Zero-Knowledge Protocols for the McEliece Encryption

Kirill Morozov and Tsuyoshi Takagi

Institute of Mathematics for Industry, Kyushu University, Japan
{morozov,takagi}@imi.kyushu-u.ac.jp

Abstract. We present two zero-knowledge protocols for the code-based McEliece public key encryption scheme in the standard model. Consider a prover who encrypted a plaintext m into a ciphertext c under the public key pk. The first protocol is a proof of plaintext knowledge (PPK), where the prover convinces a polynomially bounded verifier on a joint input (c, pk) that he knows m without actually revealing it. This construction uses code-based Véron's zero-knowledge identification scheme. The second protocol, which builds on the first one, is a verifiable McEliece encryption, were the prover convinces a polynomially bounded verifier on a joint input (c, pk, m) that c is a valid encryption of m, without performing decryption. These protocols are the first PPK and the first verifiable encryption for code-based cryptosystems.

1 Introduction

The McEliece public key encryption (PKE) scheme [26] is the first code-based cryptosystem. It uses the error-correcting codes by Goppa [20,25]. Security of the McEliece PKE is based on hardness of the problems related to general decoding [5,31]. Breaking of the McEliece PKE is believed to be infeasible for properly chosen parameters [12,13,7], even for adversaries equipped with quantum computers [6]. The later fact makes this cryptosystem a prospective candidate for the postquantum world. In fact, it is also argued by Bernstein et al [7, App. A] that the McEliece PKE is a prospective cryptosystem due to its good asymptotic performance.

Informally, a proof of plaintext knowledge (PPK) for an encryption scheme with public key pk, allows a prover P to prove knowledge of the plaintext m, corresponding to the ciphertext $c = Enc_{pk}(m)$, to a verifier V on the public inputs pk and c. Moreover, if such the proof is zero-knowledge (ZK), it will not reveal any additional information on m.

Informally, a verifiable encryption with respect to some binary relation R on the plaintexts is a zero-knowledge proof on public inputs pk, c, and δ that allows P to convince V that c is a ciphertext of m under pk such that $(m, \delta) \in R$.

1.1 Our Contributions

- We present a computational zero-knowledge PPK for the McEliece PKE using Véron's ZK identification scheme [35].

W. Susilo, Y. Mu, and J. Seberry (Eds.): ACISP 2012, LNCS 7372, pp. 180–193, 2012.

– Using this PPK, we also construct a verifiable IND-CPA McEliece encryption for equality relation by introducing a computational ZK proof of the statement "ciphertext c decrypts to the plaintext m".

1.2 Related Works

Proof of Plaintext Knowledge. PPK were introduced by Aumann and Rabin [1] (as attributed in [23]), and later studied by Katz [23], who presented PPK for RSA, Rabin, ElGamal and Paillier cryptosystems. The first PPK for a lattice-based Ajtai-Dwork PKE is due to Goldwasser and Kharchenko [19]. Xagawa et al [36] presented PPK for the two variants of the lattice-based Regev's cryptosystem. Xagawa and Tanaka presented that for NTRU [37] using a modification of Stern's code-based ZK identification scheme [34]. Bendlin and Damgård [3] presented a PPK for a variant of Regev's cryptosystem. Compared to the previous lattice-based constructions (as well as to our protocols) the latter scheme is constant-round that is achieved using the "multiparty computation in the head" paradigm of Ishai et al [21].

Stern's scheme [34] was used by Kobara et al [24] for enforcing correct behavior of a sender in code-based oblivious transfer. They even suggested verifiable encryption as a possible application for their technique, but no formal treatment of this subject was made in their work.

Verifiable Encryption. Verifiable encryption was introduced by Stadler [33] in the context of publicly verifiable secret sharing, and later generalized by Asokan et al [2] with application to fair exchange of digital signatures. Developments on this topic include further generalizations by Camenisch and Damgård [8] and Camenisch and Shoup [9].

We emphasize that none of the previous works on the above topics considered code-based PKE.

Note that assuming that one-way functions exist, one could achieve the results presented in this work using general zero-knowledge proofs for NP-statements [18], however such constructions would be prohibitively inefficient.

1.3 Discussion of Our Contributions

We present a computational zero-knowledge PPK for the McEliece PKE by showing that Véron's ZK identification scheme [35] (that is, in a sense, a dual of Stern's scheme [34]) can be directly used as PPK for the McEliece encryption. The witness in this proof is both the plaintext and the (random) error vector. Using Véron's scheme rather than Stern's (as in [24]), we avoid pre-computation on the public data.

An immediate application of this result is the interactive chosen-ciphertext secure encryption. Here, the sender uses an IND-CPA secure PKE to encrypt a message for the receiver, who must be online. Along with transmitting the ciphertext, the sender also uses the interactive PPK to convince the receiver that he knows the message. According to the observation by Katz [23], this construction results in an interactive IND-CCA1 PKE [15,16]. Combined with the IND-CPA secure McEliece encryption by Nojima et al [29], this yields the first code-based interactive IND-CCA1 PKE in the standard model.

Using the above mentioned PPK, we also construct a verifiable IND-CPA McEliece encryption for equality relation. Note that although the original McEliece encryption is not deterministic, given pk and $c = Enc_{pk}(m)$, it is trivial to check whether or not c is a ciphertext of m. Therefore, for verifiable encryption, we use an IND-CPA secure McEliece encryption [29].

It is interesting to note that in the lattice-based constructions [19,36], one first constructs a verifiable encryption for equality relation, and then use it as a building block for PPK, while in our case it works the other way around.

