Chapter 11
An Information Rate Improvement for a Polynomial Variant of the Naccache-Stern Knapsack Cryptosystem

Giacomo Micheli, Joachim Rosenthal and Reto Schnyder

Abstract We adapt an information rate improvement by Chevallier-Naccache-Stern for the Naccache-Stern knapsack cryptosystem, called the prime packing strategy, to the polynomial version of the protocol.

11.1 Introduction

In 1997 Naccache and Stern [4] proposed a new public key cryptosystem known as the *Naccache-Stern Knapsack cryptosystem*, or *NSK* for short. This system was based on modular arithmetic in the integers and had a number theoretic flavor. However, NSK suffers from a low information rate: The ratio of message to ciphertext size is less than 10 % for many practical parameters. More recently in 2008, Chevallier-Mames, Naccache and Stern [2] presented several alterations to the protocol that improve the information rate at the cost of a larger public key size.

More than a decade after the NSK protocol was invented, Micheli and Schiavina presented a generalized monoid based version of the NSK Protocol [3], as well as an instance based on polynomials over finite fields. This variant suffers from the same low information rate. In this chapter, we apply the improvements of [2] to this polynomial based variant.

G. Micheli (✉) · J. Rosenthal · R. Schnyder
Institute of Mathematics, University of Zurich,
Winterthurerstrasse 190, 8057 Zurich, Switzerland
e-mail: giacomo.micheli@math.uzh.ch

J. Rosenthal
e-mail: rosenthal@math.uzh.ch

R. Schnyder
e-mail: reto.schnyder@math.uzh.ch

M. Baldi and S. Tomasin (eds.), *Physical and Data-Link Security Techniques for Future Communication Systems*, Lecture Notes in Electrical Engineering 358, DOI 10.1007/978-3-319-23609-4_11

11.2 Recalling the NSK Protocol

We recall here the NSK protocol and its generalization. They are both based on the following problem:

Problem 11.1 Let L be a positive integer, M be a monoid and $c, v_1, \ldots, v_L$ elements of M. Find (if one exists) a vector $m = (m_1, \ldots m_L) \in \{0, 1\}^L$ for which

$$c = \prod_{i=1}^{L} v_i^{m_i}.$$

In what follows, we show some instances of the problem above and the cryptographic protocol arising from them. Let $\mathbb{F}_q$ be the finite field of order q.

Problem 11.2 Fix a positive integer L, the monoid $M = (\mathbb{F}_q[x], \cdot)$, irreducible polynomials $p_1, \ldots, p_L \in M$ and

$$c = \prod_{i=1}^{L} p_i^{m_i}.$$

for some $(m_1, \ldots m_L) \in \{0, 1\}^L$. Find the vector m.

It is immediate that Problem 11.2 can be easily solved by reducing c modulo p_i for each i: we have in fact $m_i = 1$ if and only if $c \equiv 0 \mod p_i$.

Problem 11.3 Let g be an irreducible polynomial of degree N, L a positive integer and $M = (\mathbb{F}_q[x]/(g(x)), \cdot) \cong (\mathbb{F}_{q^N}, \cdot)$. Let $v_1, \ldots, v_L \in M$ and

$$c = \prod_{i=1}^{L} v_i^{m_i}.$$

for some $(m_1, \ldots m_L) \in \{0, 1\}^L$. Find the vector m.

The generic instance of Problem 11.3 is now difficult compared to Problem 11.2. This gap is exploited in [3]. In what follows we recall their protocol, which we will refer to as the *polynomial NSK* or *pNSK* for short.

Alice sets up the system as follows:

- Alice chooses a finite field $\mathbb{F}_q$, L irreducible polynomials $p_i \in \mathbb{F}_q[x]$, an irreducible polynomial g for which $\sum_{i=1}^{L} \deg p_i < \deg g$ and a pair of integers (e, s) for which $es \equiv 1 \mod q^N - 1$.
- The private key is $(p_1, \ldots, p_L, s)$.
- The public key is $(v_1, \ldots, v_L, \mathbb{F}_q[x]/(g(x)))$, where $v_i = p_i^e$.

