Square-Vinegar Signature Scheme

John Baena[1,2], Crystal Clough[1], and Jintai Ding[1]

[1] Department of Mathematical Sciences,
University of Cincinnati,
Cincinnati, OH, 45220, USA
{baenagjb,cloughcl}@email.uc.edu,ding@math.uc.edu
http://math.uc.edu

[2] Department of Mathematics,
National University of Colombia,
Medellin, Colombia

Abstract. We explore ideas for speeding up HFE-based signature schemes. In particular, we propose an HFEv$^-$ system with odd characteristic and a secret map of degree 2. Changing the characteristic of the system has a profound effect, which we attempt to explain and also demonstrate through experiment. We discuss known attacks which could possibly topple such systems, especially algebraic attacks. After testing the resilience of these schemes against F4, we suggest parameters that yield acceptable security levels.

Keywords: Multivariate Cryptography, HFEv$^-$, Signature Scheme, Odd Characteristic.

1 Introduction

Multivariate public-key cryptosystems (MPKCs) stand among the systems thought to have the potential to resist quantum computer attacks [4]. This is because their main security assumption is based on the problem of solving a system of multivariate polynomial equations, a problem which is still as hard for a quantum computer to solve as a conventional computer [12,22].

The area of multivariate public-key cryptography essentially began in 1988 with an encryption scheme proposed by Matsumoto and Imai [17]. This system has since been broken [19], but has inspired many new encryption and signature schemes. One of these is HFE (Hidden Field Equations), proposed in 1996 by Patarin [20].

An HFE scheme could still be secure, but the parameters required would make it so inefficient as to be practically unusable. Many variants of HFE have been proposed and analyzed, in particular one called HFEv$^-$, a signature scheme which combines HFE with another system called Oil-Vinegar and also uses the "−" construction. More about HFEv$^-$ in Sect. 2.2. A recent proposal is Quartz, a signature scheme with HFEv$^-$ at its core. Quartz-7m, with slightly different parameter choices, is believed secure. These schemes have enticingly short signatures.

J. Buchmann and J. Ding (Eds.): PQCrypto 2008, LNCS 5299, pp. 17–30, 2008.

However, the problem with HFE-based signature schemes is that until now, they were quite slow. In this paper, we study how some simple but very surprising changes to existing ideas can yield a system with much faster signing and key generation at the same security levels as other HFE-based signature schemes. In particular, we set out to make an HFEv$^-$ system with similarly short signatures *and* greater efficiency in the form of fast signing times.

This paper is organized as follows. In Sect. 2, we discuss relevant background on HFE and Quartz systems. In Sect. 3, we introduce the new variant Square-Vinegar, providing a theoretical overview along with explicit constructions and experimental data. In Sect. 4, known attacks are addressed and more experimental results presented. Additional data can be found in the appendix.

2 Hidden Field Equations and Quartz

2.1 The Basic HFE Scheme

Let k be a finite field of size q and K a degree n extension field of k. In the original design, the characteristic of k is 2. K can be seen as an n-dimensional vector space over k and therefore we can identify K and k^n by the usual isomorphism $\varphi : K \to k^n$ and its inverse. HFE makes use of an internal secret map $F : K \to K$ defined by

$$F(X) = \sum_{\substack{0 \le i < j < n \\ q^i + q^j \le D}} a_{ij} X^{q^i + q^j} + \sum_{\substack{0 \le i < n \\ q^i \le D}} b_i X^{q^i} + c \,, \tag{1}$$

where the coefficients a_{ij}, b_i, c are randomly chosen from K and D is a fixed positive integer. A map of this form is often referred to as an HFE map.

By composing F with φ and its inverse we obtain the set of n quadratic multivariate polynomials $\tilde{F} = \varphi \circ F \circ \varphi^{-1} : k^n \to k^n$. Then we hide the structure of this map by means of two invertible affine linear transformations S, $T : k^n \to k^n$. The public key is the set of quadratic multivariate polynomials $(g_1, g_2, \ldots, g_n) = T \circ \tilde{F} \circ S$. The private key consists of the map F and the affine linear transformations S and T.

In such a scheme the most delicate matter is the choice of the total degree D of F. D cannot be too large since decryption (or signing) involves solving the equation $F(X) = Y'$ for a given $Y' \in K$ using the Berlekamp algorithm, a process whose complexity is determined by D. However this total degree cannot be too small either to avoid algebraic attacks, like the one developed by Kipnis and Shamir [15] and the Gröbner Bases (GB) Attack [9].

2.2 HFE Variants

There are several variations of this construction intended to enhance the security of HFE, among which we find the HFE$^-$ [23] and HFEv [20] signature schemes.

