$\mathcal{MQ}^*$-IP: An Identity-based Identification Scheme without Number-theoretic Assumptions[1]

Christopher Wolf[⋆] and Bart Preneel[†]

[⋆]Christopher.Wolf@ruhr-uni-bochum.de or chris@Christopher-Wolf.de,
Horst Görtz Institut for IT-Security, R.U.Bochum, Universitätsstr. 150,
44780 Bochum, Germany, http://www.hgi.rub.de/

[†]Bart.Preneel@esat.kuleuven.ac.be,
ESAT-COSIC, K.U.Leuven, Kasteelpark Arenberg 10,
B-3001 Leuven-Heverlee, Belgium, http://www.esat.kuleuven.ac.be/cosic/

At the time of writing, the first author was member of ESAT-COSIC.

Date: 2010-02-18

Abstract

In this article, we propose an identification scheme which is based on the two combinatorial problems $\mathcal{M}$ultivariate $\mathcal{Q}$uadratic equations ($\mathcal{MQ}$) and Isomorphism of Polynomials (IP). We show that this scheme is statistical zero-knowledge. Using a trapdoor for the $\mathcal{MQ}$-problem, it is possible to make it also identity-based, i.e., there is no need for distributing public keys or for certificates within this scheme.

The size of the public keys and the communication complexity are within the range of other non-number-theoretic identification schemes. In contrast to $\mathcal{MQ}^*$-IP, these schemes do usually no permit identity-based public keys.

1 Introduction

In an identification scheme, there are two conflicting goals: On the one hand, a prover wants to convince a verifier of its identity. On the other hand, he does not want to give the verifier the ability to impersonate as himself. This means especially that the prover cannot reuse the same information several times. If this were allowed, the verifier V could record this information, e.g., a password, and reply it against another verifier V'. One solution for this problem are identification schemes which offer the "zero-knowledge" property. This idea has been introduced by Goldwasser, Micali and Rackoff [GMR85]. In a nutshell, a zero-knowledge protocol does not reveal more information than the fact that a prover knows a specific information. In particular, the information itself is not revealed.

In addition, Shamir introduced the idea of "identity based" cryptosystems [Sha84]. The overall idea is to have public key systems without the need of certified public keys. Instead, each user knows a common parameter with some hidden internal structure. This hidden trapdoor makes it possible for a central authority CA to compute private keys for

[1] This article is based on the talk *C. Wolf: Zero-Knowledge and Multivariate Quadratic Equations, BRIC Workshop on Coding and Cryptography, Cork, Ireland, May 20, 2004.*

all users. In this setting, the public key is based on some publicly available information, e.g., the eMail-address of each user.

As we will see below, it is possible to combine both ideas, i.e., to obtain a public key identification scheme where the public keys are based on some publicly available information. One possible application is access control to a building: all users are issued smart-cards. The unique number of each card serves as the public identity of the user. Each access-point only stores the common system parameter. The private key is computed by the department which provides the users with cards. This way, there is no need for certificates as the validity of the public key is guaranteed by the construction of the scheme.

1.1 Related Work

There has been much work in the area of identification schemes. A special stress has been laid on schemes which use number-theoretic assumption, e.g., factorisation or elliptic curves. See [PBO$^+$03, Sec. 8] for a state-of-the-art overview. In particular, the security requirements for such schemes are identified as 2^{80} for the security of the whole scheme and 2^{-32} for the impersonation probability for one run of the protocol. We will use both bounds in this paper.

The focus in this paper are schemes which use others than number-theoretic assumptions. Possible (and also practical) schemes from this class are the "Permuted Kernel Problem" (PKP) from Shamir [Sha89], "Constrained Linear Equations" (CLE) [Ste94] and "Syndrome Decoding" [Ste93, Ste96], both from Stern, and the "Permuted Perceptron Problem" (PPP) by Pointcheval [Poi95, PP03]. They are all based on $\mathcal{NP}$-complete problems.

1.2 Achievement

In this article, we propose an identification scheme which combines the two combinatorial problems $\mathcal{M}$ultivariate $\mathcal{Q}$uadratic equations ($\mathcal{MQ}$) and **I**somorphism of **P**olynomials (IP). The hardness of both problems is well-established (cf Sec. 3). In fact, the $\mathcal{MQ}$-problem is $\mathcal{NP}$-complete [GJ79, p. 251]. The problem of finding the isomorphism of polynomials equations (IP) has been shown to be as hard as the graph isomorphism problem [PGC98b].

To the knowledge of the authors, there is only one other identification scheme, not based on number-theoretic assumptions, which can derive the public keys from the users' identities, namely the syndrome decoding scheme (SD) from Stern.

The scheme proposed in this paper is practical, i.e., both its computation and its communication complexity is comparable with other schemes known so far (see Section 5.3). In addition, we prove its security by showing that it is zero-knowledge (Section 2.2).

1.3 Outline of this Paper

The remainder of the paper is organised as follows: after clarifying the notation, we will move on to a description of the protocol, both in its original version and in its identity-based form. For both, we show zero-knowledge and its security related to the $\mathcal{MQ}$- and the IP-problem. Section 3 contains a cryptanalysis of the schemes proposed and gives lower bounds of security. The next section concentrates on optimisations of the protocol presented in Section 2. In Section 5, we use the results of the previous sections to obtain instances of the protocol and compare them with the protocols described in Section 1.1. The paper concludes with Section 6.

1.4 Notation

Within the next sections, we will use the following notation: let $\mathbb{F}$ be a finite field with $q := |\mathbb{F}|$ elements. Let $\mathbb{E}$ be an extension field over the ground field $\mathbb{F}$. This extension field $\mathbb{E}$ is generated by the irreducible polynomial $i(t)$ of degree $n := \partial i(i)$. The degree of the extension field $\mathbb{E}$ over the ground field $\mathbb{F}$ is also n. Moreover, let $S \in \mathrm{AGL}_n(\mathbb{F})$ be an affine transformation. The transformation S can be written as $S(x) = M_S x + v_s$ for an invertible matrix $M_S \in \mathbb{F}^{n \times n}$ and a vector $v_s \in \mathbb{F}^n$. Finally, let $\mathcal{A}$ be a system of m polynomial-equations in n variables, i.e., $\mathcal{A}$ consists of m polynomials of the form

$$a_i(x_1, \ldots, x_n) = \gamma_{i,j,k} x_j x_k + \beta_{i,j} x_j + \alpha_i$$

for $1 \le i \le m, 1 \le j, k \le n$ and $\alpha_i, \beta_{i,j}, \gamma_{i,j,k} \in \mathbb{F}$. Moreover, we denote the set of all quadratic polynomials in n variables by $\mathcal{Q}(\mathbb{F}[x_{1,\ldots,n}])$, i.e.,

$$\mathcal{Q}(\mathbb{F}[x_{1,\ldots,n}]) \quad := \quad \{\ a_i(x_1, \ldots, x_n)\}$$

where the polynomials a_i are defined as above with all possible values for $\alpha_i, \beta_{i,j}, \gamma_{i,j,k} \in \mathbb{F}$ (constant, linear, and quadratic terms, respectively).

