Hiding the carriers in the polynomial Naccache Stern knapsack cryptosystem

Giacomo Micheli, Joachim Rosenthal, Reto Schnyder

Hiding the carriers in the polynomial Naccache Stern knapsack cryptosystem

Giacomo Micheli, Joachim Rosenthal, and Reto Schnyder *

University of Zurich
giacomo.micheli@math.uzh.ch, joachim.rosenthal@math.uzh.ch,
reto.schnyder@math.uzh.ch

Abstract. In this paper we provide two variants of the Naccache-Stern knapsack scheme using the framework of [5] that are meant to hide the prime elements used for decryption.

1 Introduction

The Naccache Stern Knapsack cryptosystem (NSK) is a public key cryptosystem first presented in [6]. It is based on the subset product problem in a large prime field. However, as it has already been observed in [2, Subsection 5.2], in the NSK protocol a quite obvious DLP reduction is available. In Section 2 we recall the NSK and give few details on an attack that can be performed. In Section 3 we will produce a version of the polynomial knapsack presented in [5] that is menat to hide the carriers p_i for which one performs the attack. The main idea behind this new instance of the knapsack problem relies on the following two facts, which we present here in an informal way:

- In the isomorphism class of the prime field $\mathbb{F}_p$ (that contains the space of cyphertexts of the NSK) there is "somehow" just one standard representative, which consists of the integers modulo p.
- In the isomorphism class of the finite field $\mathbb{F}_{q^n}$ with $n > 1$ (that contains the space of cyphertexts of the polynomial knapsack cryptosystem described in [5]) there are many equally valid representatives: one for each irreducible polynomial of degree n over $\mathbb{F}_q$.

This difference will be exploited in what follows. Some security considerations are also provided in Section 5.1. Finally a more general function field version of the protocol is provided in Section 6.

1.1 Notation

Let q be a prime power and let $\mathbb{F}_q$ be the field with q elements. In this paper, the univariate polynomial ring in Z will be denoted by $\mathbb{F}_q[Z]$. Let now

* The authors were supported in part by Swiss National Science Foundation grant number 149716 and *Armasuisse*.

$g(Z) \in \mathbb{F}_q[Z]$ be an irreducible polynomial of degree d, and consider the quotient map $\pi\colon \mathbb{F}_q[Z] \longrightarrow \mathbb{F}_q[Z]/(g(Z))$. We will denote by z the element $\pi(Z)$. More generally, in the whole paper a transcendental element over $\mathbb{F}_q$ will always be denoted by a capital letter, whereas its image in some quotient field of the polynomial ring will be denoted by a lower case letter if it is clear which projection we are considering. We write an element of $\mathbb{F}_q[z] = \mathbb{F}_q[Z]/(g(Z))$ as $h(z)$, where $h(Z)$ is its unique lift to $\mathbb{F}_q[Z]$ of degree at most $d-1$. We mainly encounter the following situation in this paper: Let X, Y transcendental over $\mathbb{F}_q$ and $f(X) \in \mathbb{F}_q[X]$, $g(Y) \in \mathbb{F}_q[Y]$ irreducible polynomials over $\mathbb{F}_q$ of degree d. Then,

$$\mathbb{F}_{q^d} \cong \mathbb{F}_q[X]/(f(X)) = \mathbb{F}_q[x]$$

but also

$$\mathbb{F}_{q^d} \cong \mathbb{F}_q[Y]/(g(Y)) = \mathbb{F}_q[y].$$

2 Recalling the NSK protocol

We recall the Naccache-Stern knapsack cryptosystem [6]. Let p and $p_1, \ldots, p_n$ be primes such that $\prod_{i=1}^n p_i < p$. Often, the p_i are the first n primes, and p is the smallest prime greater than their product. The secret key is a random invertible integer modulo $p-1$: $s \in \mathbb{Z}_{p-1}^*$ as well as its inverse $t = s^{-1}$. The public key consists of $v_i = p_i^s \in \mathbb{F}_p$. We will often call these elements *carriers*. An n-bit message $m = (m_1, \ldots, m_n) \in \{0,1\}^n$ is now ecrypted as

$$c = \prod_{i=1}^n v_i^{m_i} \in \mathbb{F}_p.$$

In order to decrypt a ciphertext c, it is first raised to the power t:

$$c^t = \prod_{i=1}^n (p_i^{m_i})^{st} = \prod_{i=1}^n p_i^{m_i} \in \mathbb{F}_p.$$

Now, since $\prod_{i=1}^n p_i^{m_i} < p$, this can be lifted to the integers, where it is easy to check divisibility by the p_i.