HFE$^-$ is the signature scheme obtained from HFE in which we omit r of the polynomials $g_1, g_2, \ldots, g_n$ from the public key. The intent of doing this is to eliminate the possibility of certain attacks, in particular algebraic and Kipnis-Shamir attacks, provided the number r is not too small.

HFEv is a combination of HFE and the Unbalanced Oil & Vinegar scheme [14,21]. The main idea of HFEv is to add a small number v of new variables, referred to as the vinegar variables, to HFE. This makes the system somehow more complicated and changes the structure of the private map. In this case we replace the map F with a more complicated map $G : K \times k^v \to K$.

We can combine HFE$^-$ and HFEv to obtain the so called HFEv$^-$ signature scheme. In this scheme, r polynomials are kept secret and v additional variables are introduced.

Quartz is an HFEv$^-$ signature scheme with a special choice of the parameters, which are $k = \mathbb{F}_2$, $n = 103$, $D = 129$, $r = 3$ and $v = 4$ [24,25]. These parameters of Quartz have been chosen in order to produce very short signatures: only 128 bits. This makes Quartz specially suitable for very specific applications in which short signatures are required, like RFID. Quartz was proposed to NESSIE [18], but it was rejected perhaps due to the fact that its parameters were not chosen conservatively enough. In 2003 Faugère and Joux stated in [9] that the published version of Quartz could be broken using Gröbner bases with slightly fewer than 2^{80} computations.

At present time two modified versions of Quartz are thought to be secure, based on the estimations of [9] on Quartz. The first one, called Quartz-513d, has parameters $k = \mathbb{F}_2$, $n = 103$, $D = 513$, $r = 3$ and $v = 4$. The second version, Quartz-7m, has parameters $k = \mathbb{F}_2$, $n = 103$, $D = 129$, $r = 7$ and $v = 0$. In these versions the high degree D makes the signing process very slow. In fact Quartz-513d was considered impractical for this reason, even as it was proposed.

3 The Square-Vinegar Scheme

We now propose a way to build a fast and highly secure short signature cryptosystem, using the ideas of the HFEv$^-$ signature scheme and the new idea of using finite fields of odd characteristic. With a new choice of parameters we gain computational efficiency without risking the security of the signature scheme. From now on we call these Square-Vinegar schemes. Signatures are still short, which is very convenient to implement in small devices.

3.1 Overview of the New Idea

The set up is basically the same as in the HFEv$^-$ signature scheme. As mentioned above, we replace the map F with the more complicated map $G : K \times k^v \to K$ defined by

$$G(X, X_v) = \sum_{\substack{0 \le i < j < n \\ q^i + q^j \le D}} a_{ij} X^{q^i + q^j} + \sum_{\substack{0 \le i < n \\ q^i \le D}} \beta_i(X_v) X^{q^i} + \gamma(X_v) , \qquad (2)$$

where the coefficients a_{ij} are randomly chosen from K, $\gamma : k^v \to K$ is a randomly chosen quadratic map, $\beta_i : k^v \to K$ are randomly chosen affine linear maps, and

$X_v = (x'_1, \ldots, x'_v)$ represents the new vinegar variables. More precisely the maps β_i and γ are of the form

$$\beta_i(X_v) = \sum_{1 \le j \le v} \xi_{i,j} \cdot x'_j + \nu_i \,,$$

$$\gamma(X_v) = \sum_{1 \le j < l \le v} \eta_{j,l} \cdot x'_j x'_l + \sum_{1 \le j \le v} \sigma_j \cdot x'_j + \tau \,,$$

where $\xi_{i,j}$, ν_i, $\eta_{j,l}$, σ_j and τ are randomly chosen from K. As in HFE, we compose G with φ and its inverse we obtain the set of n quadratic multivariate polynomials. Then we compose with two invertible affine linear transformations $T : k^n \to k^n$ and $S : k^{n+v} \to k^{n+v}$, obtaining the polynomials $(g_1, g_2, \ldots, g_n) = T \circ \varphi^{-1} \circ G \circ \varphi \circ S$. Finally, we remove the last r of these polynomials. The public key is the set of quadratic multivariate polynomials $(g_1, g_2, \ldots, g_{n-r}) : k^{n+v} \to k^{n-r}$. The private key consists of the map G and the affine linear transformations S and T.

While the setup is the same, we make some significant changes. First of all, we will use a field k of odd characteristic. The benefits of working in an odd characteristic are discussed in [5] and will be summarized below in Sect. 4. After making this change, we studied the effect of changing of D and v in order to find the most efficient values. The motivation was that by using the proper number of vinegar variables, we could use a smaller degree D and hence considerably speed up the signing process with the same security level.