In our constructions, we assume that both the prover and the verifier are assured that the public key pk is valid. This assumption will require a trusted third party who generates public keys – this can be, for instance, an entity in the public key infrastructure.

The proofs of Stern's [34] and Véron's schemes [35] are in the random oracle model. In order to avoid such the strong assumption, we employ the later scheme with (efficient) computationally hiding and statistically binding commitment scheme based on hardness of syndrome decoding, as presented in [11].

2 Preliminaries

Let us fix some notation. Denote by "$\oplus$" the bitwise exclusive-or. For an ordered subset $\{j_i, \dots, j_m\} = J \subseteq \{1, \dots, n\}$, we denote the vector $(x_{j_1}, \dots, x_{j_m}) \in \mathbb{F}_2^m$ by x_J. Similarly, we denote by M_J the submatrix of a $(k \times n)$ matrix M consisting of the columns corresponding to the indexes of J. A concatenation of vectors x and y is written as $(x|y)$. We denote by $x \overset{\$}{\leftarrow} \mathcal{X}$ a uniformly random selection of an element from its domain $\mathcal{X}$. A set of $(n \times n)$ permutation matrices is denoted by $\mathcal{S}_n$.

We denote by $\langle A(a), B(b)\rangle(c)$ a random variable representing the output of a Turing machine B following an execution of an interactive two-party protocol between a Turing machine A with private input a and B with private input b on joint input c, where A and B have uniformly distributed random tapes. If a party, say A, has no input, then we omit the input by writing just A (instead of $A(a)$) in the above notation.

In our two-party protocols, we will denote an honest prover by P and an honest verifier by V, while a dishonest party will be denoted by $\widetilde{\mathsf{P}}$ and $\widetilde{\mathsf{V}}$, respectively.

We call a function $\epsilon(n)$ *negligible in* n, if $\epsilon(n) = 2^{-\omega(\log n)}$. We call a probability $1 - \epsilon(n)$ *overwhelming*, when $\epsilon(n)$ is negligible.

Occasionally, we omit the mentioning of a security parameter. In these cases, by saying that a quantity is negligible (overwhelming), we mean that it is *negligible (overwhelming) in the security parameter.*

For the relevant topics in coding theory we refer the reader to [30,25].

2.1 Security Assumptions

Definition 1 (Syndrome Decoding (SD) Problem).

Input: $H \xleftarrow{\$} \mathbb{F}_2^{(n-k)\times n}$, $y \xleftarrow{\$} \mathbb{F}_2^{n-k}$ *and* $0 < t \in \mathbb{N}$.

Output: $s \in \mathbb{F}_2^n$ *such that* $w_H(s) \leq t$, $Hs^T = y$.

This problem was shown to be NP-complete by Berlekamp et al [5]. Its equivalent dual version can be formulated as follows.

Definition 2 (General Decoding (G-SD) Problem).

Input: $G \xleftarrow{\$} \mathbb{F}_2^{k\times n}$, $y \xleftarrow{\$} \mathbb{F}_2^{n}$ *and* $0 < t \in \mathbb{N}$.

Output: $x \in \mathbb{F}_2^k$, $e \in \mathbb{F}_2^n$ *s.t.* $w_H(e) \leq t$, $xG \oplus e = y$.

The following two problems use the quantities defined in the next subsection. No polynomial-time algorithm is known for these problems [12,13,7].

Definition 3 (McEliece Problem).

Input: A McEliece public key (G^{pub}, t)*, where*
$G^{pub} \in \mathbb{F}_2^{k\times n}$, $0 < t \in \mathbb{N}$*; and a McEliece ciphertext* $c \in \mathbb{F}_2^n$.

Output: $m \in \mathbb{F}_2^k$ *such that* $d_H(mG^{pub}, c) = t$.

Definition 4 (Goppa Code Distinguishing (GD) Problem).

Input: $R \in \mathbb{F}_2^{k\times n}$.

Decide: Is R *a generator matrix of an* (n, k) *irreducible Goppa code, or of a random* (n, k)*-code?*

2.2 McEliece Cryptosystem

For a survey on code-based PKE and related schemes we refer the reader to the work by Engelbert et al [12].

The McEliece PKE consists of the following triplet of algorithms $(\mathcal{K}, \mathcal{E}, \mathcal{D})$:

- Security parameters: $n, t \in \mathbb{N}$.
- Key generation algorithm $\mathcal{K}$: On input n, t, generate the following matrices:
  - $G \in \mathbb{F}_2^{k\times n}$ – the generator matrix of an irreducible binary Goppa code correcting up to t errors. Its decoding algorithm is denoted as Dec.
  - $S \in \mathbb{F}_2^{k\times k}$ – a random non-singular matrix.
  - $P \in \mathbb{F}_2^{n\times n}$ – a random permutation matrix (of size n).
  - $G^{pub} = SGP \in \mathbb{F}_2^{k\times n}$.

  Output the public key $pk = (G^{pub}, t)$ and the secret key $sk = (S, G, P, \mathsf{Dec})$.
- Encryption algorithm $\mathcal{E}$: On input a plaintext $m \in \mathbb{F}_2^k$ and the public key pk, choose a vector $e \in \mathbb{F}_2^n$ of weight t at random, and output the ciphertext

$$c = mG^{pub} \oplus e.$$

- Decryption algorithm $\mathcal{D}$: On input c and the secret key sk, calculate:

- $cP^{-1} = (mS)G \oplus eP^{-1}$.
- $mSG = \mathsf{Dec}(cP^{-1})$.
- Let $J \subseteq \{1, \ldots, n\}$ be s.t. G_J is invertible.
  Output $m = (mSG)_J(G_J)^{-1}S^{-1}$.