Remark 1. The main parameter that has to be tuned in the NSK protocol is the information rate $n/\log_2(p)$ i.e. the ratio between the information contained in the message (number of bits of the message) divided by the modulus (number of bits of the cyphertext). For practical implementations, this ratio is about $1/10$ [5, Table 1].

Remark 2. It is clear how to attack NSK assuming we can solve a DLP in $\mathbb{F}_p$ as follows: If the p_i have been chosen as the first n primes (that is a greedy choice if we want to maximize the information rate), they are assumed to be known, and s can be recovered by solving $p_i^s = v_i$ for any i. Even if the p_i are chosen otherwise, the smallest among them has to be less than $p^{1/n}$. Since n is usually quite large (the original paper recommends $n \geq 160$ [6]), it can thus be guessed for reasonable parameters.

In what follows we will provide a a way to avoid this kind of attack.

3 Hidden field version

In what follows we will in fact show a strategy that hides the knowledge of the low degree polynomials keeping the information rate unchanged. The basic idea is to work in two representations of the same finite field, only one of which is public. We first observe that prime fields are not suited for this, since for each prime p we have exactly one representation of $\mathbb{F}_p$ for which the computations in the NSK are possible. The polynomial based variant presented in [5] on the other hand is very well suited for this, since each finite field $\mathbb{F}_{q^d}$ has roughly q^d/d representations of the form $\mathbb{F}_q[X]/(f(X))$ (see for example [4]).

Let $\mathbb{F}_q$ be a finite field and $p_1(Z), \ldots, p_L(Z)$, $f(Z)$, $g(Z) \in \mathbb{F}_q[Z]$ be irreducible polynomials such that $\sum_{i=1}^{L} \deg p_i(Z) < \deg f(Z) = \deg g(Z) =: d$. Fix furthermore a field isomorphism

$$\phi\colon \mathbb{F}_q[X]/(f(X)) \to \mathbb{F}_q[Y]/(g(Y)).$$

Fix now a random exponent $s \in \mathbb{Z}^*_{q^d-1}$ as well as its inverse $t = s^{-1}$ and let $v_i(y) = \phi(p_i(x))^s \in \mathbb{F}_q[y]$ for $i \in \{1, \ldots, n\}$. The public key consists of $(g(Y), \{v_i(Y)\}_i)$. The secret key consists of $(t, \{p_i(X)\}_i, f(X), \phi)$.

The encryption of an n-bit message $m = (m_1, \ldots, m_n) \in \{0,1\}^n$ is essentially the same as in Section 2:

$$c(y) = \prod_{i=1}^{n} v_i(y)^{m_i} \in \mathbb{F}_q[y] = \mathbb{F}_q[Y]/(g(Y)).$$

To decrypt, compute

$$m(x) := \phi^{-1}(c(y)^t) = \prod_{i=1}^{n} p_i(x)^{m_i} \in \mathbb{F}_q[x] = \mathbb{F}_q[X]/(f(X)),$$

and canonically lift the result to $\mathbb{F}_q[X]$. For all i, decrypt as

$$m_i = 1 \Leftrightarrow m(X) \equiv 0 \mod p_i(X).$$

Remark 3. Here we would like to point out the advantage of this version of the polynomial based protocol compared to the NSK: The information rate of the NSK protocol can be tuned by minimizing the degree of the modulus p, which can be done by minimizing the degree of the carriers p_i. Unfortunately, whenever this is done, a DLP reduction becomes easier, as we already pointed out in Remark 2. More generally, in the NSK and the polynomial based variant of [5] it is possible to balance *information rate* and *security*. In what we presented, the information rate can always be tuned to the top, since the carriers will be hidden by the (arbitrary) choice of the finite field representation. In Section 5 a deeper analysis of the security is provided.