With this in mind, we conducted experiments to determine new secure values for D and v. Much to our surprise, in all of our experiments we found that $D = 2$ is sufficiently secure when the field is of odd characteristic, as we will see in Sect. 4. This makes the signature scheme much faster, as we will see in Sect. 3.2.

3.2 The Signing Process

Although HFE is perfectly suitable for encryption and digital signatures, the map F defined by (1) is usually not a surjection. However, in the case of Square-Vinegar schemes, for every different set of vinegar variables we usually obtain a totally different quadratic polynomial in X, which increases the probability of finding a signature for a given document. Actually, in our experiments we were always able to find a signature.

To sign a given document $(\tilde{y}_1, \ldots, \tilde{y}_{n-r}) \in k^{n-r}$, we start by randomly choosing r elements $\tilde{y}_{n-r+1}, \ldots, \tilde{y}_n \in k$ to complete a vector in k^n. Next, we randomly choose values $(w_1, \ldots, w_v) \in k^v$ for the vinegar variables X_v, and then solve for X the equation

$$G\,(X, (w_1, \ldots, w_v)) = \varphi^{-1}(T^{-1}(\tilde{y}_1, \ldots, \tilde{y}_{n-r}, \tilde{y}_{n-r+1}, \ldots, \tilde{y}_n)) \,. \qquad (3)$$

If this equation has no solutions, a new choice of vinegar variables is made yielding a new equation to be solved. We continue in this manner until we find

Table 1. Number of tries to sign a document

q	D	n	v	r	Average number of trials to sign
2	129	103	4	3	1.74
2	2	103	4	3	2.26
13	2	27	3	0	1.85
13	2	28	3	1	1.80
13	2	36	4	3	1.88
31	2	31	4	3	2.09

Table 2. Signing times for some HFEv$^-$ systems

q	D	n	v	r	Number of documents tried	Average time to sign
2	129	103	4	3	100	2.646 s
2	2	103	4	3	100	0.166 s
13	2	27	3	0	100	0.024 s
13	2	28	3	1	100	0.026 s
13	2	36	4	3	100	0.034 s
31	2	31	4	3	100	0.041 s

a choice of vinegar variables whose associated equation in X has a solution. The probability of finding a suitable selection of vinegar variables in a few trials is high. We could confirm this fact with our computer experiments, as evidenced in Table 1 below. We used MAGMA 2.14, the latest version, on a Dell Computer with Windows XP which has an Intel(R) Pentium(R) D CPU 3.00 GHz processor with 2.00 GB of memory installed, to run the computer experiments.

In each case 100 different random documents were signed. We observed that, on average, two tries would be enough to find a solution for that equation. Now suppose that $\tilde{X}$ is a solution of (3), then a signature for the document $(\tilde{y}_1, \ldots, \tilde{y}_{n-r})$ – actually for the whole vector $(\tilde{y}_1, \ldots, \tilde{y}_n)$ – is given by

$$S^{-1}(\varphi(\tilde{X}), w_1, \ldots, w_v) \in k^{n+v} .$$

As mentioned above, with our experiments we found that $D = 2$ suffices as the degree of the secret map G; we will see more about this in Sect. 4. This is undoubtedly a novel and surprising discovery since in the previous versions of HFE and its modifications – all of which are characteristic two – D was always conservatively chosen, usually $D > 128$. These high values of D made the process of signing very slow since solving a univariate equation of such a large degree, even with the fastest algorithms, is not necessarily a fast procedure. On the other hand, when $D = 2$, once the vinegar values have been set, (3) becomes simply a quadratic equation over the field K. Berlekamp's Algorithm can solve a univariate quadratic equation rather quickly, and MAGMA's implementation automatically uses the Berlekamp-Zassenhaus algorithm when appropriate [2]. See Table 2 for signing times for several choices of parameters. Note that signing for the $q = 31$ case shown is 65 times faster than using Quartz parameters.

Another important consequence of the use of $D = 2$ is that generation of the public key for this signature scheme is more efficient. We attribute this to the large number of multiplications that are needed over the field K for $D > 128$. Some results are summarized in Table 3 below.

Table 3. Public key generation times for some HFEv$^-$ systems

q	D	n	v	r	Number of trials	Average time
2	129	103	4	3	100	58.066 s
13	2	27	3	0	100	0.780 s
13	2	28	3	1	100	0.830 s
13	2	36	4	3	100	2.019 s
31	2	31	4	3	100	1.271 s

4 Security Analysis

In this section we will consider known attacks against MPKCs (Gröbner Basis, Kipnis-Shamir, and Vinegar attacks) and discuss their effectiveness against our new scheme. This will lead us to suggest parameter values for a viable Square-Vinegar system.

