arrow[r, "\mathcal{F}^{-1}"] & Y \in \mathbb{E} \arrow[d, "\Phi^{-1}"] & \\
\mathbf{h} \in \mathbb{F}^n \arrow[r, "\mathcal{S}^{-1}"] & \mathbf{x} \in \mathbb{F}^n \arrow[u, "\Phi"] \arrow[r, "\bar{\mathcal{F}}^{-1}"] & \mathbf{y} \in \mathbb{F}^n \arrow[r, "\mathcal{T}^{-1}"] & \mathbf{z} \in \mathbb{F}^n \arrow[lll, bend left, "\mathcal{P}"]
\end{tikzcd}$$

Signature Verification

Fig. 1. General workflow of BigField schemes

Signature generation: To sign a message $\mathbf{h} \in \mathbb{F}^n$, one computes recursively $\mathbf{x} = \mathcal{S}^{-1}(\mathbf{h}) \in \mathbb{F}^n$, $X = \Phi(\mathbf{x}) \in \mathbb{E}$, $Y = \mathcal{F}^{-1}(X) \in \mathbb{E}$, $\mathbf{y} = \Phi^{-1}(Y) \in \mathbb{F}^n$ and $\mathbf{z} = \mathcal{T}^{-1}(\mathbf{y})$. The signature of the message $\mathbf{h}$ is $\mathbf{z} \in \mathbb{F}^n$.

Verification: To check the authenticity of a signature $\mathbf{z} \in \mathbb{F}^n$, one simply computes $\mathbf{h}' = \mathcal{P}(\mathbf{z}) \in \mathbb{F}^n$. If $\mathbf{h}' = \mathbf{h}$ holds, the signature is accepted, otherwise rejected.

A good overview on existing multivariate schemes can be found in [7]. Two widely used variations of multivariate BigField schemes are the Minus variation and the use of additional (Vinegar) variables.

Minus-Variation: The idea of this variation is to remove a small number of equations from the public key. The Minus-Variation was first used in schemes like SFLASH [22] to prevent Patarins Linearization Equations attack [23] against the Matsumoto-Imai cryptosystem [20].

Vinegar-Variation: The idea of this variation is to parametrize the central map $\mathcal{F}$ by adding (a small set of) additional (Vinegar) variables. In the context of multivariate BigField schemes, the Vinegar variation can be used to increase the security of the scheme against direct and rank attacks.

3 The QUARTZ Signature Scheme

QUARTZ is a multivariate signature scheme standardized by Patarin and Courtois in [21]. Roughly speaking, it is an HFEv- scheme with a specially designed signature generation process to enable secure short signatures of length 128 bit.

3.1 The HFEv- Signature Scheme

Let $\mathbb{F} = \mathbb{F}_q$ be a finite field with q elements and $\mathbb{E}$ be a degree n extension field of $\mathbb{F}$. Furthermore, we choose integers D, a and v. Let Φ be the canonical isomorphism between $\mathbb{F}^n$ and $\mathbb{E}$, i.e.

$$\Phi(x_1, \ldots, x_n) = \sum_{i=1}^{n} x_i \cdot X^{i-1}. \tag{3}$$

The central map $\mathcal{F}$ of the HFEv- scheme is a map from $\mathbb{E} \times \mathbb{F}^v$ to $\mathbb{E}$ of the form

$$\begin{aligned}\mathcal{F}(X) = & \sum_{0 \le i \le j}^{q^i+q^j \le D} \alpha_{ij} \cdot X^{q^i+q^j} \\ & + \sum_{i=0}^{q^i \le D} \beta_i(v_1, \ldots, v_v) \cdot X^{q^i} \\ & + \gamma(v_1, \ldots, v_v),\end{aligned} \tag{4}$$

with $\alpha_{ij} \in \mathbb{E}$, $\beta_i : \mathbb{F}^v \to \mathbb{E}$ being linear and $\gamma : \mathbb{F}^v \to \mathbb{E}$ being a quadratic function.
Due to the special form of $\mathcal{F}$, the map $\bar{\mathcal{F}} = \Phi^{-1} \circ \mathcal{F} \circ \Phi$ is a quadratic polynomial map from $\mathbb{F}^{n+v}$ to $\mathbb{F}^n$. To hide the structure of $\bar{\mathcal{F}}$ in the public key, one combines it with two affine (or linear) maps $\mathcal{S} : \mathbb{F}^n \to \mathbb{F}^{n-a}$ and $\mathcal{T} : \mathbb{F}^{n+v} \to \mathbb{F}^{n+v}$ of maximal rank.