It is easy to check that the decryption algorithm correctly recovers the plaintext: Since in the first step of decryption, the permuted error vector $\mathbf{e}P^{-1}$ is again of weight t, the decoding algorithm Dec successfully corrects these errors in the next step.

Randomized McEliece Encryption. In the standard model, Nojima et al [29] show that the McEliece encryption with a random padding of the plaintext (which is multi-bit) is IND-CPA secure under hardness of the learning parities with noise (LPN) problem[1] and GD problem.

A little more formally, the Randomized McEliece encryption is constructed in the same way as described above, except that the ciphertext $c = (r|m)G^{pub} \oplus e$, where $r \overset{\$}{\leftarrow} \{0,1\}^{k_0}$, $m \in \{0,1\}^{k_1}$, $k = k_0 + k_1$. A particular choice of k_0 and k_1 is discussed in [29].

2.3 Proof of Plaintext Knowledge

In this subsection, we closely follow the presentation of [23]. For a public key cryptosystem $(\mathcal{K}, \mathcal{E}, \mathcal{D})$, denote by $c = \mathcal{E}_{pk}(m; R)$ an encryption of a plaintext m under public key pk using randomness R. We will call (m, R) a *witness* to the decryption of c under pk. Informally, in a PPK protocol, a sender P proves to a receiver V the knowledge of a witness to the decryption for some ciphertext c under the known public key pk.

Definition 5. *Let $\Pi = (\mathsf{P}, \mathsf{V})$ be a tuple of PPT algorithms. Π is a* proof of plaintext knowledge *for encryption scheme $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ if the following conditions hold:*

(Completeness) *For all pk output by $\mathcal{K}(1^n)$ and all c with witness w to the decryption of c under pk, we have that* $\Pr[\langle \mathsf{P}(w), \mathsf{V}\rangle(pk, c) = 1]$. *(When V outputs 1 we say it* accepts*.)*

(Soundness) *For all pk output by $\mathcal{K}(1^n)$, all c produced under pk, and for any $\widetilde{\mathsf{P}}$, we have that* $\Pr[\langle \widetilde{\mathsf{P}}, \mathsf{V}\rangle(pk, c) = 1]$ *is negligible.*

(Zero-knowledge) *There exists a PPT Turing machine $\mathcal{SIM}$ (called a* simulator*) such that, for all pk output by $\mathcal{K}(1^n)$, all PPT $\widetilde{\mathsf{V}}$, and all w, the following distributions are computationally indistinguishable:*

$$\{c = \mathcal{E}_{pk}(m; R) : \langle \mathsf{P}(w), \widetilde{\mathsf{V}}\rangle(pk, c)\},$$

$$\{c = \mathcal{E}_{pk}(m; R) : \langle \mathcal{SIM}, \widetilde{\mathsf{V}}\rangle(pk, c)\}.$$

[1] See e.g. [29] for a formal definition of LPN problem – it is similar to G-SD problem except that in the error vector e, each bit has Bernoulli distribution with fixed p, $0 < p < 0.5$.

2.4 Verifiable Encryption

We adapt the following definition from [8].

Definition 6. *Let $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a public key encryption scheme, let R be a binary relation and let $L_R = \{x | \exists w : (x, w) \in R\}$. A* secure verifiable encryption *scheme for a relation R consists of a two-party protocol between P and V s.t. the following conditions hold:*
(Completeness) *For all pk output by $\mathcal{K}(1^n)$ and all $x \in L_R$, we have* $\Pr[\langle \mathsf{P}(x), \mathsf{V}\rangle(pk) = 1]$. *(When V outputs 1 we say it* accepts.*)*
(Soundness) *For all pk output by $\mathcal{K}(1^n)$, all $x' \notin L_R$, and for any $\widetilde{\mathsf{P}}$,* $\Pr[\langle \widetilde{\mathsf{P}}(x'), \mathsf{V}\rangle(pk, c) = 1]$ *is negligible.*
(Zero-knowledge) *There exists a PPT simulator $\mathcal{SIM}$ such that, for all pk output by $\mathcal{K}(1^n)$, all PPT $\widetilde{\mathsf{V}}$, and all $x \in L_R$, the following distributions are computationally indistinguishable:*

$$\{x \in L_R : \langle \mathsf{P}(x), \widetilde{\mathsf{V}}\rangle(pk)\},\ \{x \in L_R : \langle \mathcal{SIM}, \widetilde{\mathsf{V}}\rangle(pk)\}.$$

Note that this definition captures only the properties related to verifiability. We implicitly assume that a scheme in question is indeed a public-key encryption scheme. For a formal definition of the latter, see e.g. [17, Ch. 5].

2.5 Commitments

Zero-knowledge proof systems use commitments as a building block. A commitment scheme consists of two stages. In the first one, called *committing*, the sender P provides the receiver V with an evidence about his data b. The cheating receiver $\widetilde{\mathsf{V}}$ cannot learn b before the second stage, called *opening*, when P reveals b to V. The cheating sender $\widetilde{\mathsf{P}}$ cannot successfully open anything other than b. Let us denote by $[\mathsf{P}, \mathsf{V}]_{A,st}$ the *view* of the party $A \in \{\mathsf{P}, \mathsf{V}\}$ at the stage st, which is a concatenation of all the messages sent and received by A, along with its local randomness.