Before considering the aforementioned attacks in detail, let us mention some minor attacks. First, there do not yet seem to be any attacks against MPKCs utilizing knowledge of plaintext-ciphertext (or document-signature) pairs. Secondly, the recent attack on SFlash [8] does not apply here because that attack used hidden symmetry and invariants of the SFlash public key to overcome the omission of certain polynomials from the public key, but our public key does not have such hidden invariants or symmetry due the presence of the vinegar variables. Also, the attacks used against perturbed systems such as IPHFE, [6,7], do not seem directly applicable, especially considering the differences between even and odd characteristic and internal and external perturbation.

4.1 Gröbner Basis Attack

First let us recall what we mean by a Gröbner Basis Attack. Suppose that someone, who does not know the private key, wants to forge a signature for a given document $(\tilde{y}_1, \ldots, \tilde{y}_{n-r}) \in k^{n-r}$. This attacker has access only to the public key $(g_1, g_2, \ldots, g_{n-r}) : k^{n+v} \to k^{n-r}$. In order to find a valid signature for the given document, the attacker has to solve the system of equations

$$\begin{aligned} g_1(x_1, \ldots, x_n, x'_1, \ldots, x'_v) - \tilde{y}_1 &= 0 \\ g_2(x_1, \ldots, x_n, x'_1, \ldots, x'_v) - \tilde{y}_2 &= 0 \\ &\vdots \\ g_{n-r}(x_1, \ldots, x_n, x'_1, \ldots, x'_v) - \tilde{y}_{n-r} &= 0. \end{aligned} \tag{4}$$

Solving these equations directly, without the use of the internal structure of the system, is known as the algebraic attack. Currently the most efficient algebraic attacks are the Gröbner basis algorithms F_4 [10] and F_5 [11]. Another algorithm called XL has also been widely discussed but F_4 is seen to be more efficient [1], so we focused our energy on studying algebraic attacks via F_4. Among the best implementations of these algorithms is the F_4 function of MAGMA [2], which represents the state of the art in polynomial solving technology.

In [9], algebraic attacks were used to break HFE. The results in that paper seem to indicate that for any q, an HFE system with small D can be broken in such a way. However, this is not the case and their claims only hold up when working over characteristic 2.

Since the system (4) is underdetermined, we expect to find many solutions for it. In order to forge a signature for the given document, it suffices to find only one such solution. So we can guess values for some of the variables yielding a system with the same number of equations but fewer variables, as was done in [3]. This speeds up the attack significantly. Therefore we randomly guessed $v + r$ of the variables and then used the Gröbner basis attack to solve the resulting system of $n - r$ equations with $n - r$ variables, which is faster to solve than (4).

Based on recent observations about MPKCs over odd characteristic [5], we believe that the choices $q = 13$ or $q = 31$ provide a strong defense against an algebraic attack via Gröbner bases. The key point in the case of odd characteristic is that the field equations $x_i^q - x_i$ for $i = 1, 2, \ldots, n + v$, appear to be less useful to an attacker due to their higher degree. In particular, the efficiency of the Gröbner basis attack seems to rely on small characteristic. It is stated in [5] that this stems from the fact that characteristic 2 field equations $x_i^2 - x_i = 0$ help to keep the degrees of the polynomials used in the Gröbner basis algorithm low whereas, for example, $x_i^{13} - x_i = 0$ or $x_i^{31} - x_i = 0$ are much less useful equations in that regard.

Extensive experiments were run to test this idea on the same computer that was used for the signing experiments. For different sets of the parameters (q, D, n, v and r), we generated HFEv$^-$ systems and used F_4 to solve the system of equations in (4) for different random documents.

We sought the lowest value of D for which F_4 took an acceptably long time. By extrapolating the data we could then determine what values of n and r should be used and see if such values were practical. It turns out that $D = 2$ suffices and we did not have to test higher values of D. Notice also that the choice of odd characteristic is important since for even characteristic $X \mapsto X^2$ over K is just a linear map, which cannot be used as a secret internal map.

Further examination of the data showed that with respect to v the attack time hit a plateau at some point, and further increasing v did not appear to increase resistance to the Gröbner basis attack. This behavior can be seen on Fig. 4 in the Appendix section. By extrapolating the data we think that for our choices of D, q, r and n the plateau should occur before $v = 4$, thus we think the choice of $v = 4$ is optimal in this sense.