The *public key* of the scheme is the composed map $\mathcal{P} = \mathcal{S} \circ \bar{\mathcal{F}} \circ \mathcal{T} : \mathbb{F}^{n+v} \to \mathbb{F}^{n-a}$, the *private key* consists of $\mathcal{S}$, $\mathcal{F}$ and $\mathcal{T}$.

Signature generation: To generate a signature for a message $\mathbf{h} \in \mathbb{F}^{n-a}$, the signer performs the following three steps.

1. Compute a preimage $\mathbf{x} \in \mathbb{F}^n$ of $\mathbf{h}$ under the affine map $\mathcal{S}$.
2. Lift $\mathbf{x}$ to the extension field $\mathbb{E}$ (using the isomorphism Φ). Denote the result by X.
   Choose random values for the vinegar variables $v_1, \ldots, v_v \in \mathbb{F}$ and compute $\mathcal{F}_V = \mathcal{F}(v_1, \ldots, v_v)$.
   Solve the univariate polynomial equation $\mathcal{F}_V(Y) = X$ by Berlekamp's algorithm and compute $\mathbf{y}' = \Phi^{-1}(Y) \in \mathbb{F}^n$.
   Set $\mathbf{y} = (\mathbf{y}'||v_1|| \ldots ||v_v)$.
3. Compute the signature $\mathbf{z} \in \mathbb{F}^{n+v}$ by $\mathbf{z} = \mathcal{T}^{-1}(\mathbf{y})$.

Signature verification: To check the authenticity of a signature $\mathbf{z} \in \mathbb{F}^{n+v}$, one simply computes $\mathbf{h}' = \mathcal{P}(\mathbf{z}) \in \mathbb{F}^{n-a}$. If $\mathbf{h}' = \mathbf{h}$ holds, the signature is accepted, otherwise rejected.

3.2 QUARTZ

Patarin and Courtois suggested the following parameters for QUARTZ:

$$(\mathbb{F}, n, D, a, v) = (\mathrm{GF}(2), 103, 129, 3, 4).$$

Due to this choice, the public key $\mathcal{P}$ of QUARTZ is a quadratic map from $\mathbb{F}^{107}$ to $\mathbb{F}^{100}$. The public key size of QUARTZ is 71 kB, the private key size 3 kB.
To avoid birthday attacks, Patarin and Courtois developed a special procedure for the signature generation process of QUARTZ. Roughly spoken, one computes four HFEv- signatures (for the messages $\mathbf{h}$, $\mathcal{H}(\mathbf{h}||0x00)$, $\mathcal{H}(\mathbf{h}||0x01)$ and $\mathcal{H}(\mathbf{h}||0x02)$) and combines them to a single 128 bit signature of the message $\mathbf{h}$. Analogously, during the signature verification process, one has to use the public key $\mathcal{P}$ four times.

3.3 Security

Despite of its rather complicated signature generation process, breaking the QUARTZ scheme is still equivalent to breaking the underlying HFEv-scheme. The most important attacks against this scheme are

- the MinRank attack and
- direct algebraic attacks.

The MinRank attack on HFE In this paragraph we describe the attack of Kipnis and Shamir [18] against the HFE cryptosystem. For the simplicity of our description we restrict ourselves to homogeneous maps $\mathcal{F}$ and $\mathcal{P}$.

The key observation of the attack is to lift the maps $\mathcal{S}$, $\mathcal{T}$ and $\mathcal{P}$ to functions $\mathcal{S}^\star$, $\mathcal{T}^\star$ and $\mathcal{P}^\star$ over the extension field $\mathbb{E}$. Since $\mathcal{S}$ and $\mathcal{T}$ are linear maps, $\mathcal{S}^\star$ and $\mathcal{T}^\star$ have the form

$$\mathcal{S}^\star(X) = \sum_{i=1}^{n-1} s_i \cdot X^{q^i} \quad \text{and} \quad \mathcal{T}^\star(X) = \sum_{i=1}^{n-1} t_i \cdot X^{q^i}, \tag{5}$$