We adapt the following definition from [11].

Definition 7. *A protocol is said to* securely implement string commitment*, if at the end of its execution by PPT Turing machines P (with input $b \in \mathbb{F}_2^l$, $l \in \mathbb{N}$) and V, the following properties hold:*
(Correctness) $\Pr[\langle \mathsf{P}(b), \mathsf{V}\rangle = 1]$ *with overwhelming probability.*
(Hiding) *For any PPT $\widetilde{\mathsf{V}}$, any $l \in \mathbb{N}$, any $b \in \mathbb{F}_2^l$ and $b' \in \mathbb{F}_2^l$ such that $b' \neq b$, after the committing stage, but before the opening stage, the distributions*

$$[\mathsf{P}(b), \widetilde{\mathsf{V}}]_{\widetilde{\mathsf{V}},Commit} \quad \textit{and} \quad [\mathsf{P}(b'), \widetilde{\mathsf{V}}]_{\widetilde{\mathsf{V}},Commit}$$

are computationally indistinguishable.
(Binding) *For any $\widetilde{\mathsf{P}}$, any $l \in \mathbb{N}$, and $b' \in \mathbb{F}_2^l$ there exists $b \in \mathbb{F}_2^l$ which can be computed by P after the committing stage, such that the probability*

$$\Pr[\langle \widetilde{\mathsf{P}}(b'), \mathsf{V}\rangle = 1]$$

is negligible.

In the random oracle model, a string commitment which is both computationally hiding and binding can be implemented using (idealized) cryptographic hash functions. We avoid this additional strong assumption by employing a computationally hiding and statistically binding commitment based on syndrome decoding, which was suggested by Dowsley et al [11]. They proposed to use Naor's bit commitment scheme [27] based on pseudorandom generator, which, in turn, can be constructed assuming hardness of SD problem, as proved by Fischer and Stern [14].

3 PPK for McEliece Encryption

Our proof of knowledge for the McEliece encryption is based on Véron's zero-knowledge identification scheme [35]. We make the following modifications to it – instead of a generator matrix of the random code, we use that of the irreducible (n, k) Goppa code as described in Section 2.2, and set a weight of the error vector to be exactly t.

Our main observation is that the security proof of Véron's scheme [35] is valid for any code, for which G-SD problem is hard, not just a random one. Therefore, replacing a random code with the McEliece public key, and an assumption on the hardness of G-SD problem with that on the hardness of the McEliece problem, we preserve the validity of the original proof.

Remark 1. Note that we do not need to assume hardness of the Goppa Distinguishing problem for the proof itself.

Remark 2. In the following protocol, the probability for $\widetilde{\mathsf{P}}$ to break soundness (i.e. to make V accept the proof without knowledge of the witness (m, e)) is $2/3$. It can be reduced to an arbitrary small value $(2/3)^s$ by iterating the protocol s times.

Witness: (m, e), $m \in \mathbb{F}_2^k$, $e \in \mathbb{F}_2^n$, $w_H(e) = t$, where the parameters n, k, t are described in Section 2.2.

Common data: $(G^{pub} \in \mathbb{F}_2^{k \times n}, t)$ – the McEliece public key, and $c = mG^{pub} \oplus e$ – the McEliece PKE ciphertext (as described in Section 2.2).

Protocol 1 (McEliece PPK).

1. P computes $u \overset{\$}{\leftarrow} \mathbb{F}_2^k$, $T \overset{\$}{\leftarrow} \mathcal{S}_n$ and sends three commitments:
   – $C_1 = com(T)$,
   – $C_2 = com((u \oplus m)G^{pub}T)$,
   – $C_3 = com((uG^{pub} \oplus c)T)$.
2. V sends $b \overset{\$}{\leftarrow} \{0, 1, 2\}$.
3. In this step, V checks the validity of the quantities presented by P, and rejects if it does not hold:

- If $b = 0$,
  – P sends T, $u \oplus m$, and opens C_1, C_2.
  – V checks validity of C_1 and C_2 (using G^{pub}).

- If $b = 1$,
  - P sends $(u \oplus m)G^{pub}T$, eT, and opens C_2, C_3.
  - V checks that $w_H(eT) = t$ and validity of C_2, C_3 (using that $(u \oplus m)G^{pub}T \oplus eT = (uG^{pub} \oplus c)T$).
- If $b = 2$,
  - P sends T, u, and opens C_1, C_3.
  - V checks the validity of C_1, C_3.

Denote a protocol consisting of s independent iterations of Protocol 1 by $\mathsf{PPK}(G^{pub}, c; m, e)$, with some appropriately chosen s.

Theorem 1. *Protocol* $\mathsf{PPK}(G^{pub}, c; m, e)$ *is a proof of plaintext knowledge for the McEliece cryptosystem according to Definition 5 assuming hardness of the McEliece problem.*

Proof. We closely follow the proof in [35].

Completeness. It is easy to check that P knowing a valid (m, e) for G^{pub} can answer any of the queries correctly. Hence, we have $\Pr[\langle \mathsf{P}(w), \mathsf{V}\rangle(pk, c) = 1]$.

Soundness. First, we prove the following lemma.

