# A Non Euclidean Ring Data Scrambler (NERDS) —a public–key cryptosystem—

Emiliano Kargieman Ariel Pacetti Ariel Waissbein
(COREIabs, CORE–sdi)

September 29th, 2000

e–mail: {Emiliano_Kargieman, Ariel_Waissbein}@core-sdi.com
apacetti@math.utexas.edu

**Abstract**

In this paper, we introduce the Non–Euclidean Ring Data Scrambler public–key cryptosystem, NERDS. This cryptosystem consists of efficient linear algebra procedures and it's security relies on different problems in Algebraic Number Theory over orders of number fields, such as the non–existence of division algorithms (nor efficient factorization algorithms) over noneuclidean domains.

**Keywords:** Public–key Cryptography, Number fields, Computational Algebraic Number Theory.

## 1 Introduction

We deal with a specific task in cryptography, that of the construction and discussion of a new public–key encryption scheme. NERDS was created to be a secure encryption scheme and of a greater efficiency than the one achieved by the public–key encryption schemes in use. The underlying idea towards this end was that only a few multiplications (of height–controlled integers) are needed for both the encryption and decryption processes. This feature makes our scheme very efficient. In the construction of NERDS, and on this account, we focused only on the security issue —while having the efficiency

issue in mind. That is, the algorithms of encryption, decryption and key generation we give are efficient but not optimally efficient.

Public–key cryptosystems where introduced in 1976 by Diffie and Hellman ([DH76a, DH76b]) and have been widely used ever since (see the survey [Dif92] for a complete account). A public–key encryption scheme consists of three algorithms: key generation, encryption and decryption. It is furthermore asked that the scheme consists of *efficient* procedures, and that it is *infeasible* to violate the scheme's security feature (see [Gol99]).

As for the efficiency issue, we measure the complexity of our algorithms in number of arithmetic operations $op \in \{+, -, \times, \div\}$ over the ring of rational integers $\mathbb{Z}$. We will not state the bit complexity of our algorithms. However the bit complexity analysis can be easily derived from our results. The encryption algorithm in NERDS can be carried out in $4D^2$ arithmetic operations ($2D^2$ multiplications and $2D^2$ additions), and the decryption process in $2D^2 + \log_2 D$ arithmetic operations. The key generation cost will be analyzed later on.

As for the security issue on our public–key encryption scheme, we can only but point out infeasible ways of breaking our scheme to synthetize the scheme's *impending security*. Let us be more explicit. The cryptographic tools in use nowadays rely on the intractability of different intractable computational problems (see [GJ79] for an introduction to intractability), with exception perhaps to the upcoming Quantum Cryptography protocols —if ever implemented— such as the key–exchange protocol of [BB84] and [Wie83]. This means that to tackle these cryptographic schemes it is sufficient (but not necessary) to solve certain intractable problems. For example, the RSA public–key cryptosystem [RSA78] is a widely used public–key encryption scheme relying on the difficulty of factoring large integers, the McEliece cryptosystem [McE78] relies on a problem in Algebraic Coding Theory, and Finite Fields and Elliptic Curve cryptosystem, such as El Gammal (see [ElG85] and [Kob87, Mil86]) rely on different variants of the Discrete Logarithm Problem. The NERDS cryptosystem relies on (e.g., can be broken by solving efficiently) one of different problems in Algorithmic Algebraic Number Theory such as *norm equation solving*, or *factorization in an order*. We shall get deeper into this in the Subsection 3.2.

This paper lurks with concepts in the realm of Cryptography and Algebraic Number Theory, we refer the unschooled reader to the textbooks [Sti95], [Gol99] or [Sch95] in the case of Cryptography and [Coh96] or [PZ98] for (the algorithmic aspects of) Algebraic Number Theory.

We now give a concise introduction to the key generation, encryption and

decryption procedures in the NERDS encryption scheme. In fact we give a concise but slightly different explanation of the encryption and decryption procedures. The NERDS underlying algorithms, together with that of key generation, will be thoroughly explained in the next section.