2 The Basic Schemes

2.1 The original IP scheme

In this section, we give a concise overview of the original isomorphism of polynomials (IP) scheme of Patarin [Pat96, Sec. 18]. It uses the difficulty of finding a transformation $S \in \mathrm{AGL}_n(\mathbb{F})$ such that two given vectors of polynomials $\mathcal{A}, \mathcal{B} \in [\mathcal{Q}(\mathbb{F}[x_{1,\ldots,n}])]^m$ satisfy the equation $\mathcal{B} = \mathcal{A} \circ S$.

The private key k of the prover P consists of the affine transformation S, i.e., $k := S$. The public key K consists of two vectors of polynomials $\mathcal{A}$ and $\mathcal{B}$, i.e., $K = (\mathcal{A}, \mathcal{B})$.

The corresponding protocol works as follows: in a precomputation step, the prover P chooses a transformation $S' \in_R \mathrm{AGL}_n(\mathbb{F})$ and computes the vector $\mathcal{C} := \mathcal{A} \circ S'$. After this, there are three interactions between the prover and the verifier:

1. The prover P sends the verifier V the vector $\mathcal{C}$.

2. The verifier V tosses a coin $c \in_R \{0, 1\}$ and sends the result to the prover P.

3. The prover P sends the transformation $\tilde{S} \in \mathrm{AGL}_n(\mathbb{F})$.

   b = 0: the prover sends $\tilde{S} := S'$.

   b = 1: the prover sends $\tilde{S} := S^{-1} \circ S'$.

First, the verifier tests if the transformation $\tilde{S}$ is affine. Then, depending on c, the verifier V checks whether one of the following equations is satisfied:

b = 0: $\mathcal{C} \overset{?}{=} \mathcal{A} \circ \tilde{S}$.

b = 1: $\mathcal{C} \overset{?}{=} \mathcal{B} \circ \tilde{S}$.

The verifier V accepts if this check is successful. A dishonest prover P' can trick a verifier V with probability $\frac{1}{2}$. See [Pat96, PGC98b] for this fact and a more detailed description of the original IP scheme. In both papers, there is also a version called "IP with two secrets". See Section 3.1 for a comparison of these two schemes.

The IP scheme has three major drawbacks. First, the size of the public key is rather large as it consists of vectors of quadratic polynomials. Second, there is no way known for this scheme to use identity-based public keys K. Finally, the communication complexity is quite bad, too, as we need to transfer vectors consisting of quadratic equations plus affine transformations. The first two issues will be addressed in sections 2.2 and 2.3, the last in Section 4.2.

2.2 $\mathcal{MQ}$-IP: Extending IP with $\mathcal{MQ}$

In the previous section, we used the IP-problem to obtain an identification scheme. In this section, we combine it with the problem of finding a solution for a multivariate quadratic system of equations, i.e., the $\mathcal{MQ}$-problem. In fact, the $\mathcal{MQ}$-problem is $\mathcal{NP}$-complete (cf [GJ79, p. 251] and [PG97, Appendix] for a detailed proof). In addition, there is strong empirical and theoretical evidence [Pat96, CKPS00, FJ03], that it is also hard on average — even with an embedded trapdoor — and hence can be used as basis for a secure public key crypto system.

Using the definition of $\mathcal{Q}(\mathbb{F}[x_{1,\ldots,n}])$ from Section 1.4 we can define a whole system of polynomial equations as

$$\mathcal{QE}_m(\mathbb{F}[x_{1,\ldots,n}]) \quad := \quad \left\{ \begin{array}{rcl} a_1(x_1, \ldots, x_n) & = & 0, \\ & & \vdots \\ a_m(x_1, \ldots, x_n) & = & 0 \end{array} \right.$$

with quadratic polynomials $a_i \in \mathcal{Q}(\mathbb{F}[x_{1,\ldots,n}])$. For given $\mathcal{A} \in \mathcal{QE}_m(\mathbb{F}[x_{1,\ldots,n}])$, we are interested in a vector $X \in \mathbb{F}^n$ such that the equation $a_i(X) = 0$ holds for all polynomials $a_i \in \mathcal{A}$. In fact, for randomly chosen polynomials a_i, this is exactly an instance of the $\mathcal{MQ}$-problem with m equations in n variables.

Under the assumption that $\mathcal{P} \neq \mathcal{NP}$, this problem is one-way: for a given vector $X \in \mathbb{F}^n$, we can construct m polynomials a_i (see below) satisfying the corresponding equations. However, the reverse process, i.e., to obtain the vector X for a given system of equations $\mathcal{A}$ is difficult. To construct a system $\mathcal{A}$ for given for given X, the prover picks a vector $X \in_R \mathbb{F}^n$ and a system of equations $\mathcal{A} \in_R \mathcal{QE}_m(\mathbb{F}[x_{1,\ldots,n}])$. After evaluating the quadratic and linear terms (i.e., $\gamma_{i,j,k}x_jx_k$ and $\beta_{i,j}x_j$ resp.), he chooses the constant terms α_i such that $a_i(X) = 0$ is satisfied. As a randomly chosen system of equations $\mathcal{A}$ behaves like a random function, and the vector X has been chosen at random as well, the distribution of the constants α_i is the same as if they had been chosen at random in the first place.

After this observation, we move on to the corresponding protocol. Let the private key k of the prover P be the vector $X \in \mathbb{F}^n$, i.e., $k := X$. The public key K consists of three components: two systems of equations $\mathcal{A}, \ddot{\mathcal{A}} \in \mathcal{QE}_m(\mathbb{F}[x_{1,\ldots,n}])$ such that $\mathcal{A}(X) = 0$, $\ddot{\mathcal{A}} = \mathcal{A} \circ \ddot{S}$ for some $\ddot{S} \in \mathrm{AGL}_n(\mathbb{F})$ and a solution of the second system $\ddot{X} \in \mathbb{F}^m$, i.e., $\ddot{\mathcal{A}}(\ddot{X}) = 0$, and hence, $K = (\mathcal{A}, \ddot{\mathcal{A}}, \ddot{X})$.

In a precomputation step, the prover P selects at random a transformation $S \in_R \mathrm{AGL}_n(\mathbb{F})$. He computes $\mathcal{B} := \mathcal{A} \circ S$. After this, the interactive scheme works as follows:

1. The prover P sends the verifier V the system $\mathcal{B}$.
2. The verifier V tosses a coin $c \in_R \{0, 1\}$ and sends the result to the prover P.
3. Depending on the coin c, the prover P sends

   b = 0: the transformations S.

   b = 1: a solution $X' := S^{-1}(X)$ of the system $\mathcal{B}$.