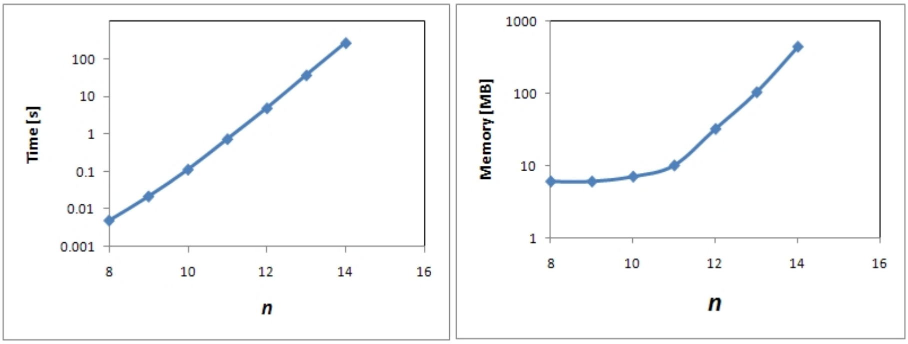

Fig. 1. Running time and required memory under GB Attack for $q = 31$, $v = 4$, $r = 3$ and $D = 2$. No field equations are used in the attack.

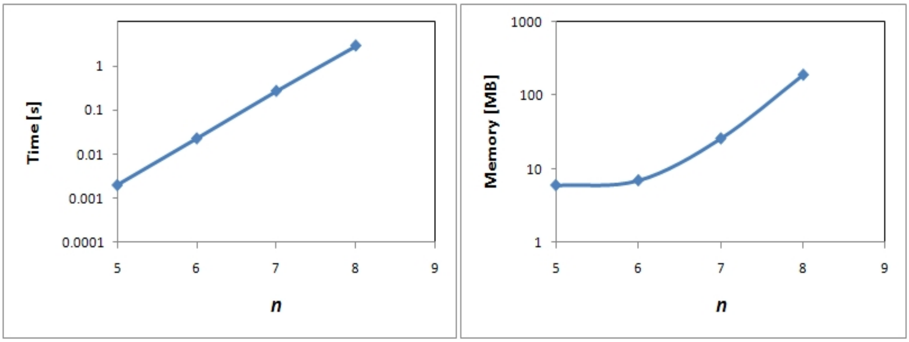

Fig. 2. Running time and required memory under GB attack for $q = 31$, $v = 4$, $r = 3$ and $D = 2$. Including the field equations in the attack.

As mentioned above, for our choice of q – 13 or 31 – the field equations are somehow useless during the Gröbner basis attack. To confirm this, we ran extensive experiments considering this situation, *i.e.*, including and excluding the field equations from the attack. On Figs. 1 and 2 we can see that, in either case, the running time and the required memory under the Gröbner bases attack are exponential in n (similar graphs for $q = 13$ can be seen on Figs. 5 and 6 in the Appendix section).

We can observe that when we include the field equations, the memory used grows much faster than when we do not include them in the attack. This agrees with what we explained above and this is why we say that the field equations are useless for the GB attack. Actually, the field equations not only require more memory but also they slow down the attack for large values of q, for instance $q = 31$. The extrapolations made to suggest parameters in Sect. 4.4 take into account both cases, including and excluding the field equations.

Another important feature that we observed when we excluded the field equations is that, for fixed n, v and r, we did not get any significant change in the

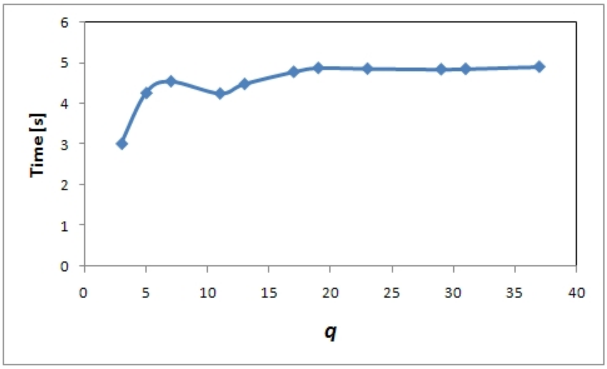

Fig. 3. Running time under GB attack for $n = 9$, $v = 4$, $r = 0$ and $D = 2$, for several values of q. No field equations are used in the attack.

Table 4. Time comparison of some Square-Vinegar systems and random equations under GB attack. $q = 31$, $d = 2$, $v = 4$, and $r = 3$.

n	Our scheme	Random equations
7	0.002 s	0.002 s
8	0.005 s	0.005 s
9	0.022 s	0.022 s
10	0.114 s	0.113 s
11	0.741 s	0.738 s
12	4.921 s	4.755 s
13	37.002 s	37.996 s
14	268.410 s	272.201 s

time required by the GB attack to forge a signature for large values of q, as seen in Fig. 3. This also justifies the choices of $q = 13$ and $q = 31$, since increasing q will not augment the security of the system.