Remark 4. Notice that breaking this variant of the polynomial based protocol is *at least as difficult* as breaking the original one: just fix $f = g$ and ϕ the identity morphism. Moreover

– the encryption is exactly as expensive as in the NSK and the polynomial based variant (under the formalism presented in [5, Section 2] it is actually the same).
– the decryption differs from the NSK for just one additional step: the computation of ϕ^{-1} of the cyphertext.

3.1 Example

In order to immediately clarify the ideas, we give a small example over $\mathbb{F}_2$. All calculations were done using the Sage computer algebra system [7]. We choose $d = 11$ and the public polynomial

$$g(Y) = Y^{11} + Y^{10} + Y^8 + Y^7 + Y^6 + Y^3 + 1.$$

Further, we pick a random element of $\mathbb{F}_2[y] = \mathbb{F}_2[Y]/(g(Y))$ that is not contained in a proper subfield:

$$a(y) = y^{10} + y^9 + y^4 + y$$

and compute its minimal polynomial in X i.e. $f(X) \in \mathbb{F}_q[X]$:

$$f(X) = X^{11} + X^{10} + X^9 + X^8 + X^7 + X^6 + X^5 + X^4 + X^3 + X + 1.$$

The isomorphism ϕ is now given by

$$\begin{aligned} \phi\colon \mathbb{F}_2[x] &\to \mathbb{F}_2[y] \\ x &\mapsto a(y). \end{aligned}$$

To compute the inverse of ϕ, we just need to find the preimage of y. Since this preimage is a root of $g(X)$, we can find it by factoring $g(X)$ over the field $\mathbb{F}_{2^{11}}$ in the representation $\mathbb{F}_q[x]$ and checking for each root $r(x)$ whether $\phi(r(x)) = y$.

$$b(x) = \phi^{-1}(y) = x^{10} + x^3 + 1.$$

We now choose $p_1(X), \ldots, p_n(X)$ to be the five irreducible polynomials of lowest degree in $\mathbb{F}_2[X]$:

$$\begin{aligned} p_1(X) &= X & p_2(X) &= X + 1 \\ p_3(X) &= X^2 + X + 1 & p_4(X) &= X^3 + X + 1 \\ p_5(X) &= X^3 + X^2 + 1, \end{aligned}$$

and randomly select the secret exponents $s = 1967 \in \mathbb{Z}^*_{2^d-1}$ and $t = s^{-1} = 1612$.

The public key consists of the elements $v_i(y) = \phi(p_i(x))^s \in \mathbb{F}_2[Y]/(g(Y))$:

$$\begin{aligned} v_1(y) &= y^9 + y^6 + y^5 + y \\ v_2(y) &= y^8 + y^7 + y^6 + y^5 + y^4 + y^2 \\ v_3(y) &= y^8 + y^4 + y^3 + 1 \\ v_4(y) &= y^9 + y^6 + y^3 + y + 1 \\ v_5(y) &= y^{10} + y^9 + y^8. \end{aligned}$$

We now encrypt the message $m = (1, 1, 0, 0, 1)$ and get the ciphertext $c = v_1(y)v_2(y)v_5(y) = y^9 + y^4 + y^2 + y$. To decrypt, compute $\phi^{-1}(c(y)^t) = x^5 + x^3 + x^2 + x$, which, when lifted to $\mathbb{F}_2[X]$, factors as

$$X^5 + X^3 + X^2 + X = X \cdot (X + 1) \cdot (X^3 + X^2 + 1) = p_1(X)p_2(X)p_5(X),$$

from which the message m is recovered.

4 Efficiency analysis

In this section, we compare the efficiency and performance of our new protocol to the traditional NSK [6] and the unmodified polynomial based variant [5], as well as other public key cryptosystems.

The size of the public key is unchanged compared to [5], being roughly $(n + 1)d \log q$ bits. The secret key contains an additional polynomial of degree d and one field element of $\mathbb{F}_q[x]$, so it is roughly twice as large, at around $4d \log q$ bits.