Depending on c, the verifier V checks if one of the following equations is satisfied:

b = 0: $\mathcal{B} \overset{?}{=} \mathcal{A} \circ S$. Moreover, the verifier tests $S \overset{?}{\in} \mathrm{AGL}_n(\mathbb{F})$, i.e., if S is an affine transformation.

b = 1: $\mathcal{B}(X') \overset{?}{=} 0$.

As in the original IP-scheme, the success-probability for a dishonest prover is $\frac{1}{2}$.

We now show that the following scheme is in fact “complete”, “sound” and “zero-knowledge” (cf [MvOV96, Sec. 10.4] and [Gol02]).

2.2.1 Completeness

Here we prove that the protocol above is in fact complete, i.e., given an honest prover P and an honest verifier V, the prover is able to persuade the verifier that he knows the secret $X \in \mathbb{F}^m$. In case $c = 0$, the prover sends the transformation S and the verifier checks if $\mathcal{B} \overset{?}{=} \mathcal{A} \circ S$. This equation is true due to the construction of the system of equations $\mathcal{B}$. In the case $c = 1$, the prover sends X' and the verifier checks if

$$\mathcal{B}(X') = (\mathcal{A} \circ S) \circ (S^{-1}(X)) = \mathcal{A}(X) = 0$$

Hence, the above protocol is complete.

2.2.2 Soundness

Next we show that the protocol is sound, i.e., the existence of an adversary which can succeed with probability $> (\frac{1}{2})^r$ implies the existence of a knowledge-extractor. Here, the constant r denotes the number of rounds the above protocol is repeated. The following proof is inspired by [Ste94].

Assume that there exists a probabilistic polynomial-time adversary which is accepted with probability $\geq (\frac{1}{2})^r + \epsilon$ by an honest verifier V. This implies that there exists a probabilistic polynomial-time machine $\mathcal{M}$ which extracts with overwhelming probability the secret X from this adversary.

To prove this statement, we consider the tree $Y(\omega)$ of all 2^r possible questions of the verifier V. The adversary's random tape ω is fixed. Denote

$$\delta \quad := \quad Pr(Y(\omega) \text{ has a node with two children})$$

Consider the case $\delta < \epsilon$. Then, the probability of the adversary's success is lower than $(\frac{1}{2})^r + \epsilon$: the term $(\frac{1}{2})^r$ comes from the case where $Y(\omega)$ has no node with two children and ϵ from the other cases. Thus $\delta \geq \epsilon$. Resetting the adversary a total of $1/\delta$ times, we find an execution tree with a vertex having two children with probability $1 - (1 - \delta)^{1/\delta}$, i.e., basically 1 if the process is repeated a few times with different random tapes. In this context, a node with two children means that we know for a pair $(\mathcal{A}, \mathcal{B}) \in \mathcal{QE}_m(\mathbb{F}[x_{1,\ldots,n}])$ both the transformation $S \in \mathrm{AGL}_n(\mathbb{F})$ and a solution $X' \in \mathbb{F}^n$ such that $\mathcal{B}(X') = 0$ and $\mathcal{B} = \mathcal{A} \circ S$. Hence we can compute $\mathcal{A} \circ (S(X')) = (\mathcal{B} \circ S^{-1} \circ S)(X') = \mathcal{B}(X') = 0$, i.e., we found a solution $\tilde{X} := S(X')$ for the $\mathcal{MQ}$-problem $\mathcal{A}(X) = 0$.

Therefore, using $\mathcal{M}$, we extracted the secret X from the adversary. But the existence of such a machine $\mathcal{M}$ violates our intractability assumption about the $\mathcal{MQ}$-problem. Thus, the above protocol is sound.

2.2.3 Zero-Knowledge

Finally, we show that the $\mathcal{MQ}$-IP protocol is zero-knowledge, i.e., the prover does not release information during the run of the protocol. We prove statistical zero-knowledge [Gol02, Sec. 3.3.3], i.e., the output of a simulator is not distinguishable from the output of a prover, even if the observer has unlimited computational power.

To construct this simulator, we need to produce different kind of objects. First, systems of equations $\mathcal{B}$ affine to $\mathcal{A}$ and second systems of equations $\mathcal{B}'$ which have a solution X' (usually, $X \neq X'$). The systems $\mathcal{B}$ can be produced by the simulator statistical zero-knowledge: he picks at random an affine transformation $S \in_R \mathrm{AGL}_n(\mathbb{F})$ and derives a system $\mathcal{B}$ by computing $\mathcal{B} = \mathcal{A} \circ S$. This is exactly the same as for the prover, i.e., a sequence of systems of equations $(\mathcal{B}_1, \ldots, \mathcal{B}_\lambda)$ for some $\lambda \in \mathbb{N}$ produced by the prover has the same statistical properties as both the simulator and the prover choose the corresponding affine transformations S in the same way.

The same is true if the simulator want to produce an equation $\mathcal{B}'(X') = 0$, i.e., with a known solution. In this case, the simulator picks at random an affine transformation $S' \in_R \mathrm{AGL}_n(\mathbb{F})$ and computes $\mathcal{B}' = \ddot{\mathcal{A}} \circ S'$. Affine transformations form a group, therefore, they are closed under composition. As $\ddot{\mathcal{A}} = \mathcal{A} \circ \ddot{S}$ for some affine transformation $\ddot{S}$, the output distribution of the simulator is the same as for the prover. In addition, we have $\ddot{\mathcal{A}}(\ddot{X}) = 0$, i.e., $\mathcal{B}'(X') = 0$ for $X' = \ddot{X} \circ S'^{-1}$, hence, the simulator knows a solution X' of the system $\mathcal{B}'$.

After these preliminaries, we can construct the simulator, i.e., a probabilistic polynomial Turing machine which builds communication tapes with a distribution indistinguishable from the real ones.

The simulator chooses an affine transformation $S \in_R \mathrm{AGL}_n(\mathbb{F})$ at random. Independently, it flips a coin $c \in_R \{0, 1\}$. Depending on c it computes

b=0: $\mathcal{B} = \mathcal{A} \circ S$. It outputs $(\mathcal{B}, 0, S)$ on the communication tape.

b=1: $\mathcal{B} = \ddot{\mathcal{A}} \circ S$ and $X' = S^{-1}(\ddot{X})$. It outputs $(\mathcal{B}, 1, X')$ on the communication tape.

These communication tapes follow the same statistical distribution as the tapes of the communication between the prover and the verifier (see above). Therefore, the $\mathcal{MQ}$-IP protocol is statistical zero knowledge.