The encryption of a message $m \in \{0, 1\}^L$ is performed as

$$m \mapsto \prod_i v_i^{m_i} = c \in \mathbb{F}_q[x]/(g(x)).$$

Alice can then decrypt by computing $c^s \in \mathbb{F}_q[x]/(g(x))$ and reducing the result modulo p_i for each i, since $c^s \mod g(x)$ (together with its factorization in terms of the p_i) suitably lifts to $\mathbb{F}_q[x]$ using the property $\sum_{i=1}^{L} \deg p_i < \deg g$.

The original NSK is obtained by replacing $\mathbb{F}_q[x]$ by $\mathbb{Z}$ and irreducible polynomials by prime numbers.

11.3 Prime Packing

In what follows our goal is to show that a direct adaptation of the NSK packing presented in [2] is also possible in the case of the polynomial variant. We pack the irreducible polynomials up to degree d as follows: Let $b, t \in \mathbb{N}$ be positive integers for which $bt \leq \overline{\pi}(d)$, where $\overline{\pi}(d)$ is the number of irreducible polynomials up to degree d. Partition the first (according to any ordering respecting the degree) bt polynomials in t sets $\{S_i\}$ each of size b satisfying that for all $i, j \in \{1, \ldots, t\}$, if $f \in S_i$ and $h \in S_j$ we have

$$i \leq j \Rightarrow \deg(f) \leq \deg(h).$$

More informally, we pack the polynomials up to degree d into t packs, each of them containing the b polynomials of the lowest possible degree. Let us denote by $p_{j,i}$ the ith polynomial living in the jth box S_j, again ordered by degree. In particular, we have $\deg p_{j,i} \leq \deg p_{j,b}$ for all i and j. The protocol will then be modified as follows. The space of messages becomes $\{1, \ldots, b\}^t$, we require now only $\sum_{j=1}^{t} \deg p_{j,b} < \deg g = N$. Again, let $es \equiv 1 \mod q^N - 1$.

The public key is set up as $\big(\{v_{j,i}\}_{i,j}, \mathbb{F}_q[x]/(g(x))\big)$, where again $v_{j,i} = p_{j,i}^e$. The secret key is analogously $\big(\{p_{j,i}\}_{i,j}, s\big)$. The encryption of a message $m = (m_1, \ldots, m_t) \in \{1, \ldots, b\}^t$ is performed as

$$m \mapsto \prod_{j=1}^{t} v_{j,m_j} = c \in \mathbb{F}_q[x]/(g(x)).$$

Alice can then decrypt by computing $c^s \in \mathbb{F}_q[x]/(g(x))$ and reducing the result modulo $p_{j,i}$ for each i, j, as before.

It is now easy to compute the information rate and public key size: The information rate is $\frac{t \log b}{N \log q}$, and the public key has size $btN \log q$.

Table 11.1 Information rate and public key size of prime packing for $q = 6287$, $\deg g = 131$ and various box sizes

b	t	Information rate (%)	Public key size (kbit)
pNSK	130	7.9	215
5	130	18.3	1074
10	130	26.1	2149
30	130	38.6	6447
50	127	43.4	10496
70	109	40.4	12612

11.3.1 Example Parameters

As an example, consider the medium prime case $q = 6287$. We compare the information rate and public key size of our scheme in the case $\deg g = 131$ for various values of the box size b in Table 11.1. Computations were done using Sage [6]. The first row corresponds to the original pNSK (which is not quite the same as setting $b = 1$). Note that for small box sizes b, we always get $t = 130$ boxes. This is because it is possible to use only degree 1 polynomials for the $p_{j,i}$. As b becomes larger, this is no longer possible, and the information rate suffers.