with coefficients s_i and $t_i \in \mathbb{E}$. The function $\mathcal{P}^\star$ can be expressed as

$$\mathcal{P}^\star(X) = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} p^\star_{ij} X^{q^i+q^j} = \underline{X} \cdot P^\star \cdot \underline{X}^T, \tag{6}$$

where $P^\star = [p^\star_{ij}]$ and $\underline{X} = (X^{q^0}, X^{q^1}, \ldots, X^{q^{n-1}})$. Due to the relation $\mathcal{P}^\star(X) = \mathcal{S}^\star \circ \mathcal{F} \circ \mathcal{T}^\star(X)$ we get $\mathcal{S}^{\star\ -1} \circ \mathcal{P}^\star(X) = \mathcal{F} \circ \mathcal{T}^\star(X)$ and

$$\tilde{P} = \sum_{k=0}^{n-1} s_k \cdot G^{\star k} = W \cdot F \cdot W^T \tag{7}$$

with ${g^\star_{ij}}^k = ({p^\star}_{i-k \bmod n, j-k \bmod n})^{q^k}$, $w_{ij} = s^{q^i}_{j-i \bmod n}$ and F being the $n \times n$ matrix representing the central map $\mathcal{F}$. Note that, due to the special structure of $\mathcal{F}$, the only non zero entries in the matrix F are located in the upper left $r \times r$ submatrix ($r = \lfloor \log_q D - 1 \rfloor + 1$).

Since the rank of the matrix $W \cdot F \cdot W^T$ is less or equal to r, we can determine the coefficients s_k of equation (7) by solving an instance of the MinRank problem.

In the setting of HFEv-, the rank of this matrix can, for odd characteristic, be bounded from above by [11]

$$\text{Rank}(\widetilde{P}) \leq r + a + v. \tag{8}$$

Under the assumption that the vinegar maps β_i look like random functions, we find that this bound is tight. For fields of even characteristic we eventually have to decrease this rank by 1, since over those fields, the matrix $\widetilde{P}$ is always of even rank. The complexity of the MinRank attack against QUARTZ like schemes is therefore given roughly by

$$\text{Complexity}_{\text{MinRank}} = \mathcal{O}(q^{n\cdot(r+v+a-1)} \cdot (n-a)^3). \tag{9}$$

Direct attacks For the HFE family of schemes, the direct attack, namely the attack by directly solving the public equation $\mathcal{P}(\mathbf{x}) = \mathbf{h}$ by an algorithm like XL or a Gröbner basis method such as F_4 [12] is a major concern due to which happened to HFE challenge 1 [13]. At the time of the design of the QUARTZ scheme, very little was known theoretically about the complexity of algebraic attacks against the HFE family of systems, in particular, the HFEv- schemes. The authors of QUARTZ did not actually give an explanation of their selection of the parameters and therefore the parameter selection of their scheme was not supported by theoretical results. We need to point out that, as has been shown by experiments [19], the public systems of HFEv- can be solved easier than random systems. Recently, there has been a fundamental breakthrough in terms of understanding how algebraic attacks on the HFE family of systems [8,9,11] work. In particular, we now have a solid insight what happens in the case of HFEv-. An upper bound for the degeneration degree of a Gröbner Basis attack against HFEv- is given by [11]

$$\mathrm{d}_{\text{reg}} \leq \begin{cases} \frac{(q-1)\cdot(r-1+a+v)}{2} + 2 & q \text{ even and } r+a \text{ odd} \\ \frac{(q-1)\cdot(r+a+v)}{2} + 2 & \text{otherwise} \end{cases}, \tag{10}$$

where r is given by $r = \lfloor \log_q(D-1) \rfloor + 1$.

Note: In [6] Courtois et al. estimated the complexity of a direct attack on QUARTZ by 2^{74} operations. However, they underestimated the degree of regularity of solving an HFEv- system drastically.

3.4 Efficiency

The most costly step during the signature generation process of QUARTZ is the inversion of the univariate polynomial equation $\mathcal{F}$ over the extension field $\mathbb{E}$. This step is usually performed by Berlekamp's algorithm, whose complexity can be estimated by [24]

$$\mathcal{O}(D^3 + n \cdot D^2). \tag{11}$$

Due to the high degree of the HFEv- polynomial used in QUARTZ, the inversion of $\mathcal{F}$ is very costly. Furthermore, we have to perform this step four times during the signature generation of QUARTZ. Additionally, the design of QUARTZ requires the central equation $\mathcal{F}(Y) = X$ to have a unique root. Since, after choosing random values for Minus equations and Vinegar variables, $\mathcal{F}$ can be seen as a random function, this requires about e trials to obtain a signature. Thus, the QUARTZ signature scheme is rather slow and it takes about 11 seconds to generate a signature [5].