Lemma 1. *If* V *accepts* $\widetilde{\mathsf{P}}$*'s proof with probability at least* $(\frac{2}{3})^s + \epsilon$*, then there exists a PPT algorithm* M *which, with overwhelming probability, computes a witness* (m, e)*.*

Proof. Let $\mathcal{T}$ be an execution tree of the protocol $(\widetilde{\mathsf{P}}, \mathsf{V})$ corresponding to all possible questions of V, when $\widetilde{\mathsf{P}}$ has a random tape RA. V may ask 3 possible questions at each stage. First, we show that as long as the binding property of the commitment holds, a witness (m, e) can be computed from a vertex with 3 descendants. Next, we show that a PPT M can find such a vertex in $\mathcal{T}$ with overwhelming probability.

Let v be a vertex with 3 descendants. This corresponds to a situation, where 3 commitments C_1, C_2, C_3 have been made and where the three queries were correctly answered.

Let T' and $u' \oplus m'$ be the answers to the query $b = 0$, y'', e'' – to the query $b = 1$, T''', u''' – to the query $b = 2$.

We have $w_H(e'') = t$, $T' = open(C_1) = T'''$,
$(u' \oplus m')G^{pub}T' = open(C_2) = y''$,
$y'' \oplus e'' = open(C_3) = (u'''G^{pub} \oplus c)T'''$.

Therefore, either $\widetilde{\mathsf{P}}$ was able to violate the binding property of the commitment, or we have $c = (u' \oplus m' \oplus u''')G^{pub} \oplus e''(T')^{-1}$, where $e''(T')^{-1}$ is a word of length n and weight t. Therefore, $(u' \oplus m' \oplus u''', e''(T')^{-1})$ is a valid witness.

Next, we show that the probability for $\mathcal{T}$ to have a vertex with 3 descendants is at least ϵ. Let us consider the random tape RA of $\widetilde{\mathsf{P}}$ as a set of μ elements, from which $\widetilde{\mathsf{P}}$ randomly picks its values and let $Q = \{1, 2, 3\}$. These two sets are considered as probability spaces, both of them with uniform distribution.

A pair $(a, b) \in (RA \times Q)^s$ represents the commitments, queries and answers communicated between $\widetilde{\mathsf{P}}$ and V in the protocol. We will call (a, b) a *valid* pair, if the execution of $(\widetilde{\mathsf{P}}, \mathsf{V})$ leads to the success state.

Let V be the subset of $(RA \times Q)^s$ composed of all the valid pairs. By the hypothesis of the lemma,

$$\frac{|V|}{|(RA \times Q)^s|} \geq \left(\frac{2}{3}\right)^s + \epsilon.$$

Let $\Omega_s \subset RA^s$ such that:

- If $a \in \Omega_s$, then $2^s + 1 \leq |\{b : (a, b) \text{ are valid}\}| \leq 3^s$,
- If $a \in RA^s \setminus \Omega_s$, then $0 \leq |\{b : (a, b) \text{ are valid}\}| \leq 2^s$.

Then, we write $V = \{\text{valid } (a, b), a \in \Omega_s\} \cup \{\text{valid } (a, b), a \in RA^s \setminus \Omega_s\}$, therefore $|V| \leq |\Omega_s| \cdot 3^s + (\mu^s - |\Omega_s|) \cdot 2^s$, so by noting that $|RA^s| = \mu^s$ and $|Q^s| = 3^s$ it follows that

$$\frac{|V|}{|(RA \times Q)^s|} \leq \left(\frac{|\Omega_s|}{|RA^s|} + 2^s \left(3^{-s} - \frac{|\Omega_s|}{|RA \times Q)^s|}\right)\right) \leq \frac{|\Omega_s|}{|RA^s|} + \left(\frac{2}{3}\right)^s, \quad (1)$$

and therefore $|\Omega_s|/|RA^s| \geq \epsilon$. This shows that the probability that $\widetilde{\mathsf{P}}$ answers to (at least) $2^s + 1$ V's queries, by choosing random values, is bigger than ϵ.

Now, if more than $2^s + 1$ queries are correctly answered by $\widetilde{\mathsf{P}}$, $\mathcal{T}(RA)$ has at least $2^s + 1$ leaves, i.e. $\mathcal{T}(RA)$ has at least one vertex with 3 descendants.

Therefore, by rewinding $\widetilde{\mathsf{P}}$ $1/\epsilon$ times, it is possible to find an execution tree with a vertex having 3 descendants with probability arbitrary close to 1. This concludes the proof of the lemma. □

Unless the binding property of the commitment was violated, the conclusion of this lemma contradicts hardness of the McEliece problem. It follows that $\Pr[\langle \widetilde{\mathsf{P}}, \mathsf{V} \rangle (pk, c) = 1] \leq (2/3)^s + \epsilon$, which is negligible in n and s.

Zero-Knowledge. Let us denote by $\mathcal{R}_{\mathsf{P},\mathsf{V}}$ the communication tape for P and V, that is a concatenation of all bits they exchanged during the protocol. We consider the probability distributions on $\mathcal{R}_{\mathsf{P},\mathsf{V}}$.

Proposition 1. *Protocol 1 is zero-knowledge according to Definition 5 assuming hardness of the McEliece problem.*

Proof. In order to simulate $\widetilde{\mathsf{V}}$, we have to assume that it will choose a particular cheating strategy depending on the information received from P. Let us denote this strategy by $St(C_1, C_2, C_3) \in \{0, 1, 2\}$.

Consider the following two functions: $\phi_m : \mathbb{F}_2^k \to \mathbb{F}_2^k$, $\phi_m(u) = u \oplus m$, which is an automorphism of $\mathbb{F}_2^k$ and $\psi : \mathbb{F}_2^k \to \mathbb{F}_2^n$, $\psi(u) = uG^{pub}$, which is an isomorphism of $\mathbb{F}_2^k$ into the code generated by G^{pub}.