We also constructed random polynomial equations of the same dimensions (same q, n, v and r) and found that the time needed to solve such random equations using Gröbner bases is essentially the same as is needed to break Square-Vinegar with our choices of parameters. Table 4 shows these times for different n.

As observed on the graphs, we could only obtain data for n up to 14, due to memory limitations (any request above 1.2 GB would be immediately rejected by the computer that we used). However, even among the data that we were able to collect, we observed that as n increases, the maximum degree of polynomial used by F4 also increases. Larger scale experiments are being conducted to study systematically how fast this degree increases as n increases; these results will be presented in a future paper.

From the information gathered with our experiments it appears that under our choices of parameters, F4 is no more efficient in solving the public key equations (4) of a Square-Vinegar scheme than a system of random equations.

4.2 Kipnis-Shamir Attack

Kipnis and Shamir developed an attack against HFE [15]. Their original claims were questioned in [13], where it was shown that the Kipnis-Shamir attack was less effective than originally thought and some arguments were made as to why this should be so.

The original attack on HFE was translated to an attack on HFEv in [6]. The resulting attack had a high complexity estimate even though the original, more generous complexity estimates for the HFE attack were used in the computation. Considering [13] and the fact that we are omitting r polynomials from the public key, it seems that a Kipnis-Shamir style attack should not work against Square-Vinegar.

4.3 Vinegar Attack

Since Square-Vinegar utilizes vinegar variables, a priori there is a possibility that it is vulnerable to an attack similar to the one that felled the original Oil-Vinegar scheme.

In the original Oil-Vinegar scheme, the core map $k^n \to k^n$ had a specific shape: each component was a polynomial in which the “oil” variables appeared only linearly, and thus had a quadratic form with a large block of zeros [14,21]. Upon inspection of the attack, we realize that it exploits this property of the quadratic forms [16]. In the Square-Vinegar construction, there are no variables which appear only linearly. The map G ensures that $x_1, \ldots, x_n$ appear quadratically, and the choice of γ ensures that $x'_1, \ldots, x'_v$ appear quadratically.

Once a specific K is fixed (in other words, once a specific irreducible polynomial is chosen to define the extension over k), certain blocks of the quadratic forms of $\varphi \circ G \circ \varphi^{-1}$ are predetermined, but nonzero and not even likely to be sparse. It appears that an attacker would have to find a matrix that simultaneously converts the quadratic forms of all public key polynomials to the prescribed forms. At present time there does not seem to be any method to solve such a problem.

4.4 Parameter Suggestions

Based on the analysis and results obtained throughout Sects. 3 and 4 we are able to suggest new sets of parameters for HFEv$^-$, which we call *Square-Vinegar-31* and *Square-Vinegar-13.* Descriptions are as follows:

Square-Vinegar-31

- $q = 31$, $D = 2$, $n = 31$, $v = 4$ and $r = 3$.
- Size of the public key: 12 Kbytes.
- Length of the signature: 175 bits.

- Time needed to sign a message[1]: 0.041 seconds on average.
- Time to verify a signature[1]: less than 1 ms.
- Best known attack: more than 2^{80} computations.

Square-Vinegar-13

- $q = 13$, $D = 2$, $n = 36$, $v = 4$ and $r = 3$.
- Size of the public key: 14 Kbytes.
- Length of the signature: 160 bits.
- Time needed to sign a message[1]: 0.034 seconds on average.
- Time to verify a signature[1]: less than 1 ms.
- Best known attack: more than 2^{80} computations.

We would also like to propose parameters as toy challenges. The first challenge is $q = 13$, $n = 27$, $v = 3$ and $r = 0$. The second challenge is $q = 13$, $n = 28$, $v = 3$ and $r = 1$. We expect that with these parameter choices, an attack may be practically possible.

5 Conclusion

In this paper we analyzed a new HFEv$^-$ system that seems to have great potential. We showed that with relatively short signatures, Square-Vinegar can be used to sign documents very fast. This was accomplished by working in an odd characteristic and using a low-degree polynomial where previously a very high degree was required. We performed computer experiments to test the security of Square-Vinegar. We used algebraic attacks against smaller-scale systems to determine proper q, D, n, r, and v values for plausible schemes. We also examined other MPKC attacks and gave reasons why Square-Vinegar should be resistant to them.