The theoretical breakthrough mentioned above indicates that it might be possible to substantially improve the original design of QUARTZ without reducing the security of the scheme, if we adapt the number of Minus equations and Vinegar variables in an appropriate way. By reducing the degree of the central HFEv- polynomial we can speed up the operations of Berlekamp's algorithm and therefore the signature generation process of the HFEv- scheme.

4 The New HFEv- design

The first question we want to answer in this section is the following: How should we choose the degree D of the central HFEv- polynomial for Gui? A small D will speed up the scheme, but choosing D too small might bring the security of the scheme into jeopardy.

Surely, $D = 2$ ("Square" systems) or 3 seems to be a bad choice, since such small values of D would lead to maps $\mathcal{F}$ of rank 2. For $D = 5$ and $D = 7$ the same can be achieved by a linear transformation of the variables. So, the smallest value of D we feel comfortable to use is $D = 9$, which leads to matrices of rank 4. Another promising value for D is $D = 17$, which leads to matrices of rank 5.

In this section we present the results of our experiments of running direct attacks on HFEv- schemes with low degree central maps. By our experiments we want to answer the following questions:

- Is the upper bound on the degree of regularity given by equation (10) reasonable tight?
- Can we, for our choices of D, find appropriate values of a and v such that the HFEv- scheme is still intractable?

In our experiments, we used $\mathbb{F} = \mathrm{GF}(2)$ as the underlying field. The degree of the HFEv- polynomial was chosen to be $D \in \{9, 17\}$ (see above). The corresponding values of r are 4 and 5 respectively. Furthermore, we set for simplicity reasons $a = v$.
We created instances of the HFEv- signature scheme for our two values of D and $a = v \in \{0, \ldots, 5\}$ with MAGMA. Then we fixed $a + v$ of the variables to get determined systems and tried to solve them with the F_4 algorithm integrated in MAGMA.
As we found, it is very important to add the field equations ($x_i^2 - x_i = 0$ ($i = 1, \ldots, n - a$)) to the system before running the F_4 algorithm. Table 1 and Figure 2 show the results of our experiments.

As our experiments show, the upper bound on the degree of regularity given by equation (10) is relatively tight. In fact, for $a = v = 3$ we could reach the upper bound for $n \geq 41$ ($D = 9$) and $n \geq 43$ ($D = 17$) respectively. However, shortly after having reached this degree, we ran out of memory. Furthermore we saw that for both the parameter sets $(D, a, v) = (9, 5, 5)$ and $(D, a, v) = (17, 4, 4)$, HFEv- systems with more than 31 equations have a degree of regularity of ≥ 7 (although we do not have enough memory to solve these systems completely).

5 The New Multivariate Signature Scheme Gui

Based on our experiments presented in the previous section we propose two different versions of our HFEv- based signature scheme over the field GF(2):

- Gui-95 with $(n, D, a, v) = (95, 9, 5, 5)$ with 90 equations in 100 variables and
- Gui-94 with $(n, D, a, v) = (94, 17, 4, 4)$ with 90 equations in 98 variables

The complexity of direct attacks against these two schemes can be estimated as follows.
According to our experiments, the degree of regularity of the F_4 algorithm against these schemes will be at least 7.
The number T of top-level monomials in the solving step of the F_4 algorithm is therefore given by

$$T = \binom{n-a}{d_{\text{reg}}} \geq \binom{90}{7} \geq 2^{33.6}$$

The number of non zero elements in each row can be estimated by $\tau = \binom{n-a}{2} > 2^{12}$. Therefore we get for the complexity of a direct attack against one of our schemes

$$\text{Complexity}_{F_4/F_5} \geq 3 \cdot \tau \cdot T^2 > 2^{80.7}. \qquad (12)$$

Note that this number is very optimistic since we assume that the degree of regularity will not rise above 7. Additionally, for better comparison to standard signature schemes, we propose a third version of Gui, Gui-127, with the parameters $(n, D, a, v) = (127, 9, 4, 6)$, providing design security levels of 120 bits.