The following PPT algorithm $\mathcal{SIM}$ produces a communication tape, whose probability distribution is indistinguishable from that of a communication tape produced by the honest parties.

1. $\mathcal{SIM}$ randomly picks a query $b \in \{0, 1, 2\}$.
   - If $b = 0$, $\mathcal{SIM}$ chooses $y \xleftarrow{\$} \mathbb{F}_2^k$, $T \xleftarrow{\$} \mathcal{S}_n$, computes $C_1 = com(T)$, $C_2 = com(yG^{pub}T)$ and sets C_3 to be a random binary vector of appropriate length.
     Let $COM = (C_1|C_2|C_3)$ and $Ans = (y|T)$, here we assume a representation of $T \in \mathcal{S}_n$ as a binary vector by concatenating its rows. Note that y and $u \oplus m$ have the same probability distribution, since for some $z \in \mathbb{F}_2^k$, and $u \xleftarrow{\$} \mathbb{F}_2^k$, we have $\Pr[u \oplus m = z] = \Pr[u = \phi_m^{-1}(z)] = 2^{-k}$ $= \Pr[y = z]$.
   - If $b = 1$, $\mathcal{SIM}$ chooses $T \xleftarrow{\$} \mathcal{S}_n$, $y \xleftarrow{\$} \mathcal{C}$ (where $\mathcal{C}$ is a code generated by G^{pub}), $e' \xleftarrow{\$} W_2^{n,t}$ (where $W_2^{n,t} = \{x \in \mathbb{F}_2^n | w_H(x) = t\}$), computes $C_2 = com(yT)$, $C_3 = com((y \oplus e')T)$, and sets C_1 to be a random binary vector of appropriate length.
     Let $COM = (C_1|C_2|C_3)$ and $Ans = (yT|e'T)$. Then, $e'T$ has the same probability distribution as eT, moreover for some $z \in \mathcal{C}$,
     $\Pr[(u \oplus m)G^{pub} = z] = \Pr[u = \phi_m^{-1}(\psi^{-1}(z))] = 2^{-k} = \Pr[y = z]$.
   - If $b = 2$, $\mathcal{SIM}$ chooses $y \xleftarrow{\$} \mathbb{F}_2^k$, $T \xleftarrow{\$} \mathcal{S}_n$, computes $C_1 = com(T)$, $C_3 = com((yG^{pub} \oplus c)T)$ and sets C_2 to be a random binary vector of appropriate length.
     Let $COM = (C_1|C_2|C_3)$ and $Ans = (y|T)$.
2. $\mathcal{SIM}$ computes $b' = St(COM)$.
3. If $b = b'$, then $\mathcal{SIM}$ writes on the tape $\mathcal{R}$ the quantities H, b, and Ans, otherwise $\mathcal{SIM}$ goes to Step 1.

Thus, in $3s$ rounds on average, the simulator $\mathcal{SIM}$ produces a communication tape $\mathcal{R}$ computationally indistinguishable from a communication tape $\mathcal{R}_{\mathsf{P},\mathsf{V}}$ produced by the honest parties running s rounds of Protocol 1. Therefore, we have that $\langle \mathsf{P}(m, e), \widetilde{\mathsf{V}} \rangle (pk, c)$ and $\langle \mathcal{SIM}, \widetilde{\mathsf{V}} \rangle (pk, c)$ are computationally indistinguishable. Note that computational indistinguishability is due to the fact that the bit commitment scheme is computationally hiding according to Definition 7. This completes the proof of the proposition. □

The above arguments of completeness, soundness and zero-knowledge conclude the proof of the theorem. □

By inspecting the construction of the randomized McEliece PKC in Sec. 2.2, the next Corollary follows immediately by replacing m with $(r|m)$ in Theorem 1.

Corollary 1. *Protocol McEliece PPK is a proof of plaintext knowledge for the Randomized McEliece PKE of [29] assuming hardness of the McEliece problem.*

3.1 Extensions

Similarly to the above construction, PPK for the Niederreiter PKE [28] (the dual of the McEliece PKE), or its semantically secure variant [29], can be constructed in a straight forward manner using Stern's zero-knowledge identification scheme [34] (the dual of Véron's scheme [35]).

We believe but do not prove formally that the result of this section can also be extended to provide PPK for the q-ary variants of the McEliece encryption [22,4] using the identification scheme by Cayrel et al [10], which is based on q-ary codes.

4 Verifiable McEliece Encryption

Let us denote by 0^l an all-zero vector of length $l \in \mathbb{N}$.

In this section, we present the verifiable IND-CPA McEliece encryption for the equality relation $R_{eq} = \{(m, m')|m = m'\}$, i.e. that a given ciphertext c is an encryption of a given plaintext m under public key G^{pub}.

Let the parameters n, k, k_0, k_1, t be as described in Section 2.2, in particular, $k = k_0 + k_1$ and $m \in \mathbb{F}_2^{k_1}$.

Witness: (r, e), where $r \in \mathbb{F}_2^{k_0}$, $e \in \mathbb{F}_2^n$, $w_H(e) = t$.

Common data: $(G^{pub} \in \mathbb{F}_2^{k \times n}, t)$ – the McEliece public key, and
$c = (r|m)G^{pub} \oplus e$ – the Randomized McEliece PKE ciphertext (as described in Section 2.2).