A ring of algebraic integers $A$ is made public. This ring $A$ is chosen to be isomorphic to the order $\mathbb{Z}[X]/(p(X))$ of the number field $\mathbb{Q}[X]/(p(X))$, where $p \in \mathbb{Z}[X]$ is an irreducible (in $\mathbb{Q}[X]$) univariate monic polynomial with integer coefficients and of degree $\deg p(X)$. We also ask of $p(X)$ to make $\mathbb{Z}[X]/(p(X))$ a noneuclidean domain. Each element in $A$ is represented (in the basis induced by $X$ in $\mathbb{Z}[X]/(p(X))$) by a unique vector of rational integers of a fixed length, namely of length $\deg p(X)$. The cleartext, the ciphertext, the private key, and the public key will all be elements belonging to $A$. The public key consists of the elements $M$ and $b$ in $A$, where $M$ is such that $b$ is not invertible in the quotient ring $A/(M)$. The private key consists of an element $n$ in $A$, such that $n$ is a divisor of the element $M$ in the public key, and $b$ is invertible in the quotient ring $A/(n)$. We note that, under our hypotheses, $A$ is noneuclidean and thus no algorithm for calculating factorizations or common divisors (if existing) are known. So that $n$ cannot be efficiently recovered by an attacker.

We now describe how to encrypt and decrypt messages. The key generation method will be explained in Section 2.1 and studied in detail in Section 3.1. Let $m \in A$ be the cleartext. To get the ciphertext $c$ we make the multiplication $b \cdot m$ in $A$ and then "take modules" in $A/(M)$, i.e. $c$ is a pseudo–random representative in $A$ of the class induced by $b \cdot m$ in the quotient ring $A/(M)$.

The recovery of the cleartext from the ciphertext is done as follows. Let $c$ denote the ciphertext. Since —by hypotheses— $n$ divides $M$ in $A$, it follows that any congruence holding modulo $(M)$ also holds modulo $(n)$, and hence the congruence $c \equiv b \cdot m$ modulo $(n)$ holds in $A$. Being $b$ invertible modulo $(n)$, the homothety endomorphism $a \mapsto a \cdot b$ of $A/(n)$ induced by multiplication by $b$ is in fact a monomorphism.

Suppose that a representative $b^{-1}$ in $A$ of the inverse $\overline{b^{-1}}$ of the class $\overline{b}$ induced by $b$ in $A/(n)$ is also known by the private party. (Else it can be calculated from $b$ and $n$; see Lemma 1). To decrypt the ciphertext we multiply $c$ by $b^{-1}$ in $A$, obtaining thus an element in $A$ which belongs to the class induced by the cleartext $m$ in $A/(n)$, i.e. $b^{-1} \cdot c \equiv m$ modulo $(n)$. The conditions imposed to the previous objects (more specifically to $n$ and $m$) will ensure us that there is —and we can calculate— a unique representative of $m$ in a certain subset to which $m$ belongs, hence we get $m$. Furthermore we will show that $m$ can be efficiently computed by linear algebra procedures

and modular arithmetic (in $\mathbb{Z}$).

We also remark that $b$ is not invertible in the quotient ring $A/(M)$, which in turn implies that the homothety endomorphism of $A/(M)$ induced by the multiplication by $b$ is not injective. Even more, to enhance the scheme's security, the parameters are chosen so that the fiber of every $\bar{a} \in A/(M)$ under the homothety homomorphism induced by multiplication by $b$ from $A/(n)$ to $A/(M)$ has a large number of elements.

# 2 The NERDS Cryptosystem

To make a detailed description of the NERDS encryption scheme we first need to make some definitions and fix the notation. Let $\mathbb{Z}$ denote the ring of rational integers, $\mathbb{Q}$ denote the rational numbers field, and let $\mathbb{C}$ denote the complex numbers field. Let $X$ be an indeterminate over the field $\mathbb{Q}$ of the rational numbers.