Similarly to QUARTZ, we must repeat the HFEv- core map several times to avoid birthday attacks against Gui. One has to sign different hash values of the same message and combine the outputs into a single signature. Here, we follow closely the design of QUARTZ and apply the core HFEv- operation 4, 3 and 4 times respectively for Gui-94, -95 and -127. The resulting key and signature sizes for our scheme can be seen from Table 3.

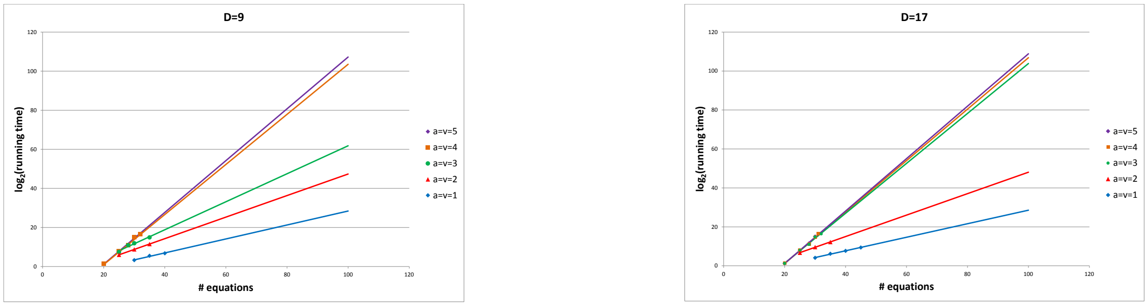

Fig. 2. Results of our experiments with the F_4 algorithm against HFEv- schemes $(D \in \{9, 17\})$

		# equations	20	25	30	32
HFEv-system D=9	$a = v = 3$	theoretical degree of regularity (formula (10)) ≤ 6				
		parameters (n,D,a,v)	(23,9,3,3)	(28,9,3,3)	(33,9,3,3)	(35,9,3,3)
		d_{reg}	5	5	5	5
		time (s)	2.6	179	2,899	28,746
	$a = v = 4$	theoretical degree of regularity (formula (10)) ≤ 8				
		parameters (n,D,a,v)	(24,9,4,4)	(29,9,4,4)	(34,9,4,4)	(36,9,4,4)
		d_{reg}	5	6	6	6
		time (s)	2.7	244	31,537	102,321
	$a = v = 5$	theoretical degree of regularity (formula (10)) ≤ 8				
		parameters (n,D,a,v)	(25,9,5,5)	(30,9,5,5)	(35,9,5,5)	(47,9,5,5)
		d_{reg}	5	6	6	7
		time (s)	2.8	255	32,481	ooM [1]
HFEv-system D=17	$a = v = 3$	theoretical degree of regularity (formula (10)) ≤ 7				
		parameters (n,D,a,v)	(23,17,3,3)	(28,17,3,3)	(33,17,3,3)	(35,17,3,3)
		d_{reg}	5	6	6	6
		time (s)	2.4	245	28,768	87,726
	$a = v = 4$	theoretical degree of regularity (formula (10)) ≤ 8				
		parameters (n,D,a,v)	(24,17,4,4)	(29,17,4,4)	(34,17,4,4)	(36,17,4,4)
		d_{reg}	5	6	6	7
		time (s)	2.4	248	31,911	ooM [1]
	$a = v = 5$	theoretical degree of regularity (formula (10)) ≤ 9				
		parameters (n,D,a,v)	(25,17,5,5)	(30,17,5,5)	(35,17,5,5)	(37,17,5,5)
		d_{reg}	5	6	6	7
		time (s)	2.4	250	32,350	ooM [1]
for comparison: random system		d_{reg}	5	6	6	7
		time (s)	3.5	310	32,533	ooM [1]

[1] out of memory

Table 1. Results of our experiments with the F_4 algorithm against HFEv- schemes ($D \in \{9, 17\}$)

6 Implementation and Comparison

6.1 Arithmetics over Finite Fields

Large binary field operations with PCLMULQDQ Time records for arithmetic operations over large binary fields were changed since the emerging of new instructions of carry-less multiplication: `PCLMULQDQ` [27]. Although we would rely on `PCLMULQDQ` to take the burden off from multiplication, in some cases old tricks help to avoid the use of `PCLMULQDQ` and therefore its long latency even in the latest Intel Haswell cpu (lantency 7 cycles, throughput 2 cycles)[15].