Encryption consists of up to $n - 1$ multiplications in $\mathbb{F}_{q^d}$. Decryption on the other hand requires raising an element of $\mathbb{F}_{q^d}$ to the exponent $t \leq q^d - 1$, which takes up to $2d \log q$ multiplications with the square-and-multiply method. Furthermore, the evaluation of the inverse field isomorphism ϕ^{-1} consists of evaluating a polynomial in $\mathbb{F}_q[Z]$ of degree less than d at an element of $\mathbb{F}_{q^d}$, taking at most d multiplications in $\mathbb{F}_{q^d}$ by Horner's method. Finally, decryption involves the reduction of a polynomial in $\mathbb{F}_q[Z]$ of degree less than d by n fixed distinct polynomials of small degree, which is cheap compared to the first two steps even when done naively. Compared to [5], the only difference in cost is the evaluation of the isomorphism, so the cost of decryption only increases by a factor of roughly $1 + (2 \log q)^{-1}$.

For the multiplication of polynomials and elements of $\mathbb{F}_{q^d}$, it may be worthwhile to use algorithms based on the fast Fourier transform [1], since our polynomials have large degree.

We now try to find concrete secure parameters. It is recommended in [6] to use at least $n \geq 160$ carriers in order to avoid birthday attacks. Following this recommendation, we can for example choose the parameters $q = 19$ and $d = 307$, which allows for $n = 162$ carriers. The reason we do not choose $q = 2$ or another very small prime is that this causes the polynomials f and g to have very large degree, making it harder to compute the preimage $\phi^{-1}(y)$. With these parameters, we get a public key size of roughly 212 kbit. Using different q and d that achieve a similar number of carriers does not seem to change the key size dramatically. Indeed, it appears that the value $d \log q$ is similar in size to $n \log n$ whenever n is chosen maximally.

Regarding information rate, we again get similar values to [5, 6]. In particular, the above parameters give an information rate of 11.8%. There are various tricks to improve on this at the cost of key size, see for example [6, 2]. We will however not discuss these in this paper.

5 Security analysis

In this section we will analyse the difficulty of recovering either a field representation in which a certain set of elements has small degree or attacking the relations between the carriers v_i.

5.1 Recovering the hidden field

Given a finite field $\mathbb{F}_q$ and $d > 1$, we consider the question of finding a representation of $\mathbb{F}_{q^d}$ in which a set of field elements have low degree. By a representation of $\mathbb{F}_{q^d}$, we mean a pair $(f(X), \phi)$ with $f(X) \in \mathbb{F}_q[X]$ irreducible of degree d and $\phi\colon \mathbb{F}_{q^d} \to \mathbb{F}_q[X]/(f(X))$ a field isomorphism fixing $\mathbb{F}_q$.

Proposition 1. *Let $a \in \mathbb{F}_{q^d}$ and $2 \leq m \leq d$. Then, there exist at most $(m-1)q^m$ representations $(f(X), \phi)$ of $\mathbb{F}_{q^d}$ such that $\phi(a)$, canonically lifted to $\mathbb{F}_q[X]$, has degree less than m. All of these pairs can be listed in time $\tilde{\mathcal{O}}(q^m)$.*

Proof. Let $(f(X), \phi)$ be such a representation. We have that $\phi(a) = h(x)$, where $x \in \mathbb{F}_q[X]/(f(X))$ is the equivalence class of X and $h(X) \in \mathbb{F}_q[X]$ has degree less than m. By inverting the isomorphism, we get that $h(\phi^{-1}(x)) - a = 0$, so $\phi^{-1}(x)$ is a root of $h(X) - a$ in $\mathbb{F}_{q^d}$.

$h(X) - a$ has at most $m - 1$ roots in $\mathbb{F}_{q^d}$. Each root ρ, if not contained in a proper subfield of $\mathbb{F}_{q^d}$, uniquely determines a representation $(f(X), \phi)$. Indeed, $f(X)$ is the minimal polynomial of ρ, and $\phi\colon \mathbb{F}_{q^d} \to \mathbb{F}_q[X]/(f(X))$ is determined by $\phi(\rho) = x$.

Hence, for a fixed polynomial $h(X)$ of degree less than m, there are at most $m - 1$ representations such that $\phi(a) = h(x)$. Since there are only q^m choices for $h(X)$, there are at most $(m - 1)q^m$ representations with $\phi(a)$ having degree less than m. They can be listed by simply enumerating all possible $h(X)$ and the roots of $h(X) - a$ in $\mathbb{F}_{q^d}$.