2.3 Identity Based Schemes $\mathcal{MQ}^*$-IP

To make our identification scheme identity based, we need another observation: to obtain a vector $\alpha \in \mathbb{F}^m$, the user chooses the quadratic and linear terms $\gamma_{i,j,k}, \beta_{i,j} \in \mathbb{F}$, a vector $X \in \mathbb{F}^n$ and compute the corresponding α (cf Sec. 2.2). However, it is also possible to keep the $\gamma_{i,j,k}, \beta_{i,j}$ fixed for all users and pick only $X \in \mathbb{F}^n$ at random to obtain a vector $\alpha \in \mathbb{F}^m$ as a public key for each user. The security of the scheme does not change as we see with the following reduction: given an $\mathcal{MQ}$-problem $\mathcal{A}(X) = 0$ for unknown X, we pick a vector $X' \in \mathbb{F}^n$ at random and compute the corresponding $\alpha' \in \mathbb{F}^m$ such that $\mathcal{A}'(X') = 0$. Now assume that this knowledge gives an advantage in finding the original $X \in \mathbb{F}^n$. If such an algorithm existed, it would solve all $\mathcal{MQ}$-problems — which is clearly a violation of the intractability assumption of the $\mathcal{MQ}$-problem. Therefore, it is save for us to keep the quadratic and linear terms $\gamma_{i,j,k}, \beta_{i,j}$ fixed and change only the vector $X \in \mathbb{F}^n$ — and hence the vector $\alpha \in \mathbb{F}^m$.

As the system of equations $\mathcal{A}$ is now a system parameter, it is possible to embed a trapdoor into it. A central authority CA knowing this trapdoor is able to compute the secrets $X \in \mathbb{F}^n$ for given vectors $\alpha \in \mathbb{F}^m$, i.e., especially for such α that depend on the identity of the prover P. This way, we obtain an identity-based identification-scheme. The rest of the scheme is as described in Section 2.2. Moreover, the security prove for $\mathcal{MQ}^*$-IP is the same as for $\mathcal{MQ}$-IP. We only have to replace our intractability assumption for $\mathcal{MQ}$ by the intractability assumption of the corresponding trapdoor.

One possible trapdoor is the C^{*--}-system [PGC98a]. We call the corresponding scheme C-IP. Its security is investigated in Section 3.4. Concrete schemes built on this trapdoors can be found in Section 5.

Generally speaking, any $\mathcal{MQ}$-trapdoor which permits a signature scheme can be used for the $\mathcal{MQ}^*$-IP identification scheme. However, due to space limitations in this paper, we will only investigate C-IP here.

3 Cryptanalysis

For the cryptanalysis of our scheme, we can dwell on various work done for other purposes, especially about the difficulty of the IP-problem [PGC98b, GMS02] and the difficulty of the $\mathcal{MQ}$-problem with a trapdoor [Pat95, FJ03].

In this section, we will first look on the previously known results on the difficulty of the IP-problem. In particular, we will compare the complexity of algorithms for the IP-problem with two secrets (IP-2), and with one secret (IP-1). Secondly, we will study a birthday attack against $\mathcal{MQ}^*$-IP and some security implications of provers cooperating to cheat. Finally, we will identify which parameters for the $\mathcal{MQ}$-problem with a trapdoor, i.e., for the C^{*--}-problem, meet the security requirements stated in Section 1.1.

3.1 Comparison between IP-1 and IP-2

In Section 2.1, we have introduced the problem "Isomorphism of Polynomials with One Secret" (IP-1) where we have two systems of equations $\mathcal{A}, \mathcal{B} \in \mathcal{QE}_m(\mathbb{F}[x_{1,\ldots,n}])$ and $\mathcal{A} = \mathcal{B} \circ S$ for some affine transformation $S \in \mathrm{AGL}_n(\mathbb{F})$. In contrast, the "Isomorphism of Polynomials with Two Secrets" (IP-2) has two affine transformations $S \in \mathrm{AGL}_n(\mathbb{F})$ and $T \in \mathrm{AGL}_m(\mathbb{F})$. Here, the transformation S is in effect a change in the variables (as for IP-1) while the transformation T also mixes the resulting equations in an affine way. So the relation between the two systems of equations becomes $\mathcal{A} = T \circ \mathcal{B} \circ S$. As in IP-1, the public key is $K := (\mathcal{A}, \mathcal{B})$. However, the private key is different as it is now $k := (S, T)$. An identification scheme based on IP-2 has to be adapted accordingly. See [Pat96, PGC98b] for a more detailed description of IP-1 and IP-2.

In [PGC98b], Patarin, Goubin, and Courtois present various algorithms for the cryptanalysis of the IP-problem with one or two secrets. For IP-2 and S, T not affine but linear, the most powerful attack described in [PGC98b] is the so-called "combined power attack". It exploits the birthday paradox. Using $n^{\mathcal{O}(1)}\mathcal{O}(q^{n/2})$ memory and $n^{\mathcal{O}(1)}\mathcal{O}(q^{n/2})$ computations, the attack is able to solve IP-2. If we have $S \in \mathrm{AGL}_n(\mathbb{F})$ and $T \in \mathrm{AGL}_m(\mathbb{F})$ being affine transformations, we have to find the corresponding vectors $v_s \in \mathbb{F}^n$ and $v_t \in \mathbb{F}^m$ first. This costs $\mathcal{O}(q^{m+n})$. Setting $m = n$, the overall attack will require $n^{\mathcal{O}}(1)\mathcal{O}(q^{2.5n})$ computations in total.

For IP-1, i.e., "Isomorphism of Polynomials with One Secret", the most powerful attacks known so far have been presented by Geiselmann, Meier, and Steinwandt [GMS02]. The complexity of the normal attack is $\mathcal{O}(q^{2n})$. They also present some heuristic improvements — which can cut down the total computation time of the system. Although ground fields with $q > 2$ are to be more vulnerable than $q = 2$, these heuristic improvements seem to contribute only a constant factor and are hence with only limited impact compared with q^{2n}. We therefore identify the workload of the attack as $\mathcal{O}(q^{2n})$.

To determine if we should use IP-1 or IP-2 for our scheme, we need to be careful about the question if the transformations S, T are affine or linear. In the first case, IP-2 is more secure, in the latter, IP-1. However, at least the transformation T must be linear as otherwise a dishonest prover P' would be able to cheat using the following strategy: denote $\mathcal{A} \in \mathcal{QE}_m(\mathbb{F}[x_{1,\ldots,n}])$ the public key of the honest prover P. First, his dishonest counterpart P' picks a vector $X' \in_R \mathbb{F}^n$ at random, an affine transformation $S \in_R \mathrm{AGL}_n(\mathbb{F})$ and the linear part of the transformation T, i.e., an invertible matrix $M_T \in_R \mathbb{F}^{m \times m}$. He then computes the vector $v_t = -(M_T \circ \mathcal{A} \circ S)(X')$. With the corresponding triple (X', S, T), the cheater P' can successfully fool a verifier V. However, having T linear instead of affine, this attack is no longer possible. On the other hand, the "combined power attack" has now a workload of $\mathcal{O}(q^{1.5n})$ and is therefore faster than the attack of Geiselmann, Meier, and Steinwandt.

As the problem "Isomorphism with Two Secrets" (IP-2) is more vulnerable against the attacks known so far than "Isomorphism with One Secret" (IP-1), we concentrated on the latter while developing our scheme. However, if the need arises, it is straightforward to change it from IP-1 to IP-2.