Remark 3. For the ciphertext c as defined above, we have that $c \oplus (0^{k_0}|m)G^{pub} = (r|0^{k_1})G^{pub} \oplus e = rG_r^{pub} \oplus e$, where $G_r^{pub} \in \mathbb{F}_2^{k_0 \times n}$ is a restriction of G^{pub} to its first k_0 rows.

Protocol 2 (Verifiable McEliece PKE).

1. P and V execute $\mathsf{PPK}(G^{pub}, c; (r|m), e)$.
   If PPK was rejected, then V rejects.
2. P and V each compute:
   $c_r = c \oplus (0^{k_0}|m)G^{pub} = rG_r^{pub} \oplus e$.
3. P and V execute $\mathsf{PPK}(G_r^{pub}, c_r; r, e)$.
   If PPK was rejected, then V rejects,
   otherwise V accepts.

Proposition 2. *Protocol 2 is a verifiable McEliece encryption for the relation R_{eq} under hardness of G-SD, LPN and GD problems.*

Proof (Sketch). We need to argue completeness, zero-knowledge, and soundness. The first two properties follow easily using the proof of Theorem 1.

As for soundness, Step 1 ensures that c is indeed of the form $(r'|m')G^{pub} \oplus e$ with $w_H(e) = t$ for some $r' \in \mathbb{F}_2^{k_0}$ and $m' \in \mathbb{F}_2^{k_1}$. Now, suppose $m \neq m'$, then we have $c_r' = r'G_r^{pub} \oplus e \oplus (m \oplus m')G_m^{pub}$, where $G_m^{pub} \in \mathbb{F}_2^{k_1 \times n}$ is a restriction of G^{pub} to its last k_1 rows. Note that $(m \oplus m')G_m^{pub}$ is not in a code generated by G_r^{pub}, since the rows of G^{pub} are linearly independent. However, since $(m \oplus m')G_m^{pub}$ is a codeword of the code generated by G^{pub}, its weight is at least $d \geq 2t + 1$. Therefore, the weight of $e \oplus (m \oplus m')G_m^{pub}$ is at least $t + 1$. This implies that if $\widetilde{\mathsf{P}}$ was accepted by V, he necessarily used an error vector of weight larger than t, that would contradict to soundness of Protocol 1 established by Theorem 1.

We note that although the above protocol does not reveal any information on the witness (r, e), the verifier learns the plaintext m and hence she will be able to construct a valid ciphertext of Randomized McEliece encryption with randomness (r, e) for any plaintext. This attack can be prevented using standard message integrity techniques, such as message authentication codes. We leave this issue for our future study.

5 Conclusion

We presented the first proof of plaintext knowledge and the first verifiable encryption for an equality relation for the McEliece PKE. Our constructions are proved secure in the standard model, under hardness of the McEliece assumptions related to coding theory. An important open question is to upgrade our scheme to non-malleable security. According to [23], this will allow us to construct password-based authentication and key exchange, as well as deniable authentication based on coding. Another important open question is to extend our verifiable encryption to more general relations and to verifiable decryption. This would, for instance, yield code-based constructions for key escrow and optimistic fair exchange, according to [9].

Acknowledgments. The authors would like to thank anonymous reviewers of ACISP 2012 for their helpful comments.

The first author was supported by a *kakenhi* Grant-in-Aid for Young Scientists (B) 24700013 from Japan Society for the Promotion of Science.

References

1. Aumann, Y., Rabin, M.O.: A Proof of Plaintext Knowledge Protocol and Applications. Manuscript (June 2001), Available as slides from 1998 IACR Distinguished Lecture by M.O. Rabin, `http://www.iacr.org/publications/dl/rabin98/rabin98slides.ps`
2. Asokan, N., Shoup, V., Waidner, M.: Optimistic Fair Exchange of Digital Signatures (Extended Abstract). In: Nyberg, K. (ed.) EUROCRYPT 1998. LNCS, vol. 1403, pp. 591–606. Springer, Heidelberg (1998)
3. Bendlin, R., Damgård, I.: Threshold Decryption and Zero-Knowledge Proofs for Lattice-Based Cryptosystems. In: Micciancio, D. (ed.) TCC 2010. LNCS, vol. 5978, pp. 201–218. Springer, Heidelberg (2010)
4. Bernstein, D.J., Lange, T., Peters, C.: Wild McEliece. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) SAC 2010. LNCS, vol. 6544, pp. 143–158. Springer, Heidelberg (2011)
5. Berlekamp, E., McEliece, R., van Tilborg, H.: On the inherent intractability of certain coding problems. IEEE Trans. on Inf. Theory 24, 384–386 (1978)
6. Bernstein, D.J.: Grover vs. McEliece. In: Sendrier, N. (ed.) PQCrypto 2010. LNCS, vol. 6061, pp. 73–80. Springer, Heidelberg (2010)
7. Bernstein, D.J., Lange, T., Peters, C.: Smaller Decoding Exponents: Ball-Collision Decoding. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 743–760. Springer, Heidelberg (2011)