We now consider the question of finding a representation in which a set of elements $a_1, \ldots, a_n$ have a low degree simultaneously. Clearly, if such a representation exists, it can again be found in time $\tilde{\mathcal{O}}(q^m)$ by listing all representations in which a_1 has low degree. However, we show that such a representation is unlikely to exist for most parameters.

Corollary 1. *Let $2 \leq m \leq d$, and let $a_1, \ldots, a_n \in \mathbb{F}_{q^d}$ be chosen independently and uniformly at random. Then, a representation $(f(X), \phi)$ with $\phi(a_i)$ having degree less than m for all i exists with probability less than $(m - 1)q^{d+n(m-d)}$.*

Proof. Fix a representation $(f(X), \phi)$ such that $\phi(a_1)$ has degree less than m. Since the a_i are independent, we have that $\phi(a_i) \in \mathbb{F}_q[X]/(f(X))$ are still uniformly random for $i \in \{2, \ldots, n\}$. Hence, the probability of all of them having degree less than m is $\left(\frac{q^m}{q^d}\right)^{n-1} = q^{(n-1)(m-d)}$. Since there are at most $(m-1)q^m$ choices for the representation, the total probability of a desired representation existing is less than $(m - 1)q^{m+(n-1)(m-d)} = (m - 1)q^{d+n(m-d)}$.

For our cryptosystem, we conclude that the exponents s are essential for security. If the public parameters are simply chosen as $v_i(y) = \phi(p_i(x))$, it is easy to recover a field representation that allows decryption. On the other hand, assuming the values $(\phi(p_i(x))^e)_{i\in\{1,\ldots,n\}}$ for a random large e are sufficiently close to uniformly random elements, it is not possible to find a representation that allows decryption without finding the secret exponent s.

5.2 Attacking the relations between the v_i

It is easy to see that in our scheme, unlike in the original NSK, there is no way to recover the secret exponents s and t from a single public key element $v_i(y) = \phi(p_i(x))^s$. Indeed, given a fixed $v_i(y) \in \mathbb{F}_q[y]$, there exists for each exponent t and low degree polynomial $p_i(X)$ a representation $(f(X), \phi)$ such that $v_i(y)^t = \phi(p_i(x)) \in \mathbb{F}[y]$, assuming that $p_i(X) - v_i(y)^t$ has a root in $\mathbb{F}_q[y]$ that is not in any proper subfield.

However, the public key consists of many elements $v_i(y) = \phi(p_i(x))^s$, and they are not independent of each other. For example, consider the case $p_1(X) = X$, $p_2(X) = X + 1$. The attacker knows that $v_1(y)^t + 1 = v_2(y)^t$ for some unknown t. Hence, the attacker has a criterion to check whether a guess for the secret exponents s and t is correct.

Since our goal is to repair the structural weakness of the polynomials NSK against attacks on the small factors p_i, let us assume for a moment that the attacker has a way to solve the DLP in the finite field $\mathbb{F}_q[y]$. For example, assume the attacker knows that $v_2(y) = v_1(y)^a$ for some a. The above relation can be rewritten as $(v_1(y)^t)^a - (v_1(y)^t) + 1 = 0$, so t can be recovered by finding roots of $X^a - X + 1 = 0$ in $\mathbb{F}_q[y]$ and solving another DLP. However, the degree of the polynomial equation is a, which in general is exponential, so standard methods for finding roots are not applicable.

To reduce the degree of the equation, the attacker can hope to find an integer $e \in [1, \alpha q^{d/2}]$ such that $(ea \bmod q^d - 1) \leq \alpha q^{d/2}$, where $\alpha > 0$ is some parameter. This can often be found by brute force. In that case, the solutions to the equation $X^{ae \bmod q^d-1} - X^e + 1 = 0$ correspond to the solutions of $X^a - X + 1 = 0$, so it is enough to solve a polynomial equation of degree at most $\alpha q^{d/2}$. However, the attack still runs in exponential time and is infeasible for practical parameters.