3.2 Birthday attack against an ID-based scheme

Any ID-based identification scheme will suffer from the following attack using the birthday paradox: the attacker compiles a list of $\sqrt{q^m}$ possible public keys, i.e., vectors from $\mathbb{F}^m$. After that, he evaluates the common vector of polynomials $[\mathcal{Q}(\mathbb{F}[x_{1,\ldots,n}])]^m$ by a solution-vector $X \in \mathbb{F}^n$ a total of $\sqrt{q^m}$ times. By the birthday paradox, he will obtain approx. one solution X_i which matches one identity id_j. The memory-requirements of this attack are $\mathcal{O}(q^{m/2})$ and the time-requirements are $\mathcal{O}(q^{m/2})$, too.

To withstand this attack, we need $q^{m/2}$ to be larger than the security bound of breaking the whole scheme, i.e., $q^{m/2} > 2^{80}$.

3.3 Group of Cheating Users

As for the identity-based SD scheme of Stern [Ste96, Sec. 4.3], it is dangerous for $\mathcal{MQ}^*$-IP if many users pool their secret keys. Using $(n+1)$ pairs $(X, \ddot{X}) \in \mathbb{F}^n$ and $\ddot{X} = \ddot{S}(X)$, they are able to compute the affine transformation $\ddot{S}$. Using the public value $\ddot{X}_P$ of the prover P, they are now able to recover his secret key X_P.

There are several ways to cope with this problem. First, we may assume that the users cannot access their private keys k, i.e., the k are stored in a temper-resistant device. Second, we can pick different $\ddot{S} \in_R \mathrm{AGL}_n(\mathbb{F})$ for different users and therefore obtain different $\ddot{\mathcal{A}} \in \mathcal{QE}_m(\mathbb{F}[x_{1,\ldots,n}])$. As all transformations $\ddot{S}$ are now different, the above attack is no longer possible. But this solution contradicts the goal of Section 4.1 to have small public keys. Finally, we can remove the information $\ddot{X} \in \mathbb{F}^n$ from the public key of each user. The price to pay: the corresponding scheme is no longer statistical zero-knowledge.

3.4 Parameters for the C^{*--}-Trapdoor

This $\mathcal{MQ}$-trapdoor has been used in the two signature schemes Flash [CGP00] and SFlash [CGP02], both submitted to [NES]. Therefore, it has been studied extensively, cf [PBO$^+$03, Sec. 7.4.3]. Here, we give a short overview of C^{*--}, see [PGC98a] for a detailed description.

The C^{*--}-scheme uses a finite field $\mathbb{F}$ of characteristic 2 and an extension field $\mathbb{E}$ of dimension n, for security reasons a prime. $\mathbb{E}$ is also identified with the vector-space $\mathbb{F}^n$. In addition, we define a polynomial $P(x) := x^\lambda$ over $\mathbb{E}$ where $\lambda := q^a + 1$ such that $\gcd(q^a + 1, q^n - 1) = 1$ and $a \in \mathbb{N}$. This way, the polynomial $P(x)$ is a permutation. In addition, we have two linear transformations $s, t \in \mathrm{GL}_n(\mathbb{F})$. The corresponding $\mathcal{MQ}$-problem is constructed as $\mathcal{A} = t \circ P \circ s$ where r equations of the system of equations $\mathcal{A}$ are removed. As there is an attack in $\mathcal{O}(q^r)$ [PGC98a], the number r has to be chosen such that $q^r \geq 2^{80}$.

In the original scheme, the two transformations s, t have been affine, not linear. However, [GSB01] describes an attack which is able to recover the constant parts of s, t. Therefore, it is advisable to have these two transformations linear.

Although the C^{*--}-scheme is a sub-class of the more general HFE-scheme, the recent attack of Faugère and Joux against HFE [FJ03] does not apply against C^{*--}-schemes: the complexity of this attack grows with the degree λ of the private polynomial. As this degree λ is far higher in C^{*--} than in HFE (here, $\lambda = 129$ or 257), their attack is no longer effective.

4 Improvements

In this section, we investigate some optimisations of the $\mathcal{MQ}$-IP protocol. In particular, we want to decrease the size of the public key and also the communication complexity.

4.1 Smaller Public Keys

As we saw in Section 2.2, it is enough to publish a vector $\alpha \in \mathbb{F}^m$ as a public key for a given user and to keep the quadratic and linear terms $\gamma_{i,j,k}, \beta_{i,j} \in \mathbb{F}$ fixed as a system parameter. We will now show that this is also true for the second system of equations $\ddot{\mathcal{A}} \in \mathcal{QE}_m(\mathbb{F}[x_{1,\ldots,n}])$: let $\ddot{S} \in \mathrm{AGL}_n(\mathbb{F})$ be an affine transformation and denote by $s_{j,k} \in \mathbb{F}$ for $0 \leq k < j \leq n$ its coefficients. In this notation, the elements $s_{1,0}, \ldots, s_{n,0}$ denote the coefficients of the vector $v_s \in \mathbb{F}^n$ while the other elements come from the matrix $M_S \in \mathbb{F}^{n\times n}$. Apply this transformation to one quadratic polynomial a_i, as defined in Section 1.4. This yields:

$$\begin{aligned}
a_i(x'_1, \ldots, x'_n) \circ \ddot{S} &= [\gamma_{i,1,1}x'^2_1 + \gamma_{i,1,2}x'_1x'_2 + \ldots + \gamma_{i,n,n}x'^2_n + \beta_{i,1}x'_1 + \ldots + \beta_{i,n}x'_n + \alpha_i] \circ S \\
&= \gamma_{i,1,1}(s_{1,1}x_1 + \ldots s_{1,n}x_n + s_{1,0})^2 \\
&\quad +\gamma_{i,1,2}(s_{1,1}x_1 + \ldots s_{1,n}x_n + s_{1,0})(s_{2,1}x_1 + \ldots s_{2,n}x_n + s_{2,0}) + \ldots \\
&\quad +\gamma_{i,n,n}(s_{n,1}x_1 + \ldots s_{n,n}x_n + s_{n,0})^2 + \beta_{i,1}(s_{1,1}x_1 + \ldots s_{1,n}x_n + s_{1,0}) \\
&\quad + \ldots + \beta_{i,n}(s_{n,1}x_1 + \ldots s_{n,n}x_n + s_{n,0}) + \alpha_i \\
&= \gamma'_{i,1,1}x^2_1 + \gamma'_{i,1,2}x_1x_2 + \ldots + \gamma'_{i,n,n}x^2_n + \beta'_{i,1}x_1 + \ldots + \beta'_{i,n}x_n + \alpha'_i
\end{aligned}$$