In our targeted field size ($\leq$ 128 bit), we choose Karatsuba algorithm to avoid 3 calls of `PCLMULQDQ` in the multiplication phase. The reduction phase of the field multiplication is heavily related to the field representation. For the original QUARTZ (GF(2^{103})) the authors used GF(2^{103}) := GF(2)$[x]/(x^{103}+x^9+1)$ [21]. For Gui, we choose the field representations GF(2^{94}) := GF(2)$[x]/(x^{94} + x^{21} + 1)$, GF($2^{95}$) := GF(2)$[x]/(x^{95} + x^{11} + 1)$ and GF(2^{127}) := GF(2)$[x]/(x^{127} + x + 1)$ respectively.

In the GF(2^{127}) case, for example, reduction costs only 2 128-bit shifts for the x^{128} part and 1 conditional XOR for x^{127}, avoiding at least 2 calls of `PCLMULQDQ` while reducing the high 128 bit register.

6.2 Inverting the HFEv- core

For inverting the central HFE equation $\mathcal{F}(Y) = Y^D + \sum_{0\le i\le j, 2^i+2^j<D} a_{ij} Y^{2^i+2^j} = X$, we have to perform Berlekamp's algorithm to find its roots. Since the design of QUARTZ and Gui requires $\mathcal{F}$ to have a unique solution (probability $1/e$ for a random X), we will only need to perform the first step of Berlekamp's algorithm:

$$\begin{aligned} &\gcd(\mathcal{F}(Y) - X, Y^{2^n} - Y) \\ &= \gcd(\mathcal{F}(Y) - X, \prod_{i\in\mathbb{F}_{2^n}, i\neq 0} (Y - i)) \\ &= \prod_{i:\mathcal{F}(i)=X} (Y - i), \end{aligned}$$

and therefore the main process in creating a signature is to compute $\gcd(\mathcal{F}(Y) - X, Y^{2^n} - Y)$. The number of roots to $\mathcal{F}(Y) = X$ (as well as the only solution when that happens) can obviously be read off from the result.

How do we optimize the computation of the gcd? The main computation consumption in this step comes from the division of the extreme high power polynomial $Y^{2^n} - Y \mod \mathcal{F}(\mathcal{Y})$. Naive long division is unacceptable due to its slow reduction pace. Instead, we choose to recursively raise the lower degree polynomial Y^{2^m} to the power of 2.

$$\begin{aligned} &(Y^{2^m} \bmod \mathcal{F}(Y))^2 \bmod \mathcal{F}(Y) \\ &= (\sum_{i<2^m} b_i Y^i)^2 \bmod \mathcal{F}(Y) \\ &= (\sum_{i<2^m} b_i^2 Y^{2i}) \bmod \mathcal{F}(Y) \end{aligned}$$

By multiplying Y to the naive relation $Y^D = \sum_{0\le i\le j, 2^i+2^j<D} a_{ij} Y^{2^i+2^j}$ from $\mathcal{F}(Y)$, we can prepare a table for $Y^{2i} \bmod \mathcal{F}(Y)$ first. The rest of computation of raising is to square all coefficients b_i in $Y^{2^m} \bmod \mathcal{F}(Y)$ and multiply them to the Y^{2i}s in the table. Although the starting relation $\mathcal{F}(Y) = Y^D + \sum_{0\le i\le j, 2^i+2^j<D} a_{ij} Y^{2^i+2^j}$ is a sparse polynomial, the polynomials become dense quickly while raising and the number of terms is restricted by D because of mod $\mathcal{F}(Y)$. We expect the number of terms be in average D during the computation. The number of field multiplications needed to compute the Y^{2i} table is $O(2 \cdot D^2)$. The number of field multiplications needed for raising Y^{2^m} to Y^{2^n} is $O((n-m) \cdot D\text{: squ} + (n-m) \cdot D^2\text{: mul})$. We can further reduce the number of computations by raising the Y^{2^m} with a higher degree Y^i table. For example, if we raise Y^{2^m} to $Y^{2^{4m}}$ in one step, the number of multiplications becomes $O((n-m) \cdot D\text{: squ} + \frac{(n-m)}{2} \cdot D^2\text{: mul})$, but the computation for preparing the Y^i table increases. Table 2 shows the time needed to compute $\gcd(Y^{2^n} - Y, \mathcal{F}(Y))$ on two different CPUs.