5.3 An attack using both a DLP-oracle and an SVP-oracle

We thank Cristophe Petit for pointing out that, having access to an SVP- and a DLP-oracle, a simple way to attack the problem is the following: Solve the discrete logarithms $v_1^{x_i} = v_i$ and $v_1^l = c$. Then try to solve the additive knapsack $m_1 + \sum_{i=2}^{L} m_i x_i \equiv l \mod q^d - 1$. This can be done using the algorithm in [3] for solving any of the subset sum problems $m_1 + \sum_{i=2}^{L} m_i x_i = l + k(q^d - 1)$ for $k \in \{1, \ldots, L\}$. It is a matter of further investigation whether this attack is practical for reasonable parameters.

The following is an attempt to prevent this kind of reduction: Restrict the space of messages to strings of constant Hamming weight T. The condition on g

then becomes $\deg(g) > \sum_{i=L-T+1}^{L} \deg(p_i)$. Moreover, T, g and L must be chosen in such a way that $q^{\deg(g)} << 2^L$. We now expect there to be many solutions to the subset sum problem, of which only one leads to the plaintext. Further analysis would be needed to establish whether this trick effectively prevents the attack.

5.4 Challenge

We provide the following instance as a challenge. Again, all calculations were done using Sage [7]. Let $q = 19$, $d = 307$, $L = 162$ and

$$g = \texttt{72CI64EI6I6D9DGA81B2EC0GH5074I42C0F3I9B7GA22}$$
$$\texttt{2874FGFF2AI5CA68B20909IH60I7DH42HD8BH3964E20}$$
$$\texttt{H4DH39128C2I2I9FHIE4371D8HBBC4A94997BHAF6F86}$$
$$\texttt{GEF9BDGH4CA7H69GC21GFGEG19GH68E8CH74HAE619IB}$$
$$\texttt{6C35F5D7A1CFE0DD61724BD8C29F235F6C65CAB2G718}$$
$$\texttt{2DD9F2I22I5GCII03E0AI9399C43IC4BC453A65B3CEE}$$
$$\texttt{497BAC44290915DF537D0F3468D49F630C037DC06D71}.$$

The polynomial is encoded as follows: The first character is the coefficient of the constant term, the last is the coefficient of x^{307}. The elements 0 through 18 of $\mathbb{F}_{19}$ are represented by the characters `0` through `9` and `A` through `I`. The values of the carriers v_i are available upon request.

The following ciphertext encrypts a random message:

$$c = \texttt{97ADFGDEEH5FI3C0700E6EF4F7117CCIEG2FAFC88123}$$
$$\texttt{5HB7H754G406BC40D278FGEA2BA80E442DBE8F96G1E8}$$
$$\texttt{B2610AF99B5386DBH921A665I581F960II9D022IH0H3}$$
$$\texttt{A02I2E1B2HHHBDG16CID71A6GD62A7B090CGE605H1BB}$$
$$\texttt{A5802I9279GI1CD317BGH1H7IB8857BC71848A6ABC99}$$
$$\texttt{21G0CE57BG3C666B4II100IGC60G2IDCA79EG857A6FF}$$
$$\texttt{0GE0E1F0HC9H53I4D2H67D949G4GHG2G0HA8CF5820H0}.$$

The corresponding message should be recovered as an element of $\{0,1\}^{162}$.

6 Function Field Knapsack Scheme

In this section we show a more theoretical version of the scheme that uses the context of function fields of curves. For the notation and definitions we refer to [8]. Let F be a function field over a finite field $\mathbb{F}_q$. We denote by $\mathcal{O}_P$ the valuation ring associated to the place P. Moreover, if $f \in \mathcal{O}_P$, then we denote by $f(P)$ its image in the residue field $\mathcal{O}_P/P$. Let $\{P_1, \ldots, P_L\}$ distinct places of F. Let x_i be a uniformizer of the place P_i invertible in the valuation ring $\mathcal{O}_{P_j}$ for $j \neq i$ (this can be done thanks to the *Approximation Theorem* for valuations).

Fix P a place of the holomorphy ring $\mathcal{O}$ containing the subring generated by $x_1, \ldots x_L$. Choose P satisfying

$$\deg(P) > \sum_{i=1}^{L} \deg((x_i)_\infty)$$

and let $D = \sum_{i=1}^{L} (x_i)_\infty$ Let $l(D) := \dim_{\mathbb{F}_q}(\mathcal{L}(D))$.