for some $\gamma'_{i,j,k}, \beta'_{i,j}, \alpha'_i \in \mathbb{F}$. The interesting point is that the values $\gamma'_{i,j,k}, \beta'_{i,j}$ only depend on the affine transformation $\ddot{S}$ but not on the value α_i, i.e., the constant term of the original polynomial. Therefore, it is not only possible to have the system of equations $\mathcal{A} \in \mathcal{QE}_m(\mathbb{F}[x_{1,\ldots,n}])$ fixed, but also the second system of equations $\ddot{\mathcal{A}}$. The only part which does not only depend on the affine transformation $\ddot{S}$ but also on the solution vector $X \in \mathbb{F}^n$ is the vector of constant terms $\ddot{\alpha} \in \mathbb{F}^m$. The public key of a prover will hence consist of the triple $(\alpha, \ddot{\alpha}, \ddot{X}) \in \mathbb{F}^m \times \mathbb{F}^m \times \mathbb{F}^n$. The systems of equations $\mathcal{A}, \ddot{\mathcal{A}} \in \mathcal{QE}_m(\mathbb{F}[x_{1,\ldots,n}])$ become system parameters. The same is also true for the affine transformation $\ddot{S} \in \mathrm{AGL}_n(\mathbb{F})$ which is known to the central authority CA only. The corresponding scheme has therefore a far smaller public key.

4.2 Less Communication

In addition to the quite large public key size, the IP — and also the $\mathcal{MQ}^*$-IP-scheme — suffer from a rather large communication complexity. In this section, we will discuss possible solutions for this problem. We dwell on ideas from [Ste93, PP03].

First, we notice that for a concrete implementation, the affine transformation S is in fact not randomly generated but pseudo-randomly generated, i.e., using a random-seed r_s of, say, 80 bits, and "expanded" using a pseudo-random function. We note that it is not necessary for our purpose to have S generated cryptographic secure (cf [PP03, Sec. 6.4]). However, the seed r_s has to be chosen in a way that it is not possible to obtain an earlier or later seed r_s' — even if the attacker has access to any number of seeds $(r_s)_1, \ldots, (r_s)_N$ for some $N \in \mathbb{N}$, i.e., the seeds must be either truly random or generated with a cryptographic secure random-number generator. Having a seed r_s, we can now generate the affine transformation $S \in \mathrm{AGL}_n(\mathbb{F})$ using the LUP-decomposition of invertible matrices. Here, $L \in \mathbb{F}^n$ is an invertible lower triangular matrix, $U \in \mathbb{F}^n$ an upper triangular matrix with only 1's on its diagonal, and P a permutation of n objects. Identifying $M_S := LUP$, we make sure that the matrix M_S is invertible and hence the transformation S is affine. As a side-effect, the computation of the function S^{-1} becomes easier. All in all, we have replaced the communication of an affine transformation with $(n^2 + n) \log_2 q$ bits by 80 bits.

Second, we notice that an equation $e := (a = 0)$ for some $a \in_R \mathcal{Q}(\mathbb{F}[x_{1,\ldots,n}])$, chosen independently from a specific solution $X \in \mathbb{F}^n$, is satisfied with probability $1/q$. Moreover, we can reduce the communication complexity by a factor of $1/m$ if we do not transmit the whole system of equations $\mathcal{B}$ but only one of its equations. Let this equation be e. Given e and a solution X, the verifier can easily check if $e(X) = 0$ holds. To make sure that prover cannot cheat by transmitting an equation which is not affine to the original systems of equations $\mathcal{A}$, we make use of hash-trees [Mer80]. We describe the whole protocol, which exploits all ideas described so far, in the next section.

In order to decrease the communication complexity further, we could decrease the number of rounds on costs of a larger public key (cf [Pat96, Sec. 18]). But as this contradicts our goal from the previous section and will therefore not be elaborated in this paper.

5 Actual Scheme

In this section, we describe actual instantiations of the $\mathcal{MQ}^*$-IP protocol, using the ideas described in Section 4 and keeping the cryptanalysis of Section 3 in mind.

5.1 Protocol

For a concrete implementation, we assume that the two systems of equations $\mathcal{A}, \ddot{\mathcal{A}} \in \mathcal{QE}_m(\mathbb{F}[x_{1,\ldots,n}])$ are system parameters. The public key of a user only consists of the triple $(\alpha, \ddot{\alpha}, \ddot{X}) \in \mathbb{F}^m \times \mathbb{F}^m \times \mathbb{F}^n$ (cf Sec. 4.1). In addition, we will make use of a hash function $H(\cdot)$, e.g., RIPEMD-160 or SHA-1 (both output 160 bit, cf [DBP96, FIP95]). This function $H(\cdot)$ will also be used for bit-commitment. We use 160 bit for this purpose as smaller values would allow attacks with the birthday paradox [GS94]

In a precomputation step, the prover P picks a random seed $r_s \in_R \{0,1\}^{80}$ and derives an affine transformation $S \in \mathrm{AGL}_n(\mathbb{F})$, using LUP-decomposition (cf Section 4.2). He computes $\mathcal{B} := \mathcal{A} \circ S$ and its solution $X' := S^{-1}(X)$. In addition, he computes a hash-tree H_T of all equations b_i. Denote the root of this hash-tree with h_0. Moreover, the prover commits himself on the solution X', i.e., computes $c_0 := H(X')$. After this, the interactive scheme works as follows:

1. The prover P sends the verifier V the values (c_0, h_0).
2. The verifier V selects a number $c \in \{0, \ldots, m\}$. The number 0 is chosen with probability $\frac{1}{2}$, the other numbers are chosen with probability $\frac{1}{2m}$.
3. Depending on the number c, the prover P sends

   c = 0: the seed r_s of the transformation S.

   c ≠ 0: the solution X', equation b_c and the hash chain $(h_{i_1}, \ldots, h_{i_d})$ which authenticates the equation b_c.

In case $c = 0$, the verifier checks if $c_0 \overset{?}{=} H(r_s)$. If yes, he computes the corresponding affine transformation S, the system of equations $\mathcal{B} = \mathcal{A} \circ S$ and the corresponding hash-tree. Finally, he checks if the root of the hash-tree, i.e., h_0 is the same as transmitted by P in Step 1.

In case $c \neq 0$, the verifier computes the hash of the equation b_c, tests the integrity of the hash-tree and checks if $b_c(X') = 0$ is satisfied.

There are a few comments on this scheme. First, it is computational rather than statistical zero-knowledge: we only work with a seed r_s of 80 bits, and hence cannot generate all affine transformations $S \in \mathrm{AGL}_n(\mathbb{F})$ anymore. Therefore, the output distribution of $\mathcal{B}' = \ddot{\mathcal{A}} \circ S$ and $\mathcal{B} = \mathcal{A} \circ S$ will be different. On the other hand, the corresponding distribution is computational indistinguishable as we choose from a set of 2^{80} elements for the affine transformations S. Second, the cheating probability is no longer $\frac{1}{2}$ but rather $\frac{q+1}{2q}$ (cf Section 4.2). Therefore, we will need one additional round to obtain the same impersonation probability (2^{-32}) as in the original scheme. Third, this scheme has the additional assumptions about the one-wayness of the hash-function $H(\cdot)$ and also its collision-resistance. Finally, the communication (in bits) in the worst case is $160(2+(\lceil log_2 m\rceil)) + \lceil \log_2 m+1\rceil +$ sizeEq: the hash bits for c_0, h_0, plus the authentication chain of the hash-tree, plus the bits for the number c, plus the size of one equation b_c. In the best case, we send 80 bits for the random-seed r_s instead of the hash-tree and the equation b_c. Both cases occur with probability $\frac{1}{2}$.