Let $p(X) \in \mathbb{Z}[X]$ be an univariate monic polynomial with integer coefficients, irreducible in $\mathbb{Q}[X]$. Let $D := \deg p(X)$ denote the degree of the polynomial $p(X)$. Suppose that the ring $\mathbb{Z}[X]/(p(X))$ is a noneuclidean domain. Let $\xi \in \mathbb{C}$ be any solution of the equation $p(X) = 0$. Let $A := \mathbb{Z}[\xi] \subset \mathbb{C}$ be an order (not necessarily maximal) of the field defined by an embedding of $\mathbb{Q}[X]/(f(X))$ in $\mathbb{C}$. Fix $p(X)$, and subsequently the ring $A$, for the remainder of this section. A discussion on the election of $p(X)$ will be held in the Subsection 3.1.

The ring $A = \mathbb{Z}[\xi]$ is the plaintext, ciphertext space, and key space, as well. Note that taking the order $A$ as the plaintext set is not a cumbersome choice, since $A$ is $\mathbb{Z}$–isomorphic (as a $\mathbb{Z}$–module) to $\mathbb{Z}^D$. So that we may construct efficient encoding and decoding functions $\mathbb{Z}^D \leftrightarrow A$.

The elements in $A$ (algebraic integers) are represented symbolically, and not as approximations to the complex numbers they represent. We use two different representation methods (see [Coh96]). The first one, called the *standard representation* in the literature, is the following. Let $E := \{1, \xi, \ldots, \xi^{D-1}\}$ denote the $\mathbb{Z}$–module basis of $A = \mathbb{Z}[\xi]$ induced by the primitive element $\xi$ of the extension $\mathbb{Q}[\xi]/\mathbb{Q}$. An element $a \in A$ can be expressed in this basis as $a = a_0 + a_1 \cdot \xi + \cdots + a_{D-1} \cdot \xi^{D-1}$, where $a_0, \ldots, a_{D-1} \in \mathbb{Z}$ are rational integers, then the standard representation of $a$ is $(a)_E = (a_0, \ldots, a_{D-1})_E$. Secondly, the *matrix (or regular) representation.* Let $a$ be in $A$, then the homothety endomorphism of $A$ induced by multiplication by $a$ is a $\mathbb{Z}$–homomorphism, and can then be represented by a $\mathbb{Z}^{D \times D}$ matrix in the basis $E$, which we call the matrix representation of $A$ and de-

note by $[a]_E$. Notice that if we use the basis $\{1, \ldots, \xi^{D-1}\}$ then the columns of the matrix representation $[a]_E$ of $a$, are exactly $(a)_E, (a \cdot \xi)_E, \ldots, (a \cdot \xi^{D-1})_E$. Also, note that the determinant of this matrix $[a]_E$ is precisely the norm $\det([a]_E) = Norm(a)$.

## 2.1 Key Generation

The public and private keys will be represented by their matrix representation, that is matrices in $\mathbb{Z}^{D \times D}$. To construct these matrices we want to chose elements $M, n, b$ in $A$, such that

- $n$ is a divisor of $M$,
- $b$ is invertible in the quotient ring $A/(n)$, and
- $b$ is not invertible in the quotient ring $A/(M)$.

These elements (their representations) will be typically of a large height, e.g. height 512 or 1024 depending on the security required. To make this election, we generate random elements $n, q, \tilde{q}$ in the ring $A$ of the required height. Then, using the Lemma 1 bellow we check if $q$ and $\tilde{q}$ are invertible in the quotient ring $A/(n)$. If any of them is not invertible, say $q$, a new element is chosen and its invertibility is tested (this process is repeated until the two elements pass the invertibility test). Finally, we generate a random element $e$ in $A$ and define $M := n \cdot q \cdot e$ and $b := q \cdot \tilde{q}$.

Suppose that $n = n_0 + \cdots + n_{D-1} \cdot \xi^{D-1}$ is already chosen, and let $[n]_B$ denote its matrix representation. By Smith's Normal Form theorem ([Smi61]), there exist unimodular matrices $U, V$ in $\mathbb{Z}^{D \times D}$ such that $U[n]_E V^{-1}$ is a diagonal matrix $U[n]_E V^{-1} = diag(d_1, \ldots, d_D)$, where $d_1, \ldots, d_D \in \mathbb{Z}$ are positive integers uniquely determined by $n$, and such that $d_i$ divides $d_{i+1}$ for $1 \leq i < D$. The elementary divisors $d_1, \ldots, d_D$ of $[n]_E$ and the transformation matrices $U, V^{-1}$ can be calculated from the entries of the matrix $[n]_E$ using the algorithmic techniques [HM91] (e.g., as in [Coh96]).