Remark 5. Notice that, for any $(m_1, \ldots, m_L) \in \{0,1\}^L$, we have

$$\prod_{i=1}^{L} x_i^{m_i} \in \mathcal{L}(D).$$

Now observe that $\mathcal{L}(D)$ evaluates at P into the finite field $\mathbb{F}_{q^{\deg(P)}}$. In addition $\mathcal{L}(D)$ embeds into $\mathbb{F}_{q^{\deg(P)}}$ since an element in the kernel of the evaluation would live in $\mathcal{L}(D-P)$ but $D-P$ is a divisor of negative degree, so $\mathcal{L}(D-P) = 0$.

Remark 6. Call ψ the embedding of $\mathcal{L}(D)$ into $\mathbb{F}_{q^{\deg(P)}}$. Observe that, given an $\mathbb{F}_q$-basis $\{e_1, \ldots, e_{l(D)}\}$ of $\mathcal{L}(D)$, $\psi^{-1}(c)$ is computable for any c in the image of ψ. In fact it is enough to write down $c = \sum_{i=1}^{L} a_i \psi(e_i)$ for some $a_i \in \mathbb{F}_q$; then $\psi^{-1}(c) = \sum_i^L a_i e_i$.

Let $\mathcal{M} = \{0,1\}^L$ be the space of messages and $\mathcal{C} := \mathbb{F}_{q^{\deg(P)}}$ be the space of cyphertexts. Let $e, d \in \mathbb{Z}$ for which $ed \equiv 1 \mod q^{\deg(P)} - 1$. Now we are able to set up public and private key:

- **Public key:** $(\mathbb{F}_{q^{\deg(P)}}, \{x_1(P)^e, \ldots, x_L(P)^e\})$
- **Private key:** $(F, d, \{x_1, \ldots, x_L\}, \psi)$

Encryption and decryption are defined as

- **Encryption:** let $m = (m_1, \ldots, m_L) \in \mathcal{M}$. The encryption map is defined by $E(m) := \prod_{i=1}^{L} (x_i(P)^e)^{m_i}$.
- **Decryption:** Let $c \in \mathcal{C}$. Compute c^d and invert ψ, getting $\overline{c} = \prod_{i=1}^{L} x_i^{m_i}$.

Remark 7. The reader should notice that in this case hiding the residue field is not necessary, since the attack does not have available the function field used for decryption, but only the residue field.

Acknowledgements

The authors were supported in part by Swiss National Science Foundation grant number 149716 and *Armasuisse*.

Bibliography

[1] David G. Cantor and Erich Kaltofen. On fast multiplication of polynomials over arbitrary algebras. *Acta Informatica*, 28(7):693–701, 1991.

[2] Benot Chevallier-Mames, David Naccache, and Jacques Stern. Linear bandwidth naccache-stern encryption. In *Security and Cryptography for Networks*, volume 5229 of *Lecture Notes in Computer Science*, pages 327–339. Springer Berlin Heidelberg, 2008. doi: 10.1007/978-3-540-85855-3_22.

[3] MatthijsJ. Coster, Antoine Joux, BrianA. LaMacchia, AndrewM. Odlyzko, Claus-Peter Schnorr, and Jacques Stern. Improved low-density subset sum algorithms. *computational complexity*, 2(2):111–128, 1992. ISSN 1016-3328. doi: 10.1007/BF01201999. URL `http://dx.doi.org/10.1007/BF01201999`.

[4] Rudolf Lidl and Harald Niederreiter. *Finite fields*, volume 20. Cambridge University Press, 1997.

[5] Giacomo Micheli and Michele Schiavina. A general construction for monoid-based knapsack protocols. *Advances in Mathematics of Communications*, 8 (3), 2014.

[6] David Naccache and Jacques Stern. A new public key cryptosystem. In *Advances in Cryptology*, pages 27–36. EUROCRYPT, 1997.

[7] W. A. Stein et al. *Sage Mathematics Software (Version 6.1.1)*. The Sage Development Team, 2014. URL `http://www.sagemath.org`.

[8] Henning Stichtenoth. *Algebraic function fields and codes*, volume 254. Springer, 2009.