5.2 16-bit and 8-bit versions

To obtain a concrete scheme, we note that the size of the public key (and also the communication complexity) profit from a larger ground-field $\mathbb{F}$. In this section, we will present two different versions of $\mathcal{MQ}$-trapdoors and the corresponding identification schemes. The first is based on a ground field of 8-bit, the second of 16-bit. All are well suited for 8-bit microprocessors as they are based on the C^{*--} trapdoor (cf Section 3.4). The actual parameters and their effects can be found in Figure 1.

5.3 Comparison with other schemes

To compare the C-IP-schemes to other schemes, we use [PP03, Fig. 10] to derive Figure 2. However, we have to take into account that this table assumes an impersonation level of

	C8-IP	C16-IP
ground-field $\mathbb{F}$	$\mathrm{GF}(2^8)$	$\mathrm{GF}(2^{16})$
extension-field $\mathbb{E}$	$\mathrm{GF}(2^{8 \cdot 37})$	$\mathrm{GF}(2^{16 \cdot 17})$
parameter λ in polynomial P(x)	$2^{8 \cdot 11} + 1$	$2^{16 \cdot 11} + 1$
variables n	37	17
equations m	26	12
equations removed r	11	5
public key size $\|K\|$ [bits]	712	656
private key size $\|k\|$ [bits]	296	272
rounds	33	33
communication [kBytes]	15.5	8.6
Size of one equation [Bytes]	741	343
Size of the $\mathcal{MQ}$-problem $\mathcal{A}$ [kBytes]	19	4

Fig. 1: $\mathcal{MQ}$-trapdoor based on C^{*--} for 16 and 8 bits

Scheme	**PKP**		**CLE**		**SD**	**PPP**	
	3p	5p	3p	5p		3p	5p
Matrix size	16 × 34		24 × 24		256 × 256	121 × 137	
over the field	$\mathbb{F}_{251}$		$\mathbb{F}_{257}$		$\mathbb{F}_2$	$\mathbb{F}_2$	
Number of Rounds	35	20	35	20	35	48	35
Public key size [bits]	272		96		256	189	
Private key size [bits]	141		96		512	137	
Communication [kBytes]	2.2	1.6	2.4	2.0	3.6	6.3	6.4

Fig. 2: Comparison of $\mathcal{NP}$-complete identification schemes

10^{-6} rather than 2^{-32} and that the overall security of the schemes has been set to 2^{64} instead of 2^{80}. It is beyond the scope of this paper to provide a full comparison for the same level of security of all schemes known so far.

As syndrome decoding (SD) is the only $\mathcal{NP}$-scheme in this table which allows identity-based public keys, we will concentrate on this scheme for a comparison. A description of the identity-based version of SD can be found in [Ste96, Sec. 4.3]. We adapted the parameters given in this paper to obtain the same level of security, i.e., 2^{-32} as impersonation probability and 2^{80} for the overall security of the scheme. Hence, we had to increase the size of several hash values to 160 and the number of rounds to 55. For the minimal parameters suggested for the first version of the scheme (i.e., $n = 512$), we obtain 7.1kB as communication complexity. The second identity based version (here, $n = 1024$) has 11.7kB.

If the communication complexity is critical, the first version of the SD scheme performs slightly better than C-IP. But if the number of rounds is important, C-IP is preferable. Using larger public keys, the figures for C-IP improve further. And finally, both SD and

C-IP outperform the other schemes when it comes to key-distribution as both allow the use of identity based keys.

6 Conclusions

In this paper, we proposed the new identification scheme $\mathcal{MQ}$-IP. The scheme is zero-knowledge and its security is based on two well-established security assumptions, namely the $\mathcal{MQ}$-problem and the IP-problem (cf Section 1.2).

Using a trapdoor for the $\mathcal{MQ}$-problem, we were able to make the scheme identity based. This way, it is no longer necessary to distribute public keys or to issue certificates. By construction of the public key, the identity of the user is linked to its public key.

Using the C^{*--}-trapdoor, we derived a practical protocol, based on $\mathcal{MQ}^*$-IP. The only other $\mathcal{NP}$-identification scheme known to the authors which allows identity-based keys is the SD-scheme. Having a security-level of 2^{80} and an impersonation probability of 2^{-32} in one run, the first SD protocol with trapdoor has a communication complexity of 7.1kB, and the second 11.7kB. In contrast, C16-IP-scheme has 8.6kB. However, SD requires a total of 55 rounds while C16-IP needs only 33.

Judging from other results concerning the implementation of the C^{*--}-trapdoor (cf [ACDG03]), we expect C-IP to be very suitable for smart-card implementations.

References

[ACDG03] Mehdi-Laurent Akkar, Nicolas T. Courtois, Romain Duteuil, and Louis Goubin. A fast and secure implementation of SFlash. In *Public Key Cryptography — PKC 2003*, volume 2567 of *Lecture Notes in Computer Science*, pages 267–278. Y. Desmedt, editor, Springer, 2002.

[CGP00] Nicolas Courtois, Louis Goubin, and Jacques Patarin. *Flash: Primitive specification and supporting documentation*, 2000. `https://www.cosic.esat.kuleuven.be/nessie`, submissions, 9 pages.

[CGP02] Nicolas Courtois, Louis Goubin, and Jacques Patarin. *Sflash: Primitive specification (second revised version)*, 2002. `https://www.cosic.esat.kuleuven.be/nessie`, Submissions, Sflash, 11 pages.

[CKPS00] Nicolas T. Courtois, Alexander Klimov, Jacques Patarin, and Adi Shamir. Efficient algorithms for solving overdefined systems of multivariate polynomial equations. In *Advances in Cryptology — EUROCRYPT 2000*, volume 1807 of *Lecture Notes in Computer Science*, pages 392–407. Bart Preneel, editor, Springer, 2000. Extended Version: `http://www.minrank.org/xlfull.pdf`.

[Cr94] Yvo Desmedt, editor. *Advances in Cryptology — CRYPTO 1994*, volume 839 of *Lecture Notes in Computer Science*. Springer, 1994. ISBN 3-540-58333-5.