Once the elements $n, q, \tilde{q}$ are elected, we define the private key to be $[n]_E, U, d_1, \ldots, d_D$. And define the matrices $[M]_E$ and $[b]_E \cdot U^{-1}$ of $\mathbb{Z}^{D \times D}$ as the public key, where $U^{-1}$ is the inverse of the transformation matrix $U$ and $[b]_E \cdot U^{-1}$ denotes the matrix multiplication of these matrices (the purpose of this election will be clear in the section "Decryption"). Notice that the elements $q$ and $\tilde{q}$ are only used for this construction, but are no longer used for encryption nor decryption. Furthermore, we may want to store additional information, which can be derived from $[n]_E, U, d_1, \ldots, d_D$, as the private

key to facilitate the decryption process. This additional information will be specified in the "Decryption" (Section 2.3).

We now show how to make the invertibility test for $q$ and $\tilde{q}$. Further indications for the selection of $q$ and $\tilde{q}$ will be given in the subsection "Parameters" (Subsection 3.1). Let $n \in A$ be fixed, and consider the quotient ring $A/(n)$. We need to answer whether an element in this quotient ring is or is not invertible, i.e. if given $b$ in $A$ there exists an element $a$ in $A$ such that the congruence $a \cdot b \equiv 1$ modulo $(n)$ holds; in that case we also want to calculate a representative in $A$ of the element $a$.

The decision question can be easily tackled, for $b$ will have an inverse in $A/(n)$ if their respective norms $Norm(n), Norm(b)\mathbb{Z}$ are relatively prime rational integers. The representation question needs a little more care and will be the treated in the next lemma.

**Lemma 1** *Let $b$ be an element in $A$ given by its matrix representation, and let $n$ be an element in $A$ such that its norm $Norm(n)$ is known. Suppose furthermore, that the norms $Norm(b)$ of $b$, and $Norm(n)$ of $n$ in the extension $\mathbb{Q} \rightarrow \mathbb{Q}[\xi]$ are relatively prime rational integers, i.e. in symbols $gcd_{\mathbb{Z}}(Norm(b), Norm(n)) = 1$.*

*Then $b$ is invertible in $A/(n)$, and we can calculate, from input $[b]_E$ and $Norm(n)$, the matrix representation $[b^{-1}]_E \in \mathbb{Z}$ of an element $b^{-1} \in A$ such that $b \cdot b^{-1} \equiv 1$ modulo $(n)$ holds in $A$. This process can be done in $\log^2 h + 3D^{3.5}$ arithmetic operations where $h$ is an upper bound for $Norm(b)$ and $Norm(n)$.*

PROOF.– Calculate the determinant $det([b]_E) = Norm(b)$ of the matrix $[b]_E$. Since by hypothesis, $Norm(n)$ and $Norm(b)$ are relatively prime integers, there exist integers $s, t \in \mathbb{Z}$ such that the identity $s \cdot Norm(b) + t \cdot Norm(n) = 1$ holds in $\mathbb{Z}$. Thus, the congruence $s \cdot Norm(b) \equiv 1 \bmod (n)$ holds in $\mathbb{Z}$ (and then in $A \supseteq \mathbb{Z}$).

Let $w := \prod_{\sigma \neq id} \sigma(b)$ be the product of all the conjugates of $b$ (in a Galois closure of $\mathbb{Q}[\xi]$) different from $b$ itself. Then the identity $b \cdot w = Norm(b)$ holds in $\mathbb{Q}[\xi]$. (Hence $w$ belongs to $A$.) And in particular, it follows that $[b]_E \cdot [w]_E = diag(Norm(b), \ldots, Norm(b)) = Norm(b) \cdot Id$. Note that the matrix $Norm(b) \cdot Id$ is the matrix representation of the element $Norm(b)$ of $A$, where $Id$ denotes the identity matrix. So that $(s \cdot w) \cdot b = 1 - t \cdot Norm(n)$ and in particular $(s \cdot w) \cdot b \equiv 1 \bmod (n)$, since $Norm(n)$ is a multiple of $n$ in $A$.