Evidently, the information rate can be greatly improved at the cost of a much larger public key size. This cost can be somewhat reduced by applying the "powers of primes" technique of [2], and we will do so in Sect. 11.4.

11.3.2 Asymptotic Information Rate

As in [2], we can obtain linear bandwidth by setting the number of packs equal to their size. Indeed, we show that if we set $n := b = t$, then the information rate of pNSK using prime packing is asymptotically equal to $\frac{1}{2}$.

To analyze the information rate, we first need to find the degree of the nth irreducible polynomial p_n, according to any order respecting the degree. In [3, Sect. 3.2.2], it was shown that the number of irreducible polynomials in $\mathbb{F}_q[x]$ of degree at most d is asymptotically equal to $\frac{q}{q-1}\frac{q^d}{d}$. Hence, the polynomials with a given degree d should be numbered roughly between $\frac{q}{q-1}\frac{q^{d-1}}{d-1}$ and $\frac{q}{q-1}\frac{q^d}{d}$. Thus, if the polynomial p_n has degree d_n, we have

$$\frac{q}{q-1}\frac{q^{d_n-1}}{d_n-1} \lesssim n \lesssim \frac{q}{q-1}\frac{q^{d_n}}{d_n},$$

where $a_n \lesssim b_n$ means that $\limsup_{n\to\infty} a_n/b_n \leq 1$. Taking logarithms gives

$$(d_n - 1) - \log_q(d_n - 1) \lesssim \log_q n - \log_q \frac{q-1}{q} \lesssim d_n - \log_q d_n,$$

which asymptotically is the same as

$$d_n - 1 \lesssim \log_q n \lesssim d_n.$$

We hence see that $d_n = \deg p_n \sim \log_q n$.

Now we can approximate the degree of g:

$$\begin{aligned} N = \deg g &= 1 + \sum_{i=1}^{n} \deg p_{in} \\ &\sim \sum_{i=1}^{n} \log_q(in) \sim \sum_{i=1}^{n} \log_q(n^2) \sim 2n \log_q n. \end{aligned}$$

For the first $\sim$, note that the indices of p_{in} in the sum are all at least n, and so only the asymptotic behavior of $\deg p_{in}$ is relevant. Finally, we get for the information rate

$$\frac{t \log_2 b}{N \log_2 q} \sim \frac{n \log_2 n}{2n \log_q n \log_2 q} = \frac{n \log_2 n}{2n \log_2 n} = \frac{1}{2}.$$

11.4 Powers of Primes

In [2, Sect. 4], prime packing was applied to a variant of NSK using a base larger than 2 in order to further improve information rate and reduce public key size. This method can also be applied to the polynomial NSK variant.

As in Sect. 11.3, we again choose a degree d and integers b and t satisfying $bt \leq \overline{\pi}(d)$, and we partition the first bt irreducible polynomials into t sets S_i of size b. We further choose an integer parameter $\ell \geq 1$. We again denote by $p_{j,i}$ the ith polynomial in the jth box, ordered by degree. As before, we need an irreducible polynomial $g \in \mathbb{F}_q[x]$ of large degree as our modulus, but this time, we require that $\sum_{j=1}^{t} \ell \deg p_{j,b} < \deg g = N$. Again, we choose integers e and s with $es \equiv 1 \mod q^N - 1$ and set $v_{j,i} = p_{j,i}^e$. The public key is $\big(\{v_{j,i}\}_{i,j}, \ell, \mathbb{F}_q[x]/(g(x))\big)$ and the private key is $\big(\{p_{j,i}\}_{i,j}, s\big)$.