scheme	security level (bit)	time needed for inverting $\mathcal{F}$ (kilo-cycles) [1]
HFEv-(95,9,5,5)	80	159 / 135
HFEv-(94,17,4,4)	80	533 / 453
HFEv-(103,129,3,4)	80	25,793 / 20,784
HFEv-(127,9,4,6)	123	170 / 156

[1] [1] running time on AMD Opteron 6212, 2.5 GHz / Intel Xeon CPU E5-2620, 2.0 GHz

Table 2. Time to compute $\gcd(Y^{2^n} - Y, \mathcal{F}(Y))$ for HFEv- instances

6.3 Experiments and Comparison

Table 3 shows key sizes and running time for signature generation and verification of Gui and compares these data with those of some standard signature schemes. The data are benchmarked according to specifications given by the eBACS project [3].

scheme	security	public key	private key	signature	signing time	verify time
	level(bits)	size (Bytes)	size (Bytes)	size(bits)	(k-cycles) [1]	(k-cycles) [1]
Gui-95 (95,9,5,5)	80	60,600	3,053	120	1,479/1,186	325/230
Gui-94 (94,17,4,4)	80	58,212	2,943	124	4,945/5,421	357/253
Gui-127 (127,9,4,6)	123	142,576	5,350	160	1,966/1,249	707/427
QUARTZ (103,129,3,4)	80	75,514	3,774	128	167,485/168,266	375/235
RSA-1024	80	128	128	128	2,080/1,058	74/ 32
RSA-2048	112	256	256	256	8,834/5,347	138/ 76
ECDSA P160	80	40	60	320	1,283/ 558	1,448/635
ECDSA P192	96	48	72	384	1,513/ 773	1,715/867
ECDSA P256	128	64	96	512	830/ 388	2,111/920

[1] Tested on AMD Opteron 6212, 2.5 GHz / Intel Xeon CPU E5-2620, 2.0 GHz

Table 3. Comparison between Gui and standard signature schemes

We should note that the timings for Gui given by Table 3 are for C programs with a few intrinsic function calls to invoke `PCLMULQDQ`. The PKCs benchmarked in the eBACs project (represented by the eBATs programs as their implementations) also do not represent optimal implementations of RSA and ECC. We present these numbers in an effort to compare apples to apples by using only reference implementations.

7 Conclusion and Future Work

In this paper, we analyzed the behavior of direct attacks against the HFEv- signature scheme. Experiments show that the upper bound on the degree of regularity found by Ding and Yang in [11] is relatively tight. Based on our results, we can give new recommendations on the parameter choice of HFEv- based schemes, by which it is possible to speed up the signature generation process by two orders of magnitude (nearly 150×) compared to the original QUARTZ. We name our new design Gui and show that the running time of Gui is comparable to that of standard signature schemes, including signatures on elliptic curves.

The most obvious future work would be to create for every common existing platform an optimal implementation of HFEv- (Gui) and compare it with some of the best optimized code for ECC and RSA, such as Ed25519 [2]. The other would be to verify such optimal Gui code for formal correctness. In short, we believe that there is still much to be done about the HFEv- digital signature schemes.

References

1. D.J. Bernstein, J. Buchmann, E. Dahmen (eds.): Post Quantum Cryptography. Springer, 2009.
2. D. J. Bernstein, N. Duif, T. Lange, P. Schwabe, and B.-Y. Yang, High-speed high-security signatures, J. Cryptographic Engineering 2:2, pp. 77-89 (2012).
3. D.J. Bernstein and T. Lange (editors). eBACS: ECRYPT Benchmarking of Cryptographic Systems. http://bench.cr.yp.to, accessed 14 May 2014.
4. A. Bogdanov, T. Eisenbarth, A. Rupp, C. Wolf. Time-area optimized public-key engines: MQ-cryptosystems as replacement for elliptic curves? CHES 2008, LNCS vol. 5154, pp. 45-61. Springer, 2008.
5. A.I.T. Chen, M.-S. Chen, T.-R. Chen, C.-M. Cheng, J. Ding, E. L.-H. Kuo, F. Y.-S. Lee, B.-Y. Yang. SSE implementation of multivariate PKCs on modern x86 cpus. CHES 2009, LNCS vol. 5747, pp. 33 - 48. Springer, 2009.
6. N.T. Courtois, M. Daum, P. Felke: On the Security of HFE, HFEv- and QUARTZ. PKC 2003, LNCS vol. 2567, pp. 337 - 350. Springer 2003.