Let $[Adj(b)]$ denote the adjoint matrix of $[b]_E$. The matrix $[Adj(b)]$ verifies the equality

$$[Adj(b)] \cdot [b]_E = Norm(b) \cdot Id$$

in $\mathbb{Z}^{D\times D}$. So that $[Adj(b)] = [w]_E$ must hold in $\mathbb{Z}^{D\times D}$, and then $s\cdot[Adj(b)]$ is the matrix representation of an element $b^{-1}$ in $A$ which represents the inverse of $b$ modulo $(n)$.

Algorithmically we calculate the determinant of the matrix $[b]_E$ with $O(D^{3.5})$ arithmetic operations. We apply the extended Euclidean algorithm to the norms $Norm(b)$ and $Norm(n)$, computing $s$ in $\log(Norm(b))\log(Norm(n)) \leq \log^2(h)$ operations. We calculate the adjoint $[Adj(b)]$ of $[b]$ by Paterson–Stockmeyer's strategy [PS73] using no more than $3D^{3.5}$ arithmetic operations. Thus, this procedure results in the announced complexity. ∎

## 2.2 Encryption

A cleartext $m$ is called valid if $\|(m)_E\|_\infty \leq k := d_1$. Let $m \in A$ be the (valid) cleartext we want to encrypt. And let $[M]_E$ the matrix representation of $M$, and $[b]_E \cdot U^{-1}$, the product of $[b]_E$ and $U^{-1}$, be the public key. We define the procedure for taking congruences modulo $(M)$ in $A$.

**Remark 2 (Taking congruences mod $(M)$ in A)** *Let be given two elements $M$ and $a$ in $A$ by their matrix $[M]_E$ and standard $(a)_E$ representation respectively. Then we can calulate a* pseudo–random *element $\bar{a} = \bar{a}_0 + \cdots + \bar{a}_{D-1}\cdot\xi^{D-1} \in A$ in the class induced by $a$ in $A/(M)$. The standard representation $(\bar{a})_E$ of $\bar{a}$ can be calculated in $2D^2$ arithmetic operations.*

We now explain how is this done. Let $[M]_E$ stand for the matrix of the homothety induced by $M$ in $\mathbb{Z}[\xi]$ in the canonical base. Then by taking congruences we mean calculating the vector $(\bar{a})_E = (\bar{a}_0, \ldots, \bar{a}_{D-1}) = (a)_E + [n]_E(r)_E$, where $(r)_E$ is a vector of randomly chosen integers. Hence, only a vector–to–matrix multiplication is needed, and thus the process can be executed in the stated complexity. (We do not count the complexity needed for random number generation.)

To produce the ciphertext out of the cleartext using public key $[b]_E\cdot U^{-1}$, $[M]_E$, let $m'$ be the unique element in $A$ with standard representation $(m')_E := U^{-1}(m)_E$, then we scramble the cleartext by calculating a representative $c \in A$ of $b\cdot m'$ modulo $(M)$ in $A$ as explained in Remark 2, e.g., such that

$$c \equiv b\cdot m' \bmod (M) \tag{1}$$

Specifically, the encryption process consists of the determination of a vector of random integers $(r)_E = (r_0, \ldots, r_{D-1})$, and the subsequent calculation of the integer vector $(c)_E = (c_0, \ldots, c_{D-1}) = ([b]_E U^{-1})(m)_E + [M]_E(r)_E$ which can be done with $4D^2$ arithmetic operations.

## 2.3 Decryption

Let $[n]_E, U, d_1, \ldots, d_D$ be the private key and suppose calculated for once and for all the matrix $U[b^{-1}]_E$ (this task can be fulfilled following Lemma 1). And let $(c)_E$ denote the ciphertext.