For each box S_i, we now have more options available for encryption than simply choosing one element of S_i: we can choose up to ℓ elements, allowing repetitions, and multiply those. Each of these possibilities corresponds to a b-tuple in $T = \{(k_1, \ldots, k_b) \in \mathbb{N}^b \mid k_1 + \cdots + k_b \leq \ell\}$. As shown in [2, Appendix A], there are $\binom{b+\ell}{\ell} = B$ such tuples, and there is a bijection $\varphi\colon \{1, \dots, B\} \to T$ that can be computed efficiently [5]. Hence, we use the message space $\{1, \dots, B\}^t$, and we encrypt a message $m = (m_1, \dots, m_t)$ as

$$m \mapsto \prod_{j=1}^{t} \prod_{i=1}^{b} v_{j,i}^{k_{j,i}} = c \in \mathbb{F}_q[x]/(g(x)),$$

where $\varphi(m_j) = (k_{j,1}, \dots, k_{j,b}) \in T$.

Decryption is again done by lifting and factoring c^s and inverting φ.

We can again give a formula for information rate and public key size. The information rate is $\frac{t \log B}{N \log q}$, and the public key still has size $btN \log q$.

11.4.1 Toy Example

We present a small example to clarify the "powers of primes" method. Let $q = 2$, and we consider a system with $t = 2$ packs of $b = 3$ irreducible polynomials each. Let furthermore $\ell = 2$. The first six irreducible polynomials are

$$\begin{aligned} p_{1,1} &= x & p_{2,1} &= x^3 + x + 1 \\ p_{1,2} &= x + 1 & p_{2,2} &= x^3 + x^2 + 1 \\ p_{1,3} &= x^2 + x + 1 & p_{2,3} &= x^4 + x^3 + 1. \end{aligned}$$

We need $\ell \deg p_{1,3} + \ell \deg p_{2,3} = 12 < \deg g = N$, so we choose

$$g = x^{13} + x^4 + x^3 + x + 1.$$

We randomly choose secret exponents $e = 6020$ and $s = 6380 \equiv e^{-1} \mod 2^{13} - 1$. The public elements are now given by $v_{j,i} \equiv p_{j,i}^e \mod g$:

$$\begin{aligned} v_{1,1} &= x^{11} + x^{10} + x^9 + x^8 + x^6 + x^5 & v_{2,1} &= x^8 + x^7 + x^6 + x^5 + x^4 + 1 \\ v_{1,2} &= x^{11} + x^{10} + x^9 + x^8 + x^6 + x & v_{2,2} &= x^{12} + x^{11} + x^6 + x^5 + x^3 \\ v_{1,3} &= x^{12} + x^{11} + x^{10} + x^9 + x^5 + x^3 + x^2 + 1 & v_{2,3} &= x^{12} + x^{11} + x^{10} + x^6 + x^5 + x^2. \end{aligned}$$

Note that $B = \binom{3+2}{2} = 10$, so we can represent a message in base 10. We choose the following encoding from integers 0 to 9 to 3-tuples (k_1, k_2, k_3) satisfying $k_1 + k_2 + k_3 \leq 2$.

$$\begin{array}{lllll} 0 \mapsto (0,0,0) & 1 \mapsto (1,0,0) & 2 \mapsto (2,0,0) & 3 \mapsto (0,1,0) & 4 \mapsto (1,1,0) \\ 5 \mapsto (0,2,0) & 6 \mapsto (0,0,1) & 7 \mapsto (1,0,1) & 8 \mapsto (0,1,1) & 9 \mapsto (0,0,2). \end{array}$$

Table 11.2 Information rate and public key size of the "powers of primes" variant for $q = 6287$, $\deg g = 131$ and various box sizes and bases

b	ℓ	t	Information rate (%)	Public key size (kbit)
1	1	130	7.9	215
2	2	65	10.1	215
10	10	13	13.8	215
30	1	130	39.0	6447
42	2	65	38.9	4513
310	26	5	38.8	2562
83	26	5	25.1	686

To encrypt the message $m = 94$, we hence compute

$$v_{1,1}^0 v_{1,2}^0 v_{1,3}^2 \cdot v_{2,1}^1 v_{2,2}^1 v_{2,3}^0 \equiv x^{12} + x^9 + x^8 + x^3 + x^2 + 1 = c \mod g.$$