7. J. Ding, J. E. Gower, D. S. Schmidt: Multivariate Public Key Cryptosystems. Springer, 2006.
8. J. Ding, T. Hodges: Inverting HFE Systems Is Quasi-Polynomial for All Fields. CRYPTO 2011, LNCS vol. 6841, pp. 724-742. Springer 2011.
9. J. Ding, T. Kleinjung: Degree of regularity for HFE-. IACR eprint 2011/570.
10. J. Ding, D. S. Schmidt: Rainbow, a new multivariate polynomial signature scheme. ACNS 2005, LNCS vol. 3531, pp. 164-175. Springer 2005.
11. J. Ding, B.Y. Yang: Degree of Regularity for HFEv and HFEv-. PQCrypto 2013, LNCS vol. 7932, pp. 52-66. Springer, 2013.
12. J.C. Faugère: A new efficient algorithm for computing Gröbner bases (F4). Journal of Pure and Applied Algebra 139, pp. 61-88 (1999).
13. J.C. Faugère, A. Joux: Algebraic Cryptanalysis of Hidden Field Equation (HFE) Cryptosystems Using Gröbner Bases. CRYPTO 2003, LNCS vol. 2729, pp. 44-60. Springer 2003.
14. M. R. Garey and D. S. Johnson: Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman and Company 1979.
15. Intel Corporation, Hashwell Cryptographic Performance. http://www.intel.com/content/dam/www/public/us/en/documents/white-papers/haswell-cryptographic-performance-paper.pdf
16. D. Kravitz: Digital Signature Algorithm. US patent 5231668 (July 1991).
17. A. Kipnis, L. Patarin, L. Goubin: Unbalanced Oil and Vinegar Schemes. EUROCRYPT 1999, LNCS vol. 1592, pp. 206–222. Springer 1999.
18. A. Kipnis, A. Shamir: Cryptanalysis of the HFE Public Key Cryptosystem. CRYPTO 99, LNCS vol. 1666, pp. 19 - 30. Springer 1999.
19. M. S. E. Mohamed, J. Ding and J. Buchmann: Towards Algebraic Cryptanalysis of HFE Challenge 2, ISA 2011, Communications in Computer and Information Science vol. 200, pp. 123-131. Springer 2011.
20. T. Matsumoto, H. Imai: Public quadratic polynomial-tuples for efficient signature-verification and message-encryption. EUROCRYPT 1988. LNCS vol. 330, pp. 419-453. Springer 1988.
21. J. Patarin, N. Courtois, L. Goubin: QUARTZ, 128-Bit Long Digital Signatures. CTRSA 2001, LNCS vol. 2020, pp. 282-297. Springer, 2001.
22. J. Patarin, N. Courtois, L. Goubin: Flash, a fast multivariate signature algorithm. CTRSA 2001, LNCS vol. 2020, pp. 298 - 307. Springer, 2001.
23. J. Patarin: Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt 88. CRYPTO 95. LNCS vol. 963, pp. 248 - 261. Springer 1995.
24. C. Richards: Algorithms for Factoring Square-Free Polynomials over Finite Fields. Master Thesis, Simon Fraser University (Canada), 2009.
25. R. L. Rivest, A. Shamir, L. Adleman: A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Commun. ACM 21 (2), pp. 120-126 (1978).
26. P. Shor: Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer, SIAM J. Comput. 26 (5), pp. 1484 - 1509 (1997).
27. J. Taverne, A. Faz-Hernàndez, D. F. Aranha, F. Rodrquez-Henrïquez, D. Hankerson, J. López: Software Implementation of Binary Elliptic Curves: Impact of the Carry-Less Multiplier on Scalar Multiplication. CHES 2011, LNCS vol. 6917, pp. 108-123. Springer 2011.
28. http://en.wikipedia.org/wiki/File:CMOC_Treasures_of_Ancient_China_exhibit_white_pottery_gui_1.jpg, Wikipedia.