To recover the cleartext, first the decrypter calculates $U[b^{-1}]_E(c)_E$. Let $m'$ denote the unique element in $A$ with standard representation $(m')_E := U^{-1}(m)_E$ as in the previous subsection. Since $n$ divides $M = n \cdot q \cdot e$ in $A$, and $c \equiv b \cdot m'$ modulo $(M)$ holds in $A$, the congruence $c \equiv b \cdot m' \mod (n)$ also holds. Then we have that

$$b^{-1} \cdot c \equiv b^{-1} \cdot \left(b \cdot m'\right) \equiv m' \mod (n)$$

holds in $A$. So we deduce that $c \cdot b^{-1}$ is an element in $A$ congruent to $m'$ modulo $(n)$. That is, there exists an element $\hat{r}$ in $A$ such that

$$\begin{aligned} b^{-1} \cdot c = & \quad b^{-1}(b \cdot m' + n \cdot r) \\ = & \quad m' + n \cdot \hat{r} \end{aligned}$$

and then, multiply the standard/matrix representation of these identity to the left by $U$, we get

$$U[b^{-1}]_E(c)_E = (m)_E + U[n]_E(\hat{r})_E \tag{2}$$

Hence, recalling the Smith Normal Form $U[n]_E V^{-1} = diag(d_1 \ldots, d_D)$ of $[n]_E$, we have

$$\begin{aligned} U[b^{-1}]_E(c)_E & = (m)_E + U[n]_E(\hat{r})_E \\ & = (m)_E + (U[n]_E V^{-1})V(\hat{r})_E \\ & = (m)_E + diag(d_1, \ldots, d_D)(\hat{\hat{r}})_E \end{aligned}$$

(where $V(\hat{r})_E = (\hat{\hat{r}})_E$ in $\mathbb{Z}\!\!\mathbb{Z}^D$.) In particular, for $1 \leq i \leq D$, the $i$–th entry of the vector $diag(d_1, \ldots, d_D)(\hat{\hat{r}})_E$ is a multiple of $d_i$. Recall that $(m)_E$ was

chosen so that $\|(m)_E\|_\infty \leq k = d_1 = \min\{d_1, \ldots, d_D\}$. Hence, for $1 \leq i \leq D$, the decrypter computes the $i$–th entry of the vector $(m)_E$ by taking congruences in the $i$–th entry of the vector $U[b^{-1}]_E(c)_E$ modulo $d_i$ therefore computing their respective representatives in $0 \leq m_i < d_i$.

Following the above procedures we are able to decrypt a given ciphertext in $2D^2 + D \log D$.

# 3 Implementation and Cryptanalysis

## 3.1 Choosing the Parameters

There are two stages in the choosing of parameters. First the election of $p(X)$ (and thence $A$), and second the selection of the private and public keys.

In choosing the polynomial $p(X)$ (and $A$) we consider both the efficiency issue and the security issue as well. Recall that $p(X) \in \mathbb{Z}[X]$ is chosen to be an univariate polynomial with integer coefficients, monic and irreducible in $\mathbb{Q}[X]$. We asked furthermore, that the ring $\mathbb{Z}[X]/(p(X))$ turns to be a noneuclidean domain. As previously seen, the elements in the ring $A \cong \mathbb{Z}[X]/(p(X))$ are represented by integer vectors of size $D = \deg p(X)$ (or integers matrices of size $D \times D$ alternatively). Hence, polynomials $p(X)$ of low degree $D$ are preferable. Namely, because the complexities of the algorithms used in NERDS (key generation, encryption and decryption) depend polynomially on $D$.

Another parameter to be taken into consideration is the signature $(r_1, r_2)$ of the field $\mathbb{Q}[X]/(p(X))$. (By signature we understand a pair of integers $(r_1, r_2)$, $r_1$ being the number of real embeddings of $\mathbb{Q}[X]/(p(X))$ in $\mathbb{C}$, and $2r_2$ being the number of non-real embeddings of $\mathbb{Q}[X]/(p(X))$ in $\mathbb{C}$.) We remark that it is preferable to have a polynomial $p(X)$ with the largest possible amount $r_1$ of real roots, since the group of units of $\mathbb{Z}[X]/(p(X)) \cong A$ is isomorphic to the direct product of a finite cyclic group (the torsion subgroup) and $r_1 + r_2 - 1$ infinite cyclic groups. Because the larger $r_1$, the more elements having the same norm occur in $A$ (see Subsection 3.2).