To decrypt, raise the ciphertext to s and factor:

$$\begin{aligned} m^s &\equiv x^{10} + x^9 + x^6 + x^5 + x^4 + x + 1 \mod g \\ &= (x^2 + x + 1)^2 \cdot (x^3 + x + 1) \cdot (x^3 + x^2 + 1) \\ &= p_{1,1}^0 p_{1,2}^0 p_{1,3}^2 \cdot p_{2,1}^1 p_{2,2}^1 p_{2,3}^0, \end{aligned}$$

from which the message is recovered.

11.4.2 Example Parameters

We again consider the case $q = 6287$ and compare the information rate and public key size of the "powers of primes" variant in the case $\deg g = 131$ for different values for b and ℓ in Table 11.2. The first row corresponds to the original pNSK, which is obtained by setting $b = 1$ and $\ell = 1$.

As we can see, the "powers of primes" method allows, to an extent, for larger information rates at the same key size, or for smaller keys for a given information rate.

11.5 Security

As for the original Naccache-Stern cryptosystem, we do not know of a security proof for the pNSK, with or without our information rate improvements. However, we can recall a few considerations regarding the security of NSK from [2, 4], which also apply to our variant.

First of all, note that our system is broken if one can solve a discrete logarithm problem $p_{j,i}^s = v_{j,i}$, as this directly reveals the secret key. Although the $p_{j,i}$ don't have to be released publicly, they must have low degree and can thus be guessed easily. Hence, it is important to choose parameters in such a way that the field $\mathbb{F}_q[x]/(g(x))$ is large enough to withstand a DLP attack. Compared to the original NSK, we have to be even more careful due to recent quasipolynomial attacks on small characteristic [1].

As remarked in [4], a birthday-search attack on the message is possible on all NSK variants. In our case, this happens by dividing the packs S_j into two sets T_1 and T_2 of similar size and searching for a collision in an appropriate way. For example, in the "powers of primes" situation, one could look for exponents $k_{j,i}$ such that

$$\prod_{j \in T_1} \prod_{i=1}^{b} v_{j,i}^{k_{j,i}} = c \cdot \prod_{j \in T_2} \prod_{i=1}^{b} v_{j,i}^{-k_{j,i}}.$$

To prevent this, the size of the message space should be chosen to be at least twice the desired security level.

Furthermore, since $2 \mid q^d - 1$ for odd q, it is possible to find the parity of the number of factors $v_{j,i}$ in a ciphertext c that are quadratic nonresidues in $\mathbb{F}_q[x]/(g(x))$ by simply checking whether c itself is a quadratic residue. This is only a small information leakage, but nonetheless it should be avoided by encoding messages in such a way that this parity is always the same. A similar attack can be applied for other small factors of $q^d - 1$, so it should be chosen to have few such factors.

Acknowledgments The authors were supported by Swiss National Science Foundation grant number 149716 and *Armasuisse*

References

1. Barbulescu R, Gaudry P, Joux A, Thomé E (2014) A heuristic quasi-polynomial algorithm for discrete logarithm in finite fields of small characteristic. In: Advances in Cryptology-Eurocrypt 2014. Springer, pp 1–16
2. Chevallier-Mames B, Naccache D, Stern J (2008) Linear bandwidth naccache-stern encryption. In: Security and Cryptography for Networks. Springer, pp 327–339
3. Micheli G, Schiavina M (2014) A general construction for monoid-based knapsack protocols. Adv Math Commun 8(3)
4. Naccache D, Stern J (1997) A new public-key cryptosystem. In: Advances in Cryptology, EUROCRYPT. pp 27–36
5. Stanton D, White D (1986) Constructive combinatorics. Springer, New York
6. Stein W, et al. (2014) Sage mathematics software (version 6.1.1). The Sage Development Team, http://www.sagemath.org