8. Camenisch, J., Damgård, I.: Verifiable Encryption, Group Encryption, and Their Applications to Separable Group Signatures and Signature Sharing Schemes. In: Okamoto, T. (ed.) ASIACRYPT 2000. LNCS, vol. 1976, pp. 331–345. Springer, Heidelberg (2000)
9. Camenisch, J.L., Shoup, V.: Practical Verifiable Encryption and Decryption of Discrete Logarithms. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 126–144. Springer, Heidelberg (2003)
10. Cayrel, P.-L., Véron, P., El Yousfi Alaoui, S.M.: A Zero-Knowledge Identification Scheme Based on the q-ary Syndrome Decoding Problem. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) SAC 2010. LNCS, vol. 6544, pp. 171–186. Springer, Heidelberg (2011)
11. Dowsley, R., van de Graaf, J., Müller-Quade, J., Nascimento, A.C.A.: Oblivious Transfer Based on the McEliece Assumptions. In: Safavi-Naini, R. (ed.) ICITS 2008. LNCS, vol. 5155, pp. 107–117. Springer, Heidelberg (2008)
12. Engelbert, D., Overbeck, R., Schmidt, A.: A Summary of McEliece-Type Cryptosystems and their Security. Journal of Mathematical Cryptology 1, 151–199 (2007), Walter de Gruyter
13. Finiasz, M., Sendrier, N.: Security Bounds for the Design of Code-Based Cryptosystems. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 88–105. Springer, Heidelberg (2009)
14. Fischer, J.-B., Stern, J.: An Efficient Pseudo-random Generator Provably as Secure as Syndrome Decoding. In: Maurer, U.M. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 245–255. Springer, Heidelberg (1996)
15. Galil, Z., Haber, S., Yung, M.: Symmetric Public-Key Encryption. In: Williams, H.C. (ed.) CRYPTO 1985. LNCS, vol. 218, pp. 128–137. Springer, Heidelberg (1986)
16. Goldreich, O.: Foundations of Cryptography I: Basic Tools. Cambridge University Press (2001)
17. Goldreich, O.: Foundations of Cryptography II: Basic Applications. Cambridge University Press (2004)
18. Goldreich, O., Micali, S., Wigderson, A.: Proofs that Yield Nothing But Their Validity for All Languages in NP Have Zero-Knowledge Proof Systems. J. ACM 38(3), 691–729 (1991)
19. Goldwasser, S., Kharchenko, D.: Proof of Plaintext Knowledge for the Ajtai-Dwork Cryptosystem. In: Kilian, J. (ed.) TCC 2005. LNCS, vol. 3378, pp. 529–555. Springer, Heidelberg (2005)
20. Goppa, V.D.: A new class of linear error-correcting code. Probl. Peredach. Inform. 6, 24–30 (1970) (in Russian)
21. Ishai, Y., Kushilevitz, E., Ostrovsky, R., Sahai, A.: Zero-knowledge from secure multiparty computation. In: STOC, pp. 21–30 (2007)
22. Janwa, H., Moreno, O.: McEliece Public Key Cryptosystems Using Algebraic-Geometric Codes. Des. Codes Cryptography 8(3), 293–307 (1996)
23. Katz, J.: Efficient and Non-Malleable Proofs of Plaintext Knowledge and Applications. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 211–228. Springer, Heidelberg (2003)
24. Kobara, K., Morozov, K., Overbeck, R.: Coding-Based Oblivious Transfer. In: MMICS, pp. 142–156 (2008)
25. MacWilliams, F.J., Sloane, N.J.A.: The Theory of Error-Correcting Codes. North-Holland, Amsterdam (1992)
26. McEliece, R.J.: A Public-Key Cryptosystem Based on Algebraic Coding Theory. Deep Space Network Progress Rep. (1978)

27. Naor, M.: Bit Commitment Using Pseudo-Randomness (Extended Abstract). In: Brassard, G. (ed.) CRYPTO 1989. LNCS, vol. 435, pp. 128–136. Springer, Heidelberg (1990)
28. Niederreiter, H.: Knapsack-type Cryptosystems and Algebraic Coding Theory. Prob. of Control and Inf. Theory 15(2), 159–166 (1986)
29. Nojima, R., Imai, H., Kobara, K., Morozov, K.: Semantic security for the McEliece cryptosystem without random oracles. Des. Codes Cryptography 49(1-3), 289–305 (2008)
30. Roth, R.: Introduction to coding theory. Cambridge University Press (2006)
31. Sendrier, N.: On the security of the McEliece public-key cryptosystem. In: Information, Coding and Mathematics – Proceedings of Workshop honoring Prof. Bob McEliece on his 60th Birthday, pp. 141–163. Kluwer (2002)
32. Shor, P.W.: Polynominal Time Algorithms for Discrete Logarithms and Factoring on a Quantum Computer. In: Huang, M.-D.A., Adleman, L.M. (eds.) ANTS 1994. LNCS, vol. 877, p. 289. Springer, Heidelberg (1994)
33. Stadler, M.: Publicly Verifiable Secret Sharing. In: Maurer, U.M. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 190–199. Springer, Heidelberg (1996)
34. Stern, J.: A New Identification Scheme Based on Syndrome Decoding. In: Stinson, D.R. (ed.) CRYPTO 1993. LNCS, vol. 773, pp. 13–21. Springer, Heidelberg (1994)
35. Véron, P.: Improved identification schemes based on error-correcting codes. Appl. Algebra Eng. Commun. Comput. 8(1), 57–69 (1996)
36. Xagawa, K., Kawachi, A., Tanaka, K.: Proof of Plaintext Knowledge for the Regev Cryptosystems. Tech.rep. C-236, Tokyo Inst. of Technology (2007)
37. Xagawa, K., Tanaka, K.: Zero-Knowledge Protocols for NTRU: Application to Identification and Proof of Plaintext Knowledge. In: Pieprzyk, J., Zhang, F. (eds.) ProvSec 2009. LNCS, vol. 5848, pp. 198–213. Springer, Heidelberg (2009)