In the future we would like to have a better understanding of the apparent benefit of odd characteristic. We will also, as mentioned above, study the relationship between n and the polynomials used in GB attacks. In addition, we will further study the effectiveness of attacks similar to those against perturbed systems.

References

1. Ars, G., Faugère, J.-C., Imai, H., Kawazoe, M., Sugita, M.: Comparison Between XL and Gröbner Basis Algorithms. In: Lee, P.J. (ed.) ASIACRYPT 2004. LNCS, vol. 3329, pp. 338–353. Springer, Heidelberg (2004)
2. Computational Algebra Group, University of Sydney. The MAGMA computational algebra system for algebra, number theory and geometry (2005), http://magma.maths.usyd.edu.au/magma/
3. Courtois, N., Daum, M., Felke, P.: On the Security of HFE, HFEv- and Quartz. In: Desmedt, Y.G. (ed.) PKC 2003. LNCS, vol. 2567, pp. 337–350. Springer, Heidelberg (2002)

[1] On an Intel(R) Pentium(R) D CPU 3.00 GHz.

4. Ding, J., Gower, J.E., Schmidt, D.: Multivariate Public Key Cryptosystems. Springer, Heidelberg (2006)
5. Ding, J., Schmidt, D., Werner, F.: Algebraic Attack on HFE Revisited. In: The 11th Information Security Conference, Taipei, Taiwan (September 2008)
6. Ding, J., Schmidt, D.: Cryptanalysis of HFEv and the Internal Perturbation of HFE cryptosystems. In: Vaudenay, S. (ed.) PKC 2005. LNCS, vol. 3386, pp. 288–301. Springer, Heidelberg (2005)
7. Dubois, V., Granboulan, L., Stern, J.: Cryptanalysis of HFE with Internal Perturbation. In: Okamoto, T., Wang, X. (eds.) PKC 2007. LNCS, vol. 4450, pp. 249–265. Springer, Heidelberg (2007)
8. Dubois, V., Fouque, P.-A., Shamir, A., Stern, J.: Practical Cryptanalysis of SFLASH. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 1–12. Springer, Heidelberg (2007)
9. Faugère, J.-C., Joux, A.: Algebraic cryptanalysis of hidden field equation (HFE) cryptosystems using Gröbner bases. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 44–60. Springer, Heidelberg (2003)
10. Faugère, J.-C.: A new efficient algorithm for computing Gröbner bases (F_4). Journal of Pure and Applied Algebra 139, 61–88 (1999)
11. Faugère, J.-C.: A new efficient algorithm for computing Gröbner bases without reduction to zero (F_5). In: International Symposium on Symbolic and Algebraic Computation — ISSAC 2002, pp. 75–83. ACM Press, New York (2002)
12. Gray, M.R., Johnson, D.S.: Computers and Intractability – A guide to the Theory of NP-Completeness. W.H. Freeman and Company, New York (1979)
13. Jiang, X., Ding, J., Hu, L.: Kipnis-Shamir's Attack on HFE Revisited. Cryptology ePrint Archive, Report 2007/203, `http://eprint.iacr.org/`
14. Kipnis, A., Patarin, J., Goubin, L.: Unbalanced oil and vinegar signature schemes. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 206–222. Springer, Heidelberg (1999)
15. Kipnis, A., Shamir, A.: Cryptanalysis of the HFE public key cryptosystem by relinearization. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 19–30. Springer, Heidelberg (1999)
16. Kipnis, A., Shamir, A.: Cryptanalysis of the Oil and Vinegar Signature Scheme. In: Krawczyk, H. (ed.) CRYPTO 1998. LNCS, vol. 1462, pp. 257–267. Springer, Heidelberg (1998)
17. Matsumoto, T., Imai, H.: Public quadratic polynomial-tuples for efficient signature verification and message encryption. In: Günther, C.G. (ed.) EUROCRYPT 1988. LNCS, vol. 330, pp. 419–453. Springer, Heidelberg (1988)
18. NESSIE: New European Schemes for Signatures, Integrity, and Encryption. Information Society Technologies Programme of the European Commission (IST-1999-12324), `http://www.cryptonessie.org/`
19. Patarin, J.: Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt 1988. In: Coppersmith, D. (ed.) CRYPTO 1995. LNCS, vol. 963, pp. 248–261. Springer, Heidelberg (1995)
20. Patarin, J.: Hidden Field Equations (HFE) and Isomorphism of Polynomials (IP): Two new families of asymmetric algorithms. In: Maurer, U. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 33–48. Springer, Heidelberg (1996); extended Version, `http://www.minrank.org/hfe.pdf`
21. Patarin, J.: The Oil and Vinegar Signature Scheme. In: Dagstuhl Workshop on Cryptography (September 1997)