[DBP96] Hans Dobbertin, Antoon Bosselaers, and Bart Preneel. RIPEMD-160: A strengthened version of RIPEMD. In *Fast Software Encryption — FSE 1996*, volume 1039 of *Lecture Notes in Computer Science*, pages 71–82. Dieter Gollmann, editor, Springer, 1996. Updated version at `http://www.esat.kuleuven.be/~cosicart/ps/AB-9601/AB-9601.ps.gz`.

[FIP95] National Institute of Standards and Technology. *Federal Information Processing Standards Publication 180-1: Secure Hash Standard*, 17th April 1995. `http://www.itl.nist.gov/fipspubs/fip180-1.htm`.

[FJ03] Jean-Charles Faugère and Antoine Joux. Algebraic cryptanalysis of Hidden Field Equations (HFE) using Gröbner bases. In *Advances in Cryptology — CRYPTO 2003*, volume 2729 of *Lecture Notes in Computer Science*, pages 44–60. Dan Boneh, editor, Springer, 2003.

[GJ79] Michael R. Garey and David S. Johnson. *Computers and Intractability — A Guide to the Theory of NP-Completeness.* W.H. Freeman and Company, 1979. ISBN 0-7167-1044-7 or 0-7167-1045-5.

[GMR85] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof-systems. In *Annual ACM Symposium on Theory of Computing — STOC 1985*, pages 291–304. Robert Sedgewick, chair, ACM Press, 1985.

[GMS02] Willi Geiselmann, Willi Meier, and Rainer Steinwandt. An attack on the Isomorphisms of Polynomials problem with one secret. Cryptology ePrint Archive, Report 2002/143, 2002. `http://eprint.iacr.org/2002/143`, version from 2002-09-20, 12 pages.

[Gol02] Oded Goldreich. Zero-knowledge twenty years after its invention. Cryptology ePrint Archive, Report 2002/186, 2002. `http://eprint.iacr.org/2002/186`, version from 2002-12-05, 33 pages.

[GS94] Marc Girault and Jacques Stern. On the length of cryptographic hash-values used in identification schemes. In Cr [Cr94], pages 202–215.

[GSB01] W. Geiselmann, R. Steinwandt, and Th. Beth. Attacking the affine parts of SFlash. In *Cryptography and Coding - 8th IMA International Conference*, volume 2260 of *Lecture Notes in Computer Science*, pages 355–359. B. Honary, editor, Springer, 2001. Extended version: `http://eprint.iacr.org/2003/220/`.

[Mer80] R. C. Merkle. Protocols for public key cryptosystems. In *IEEE Symposium on Security and Privacy 1980*, pages 122–133. IEEE Computer Society Press, 1980.

[MvOV96] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography.* CRC Press, 1996. ISBN 0-8493-8523-7, online-version: `http://www.cacr.math.uwaterloo.ca/hac/`.

[NES] NESSIE: New European Schemes for Signatures, Integrity, and Encryption. Information Society Technologies programme of the European commission (IST-1999-12324). `http://www.cryptonessie.org/`.

[Pat95] Jacques Patarin. Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt'88. In *Advances in Cryptology — CRYPTO 1995*, volume 963 of *Lecture Notes in Computer Science*, pages 248–261. Don Coppersmith, editor, Springer, 1995.

[Pat96] Jacques Patarin. Hidden Field Equations (HFE) and Isomorphisms of Polynomials (IP): two new families of asymmetric algorithms. In *Advances in Cryptology — EUROCRYPT 1996*, volume 1070 of *Lecture Notes in Computer Science*, pages 33–48. Ueli Maurer, editor, Springer, 1996. Extended Version: `http://www.minrank.org/hfe.pdf`.

[PBO$^+$03] B. Preneel, A. Biryukov, E. Oswald, B. Van Rompay, L. Granboulan, E. Dottax, S. Murphy, A. Dent, J. White, M. Dichtl, S. Pyka, M. Schenfheutle, P. Serf, E. Biham, E. Barkan, O. Dunkelman, J.-J. Quisquater, M. Ciet, F. Sica, L. Knudsen, M. Parker, and H. Raddum. NESSIE security report, version 2.0. Document NES/DOC/ENS/WP5/D20/2, 19th of February 2003. `https://www.cosic.esat.kuleuven.be/nessie/deliverables/D20-v2.pdf`, see [NES], 342 pages.

[PG97] Jacques Patarin and Louis Goubin. Trapdoor one-way permutations and multivariate polynomials. In *International Conference on Information Security and Cryptology 1997*, volume 1334 of *Lecture Notes in Computer Science*, pages 356–368. International Communications and Information Security Association, Springer, 1997. Extended Version: `http://citeseer.nj.nec.com/patarin97trapdoor.html`.

[PGC98a] Jacques Patarin, Louis Goubin, and Nicolas Courtois. C^*_{-+} and HM: Variations around two schemes of T.Matsumoto and H.Imai. In *Advances in Cryptology — ASIACRYPT 1998*, volume 1514 of *Lecture Notes in Computer Science*, pages 35–49. Kazuo Ohta and Dingyi Pei, editors, Springer, 1998. Extended Version: `http://citeseer.nj.nec.com/patarin98plusmn.html`.

[PGC98b] Jacques Patarin, Louis Goubin, and Nicolas Courtois. Improved algorithms for Isomorphisms of Polynomials. In *Advances in Cryptology — EUROCRYPT 1998*, volume 1403 of *Lecture Notes in Computer Science*, pages 184–200. Kaisa Nyberg, editor, Springer, 1998. Extended Version: `http://www.minrank.org/ip6long.ps`.

[Poi95] David Pointcheval. A new identification scheme based on the perceptrons problem. In *Advances in Cryptology — EUROCRYPT 1995*, volume 921 of *Lecture Notes in Computer Science*, pages 319–328. Louis C. Guillou and Jean-Jacques Quisquater, editors, Springer, 1995.

[PP03] David Pointcheval and Guillaume Poupard. A new $\mathcal{NP}$-complete problem and public-key identification. *Designs, Codes and Cryptography*, 28(1):5–31, January 2003. ISSN 0925–1022.

[Sha84] Adi Shamir. Identity-based cryptosystems and signature schemes. In *Advances in Cryptology — CRYPTO 1984*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. G. R. Blakley and David Chaum, editors, Springer, 1984.

[Sha89] Adi Shamir. An efficient identification scheme based on permuted kernels (extended abstract). In *Advances in Cryptology — CRYPTO 1989*, volume 435 of *Lecture Notes in Computer Science*, pages 606–609. Gilles Brassard, editor, Springer, 1989.

[Ste93] Jacques Stern. A new identification scheme based on syndrome decoding. In *Advances in Cryptology — CRYPTO 1993*, volume 773 of *Lecture Notes in Computer Science*, pages 13–21. Douglas R. Stinson, editor, Springer, 1993.

[Ste94] Jacques Stern. Designing identification schemes with keys of short size. In Cr [Cr94], pages 164–173.

[Ste96] Jacques Stern. A new paradigm for public key identification. *IEEE Transactions on Information Theory*, 42(6):1757–1768, 1996.