22. Patarin, J., Goubin, L.: Trapdoor one-way permutations and multivariate polynomials. In: Han, Y., Quing, S. (eds.) ICICS 1997. LNCS, vol. 1334, pp. 356–368. Springer, Heidelberg (1997); extended Version, `http://citeseer.nj.nec.com/patarin97trapdoor.html`
23. Patarin, J., Goubin, L., Courtois, N.: C^*_{-+} and HM: variations around two schemes of T. Matsumoto and H. Imai. In: Ohta, K., Pei, D. (eds.) ASIACRYPT 1998. LNCS, vol. 1514, pp. 35–50. Springer, Heidelberg (1998)
24. Patarin, J., Goubin, L., Courtois, N.: Quartz, 128-bit long digital signatures. In: Naccache, D. (ed.) CT-RSA 2001. LNCS, vol. 2020, pp. 352–357. Springer, Heidelberg (2001)
25. Patarin, J., Goubin, L., Courtois, N.: Quartz, 128-bit long digital signatures. An updated version of Quartz specification, pp. 357-359, `http://www.cryptosystem.net/quartz/`

Appendix: Some Additional Graphs

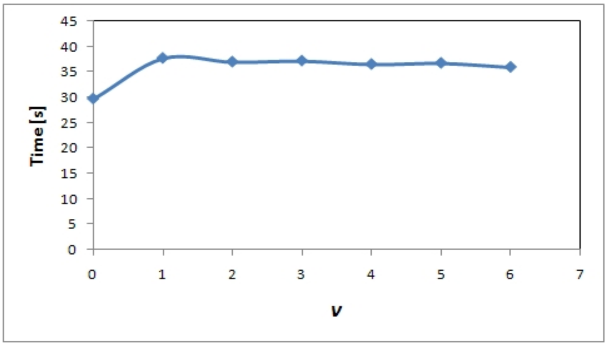

Fig. 4. Running time under GB attack for $n = 13$, $r = 3$ and $D = 2$, for several values of v. No field equations are used in the attack.

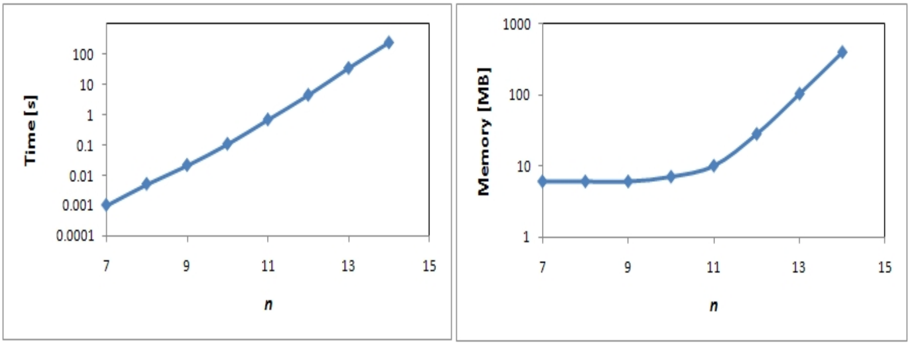

Fig. 5. Running time and required memory under GB attack for $q = 13$, $v = 4$, $r = 3$ and $D = 2$. No field equations are used in the attack.

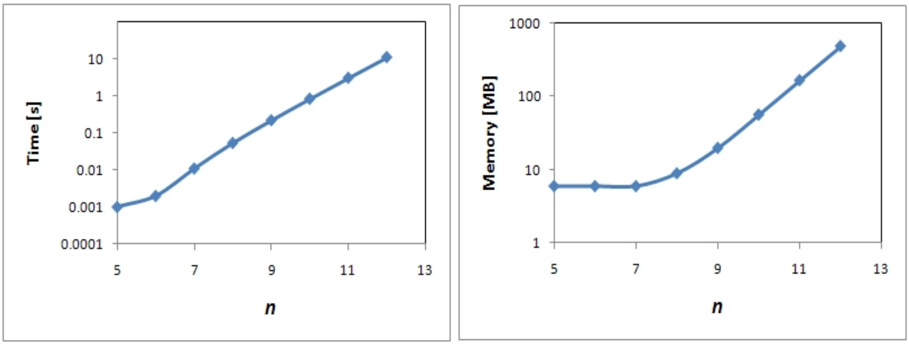

Fig. 6. Running time and required remory under GB attack for $q = 13$, $v = 4$, $r = 3$ and $D = 2$. Including the field equations in the attack.