A further characteristic which can also be asked for from $p(X)$ is having large discriminant, or rather that the splitting field of $\mathbb{Q}[X]/(p(X))$ has large discriminant, since the complexity of some algorithms in Algebraic Number Theory (that might tackle NERDS) also depend on this discriminant.

Polynomials $p(X)$ satisfying our requirements can be chosen from existing number fields tables (see [Coh96], [PZ98] and [Lem99]), or by building new

tables (see *op. cit.*).

A discussion on the election of $n$ was held in the Section 2.1. We shall only ask of $n$ to have a large norm, since the security of our scheme (also) relies in this parameter, and that the resulting normal form consists in large integers $d_i$, since $k^D = d_1^D =\leq d_1 \cdot \ldots \cdot d_D$ is the amount of encryptable/decryptable cleartexts.

Finally, a word on the elements $q, \tilde{q}$ and $e$. Since the quotient rings $A/(n)$ and $A/(M)$ have $Norm(n)$ and $Norm(M)$ elements respectively, we note that each element in $A/(n)$ has $\frac{Norm(M)}{Norm(n)} = Norm(q \cdot e)$ distinct representatives in $A/(M)$. Hence $q$ will be chosen so that $Norm(q)$ is a large integer and it is invertible in the quotient ring $A/(n)$. The elements $\tilde{q}$ and $c$ are chosen randomly subjected to the constrain that $\tilde{q}$ is invertible in $A/(n)$.

## 3.2 Cryptanalysis

In this section we introduce two cryptanalytic unsuccessful attacks around the which the security of NERDS was built.

The following attacks on NERDS retrieve the private key if successful. By any of these attacks, the problem of recovering the private key can be reduced to solving norm equations of the type

> *find* $x \in A$ such that $Norm_{\mathbb{Q}(\xi)/\mathbb{Q}}(x) = t$, *where* $t$ *is a (typically large) given integer.*

The first attack goes as follows. Given the public key $[M]_E, [b]_E U^{-1}$, an attacker can calculate the norms $Norm(M)$ of $M$ and $Norm(b)$ of $b$ as the determinants $Norm(M) = \det([M]_E)$ and $Norm(b) = \det([b]_E U^{-1}) = \det([b]_E)$. Then the norm $Norm(q)$ of $q$ can be calculated as a divisor of the greatest common divisor $gcd_{\mathbb{Z}}(Norm(M), Norm(b))$. Note that $Norm(n)$ cannot be directly calculated, we only know that $Norm(n)$ is an (integer) factor of $Norm(M)/gcd_{\mathbb{Z}}(Norm(b), Norm(M))$, which in turn is a divisor of $Norm(n \cdot e)$. We point out that $Norm(M)$ is typically a large integer, and thus it is computationally intractable to factorize. However, we shall show that even if the norm of $n$ is known, this attack remains fruitless.

Suppose, for what follows in this attack, that the norms $Norm(n), Norm(q)$ and $Norm(\tilde{q})$ of the elements $n, q, \tilde{q}$ in the private key are known to an attacker. Then the attacker is (actually, would be) able to compute the elements in the private key from their norms, using an *efficient* norm equation solver. That is, an attacker might calculate all the possible solutions $x$ of the equation $Norm(x) = Norm(n)$ (of a bounded height) and then check

if $x = n$. To check this, the attacker uses *the standard SNF algorithm* to calculate the the elementary divisors $d_1^{(x)}, \ldots, d_D^{(x)}$, the matrix $U[b^-1]_E$, and then encrypt a message using the public key and attempt decryption using the proposed private key $[x]_E$.

The question raised then is, how good are norm equation solvers? To this end we point out the paper [FJP97] and the book [PZ98]. The algorithms introduced in the cited works are inefficient to tackle the problem with the parameters used in NERDS. Roughly, the algorithm of Fieker, Jurk and Pohst needs $O\left(r^{r/3+4}\lambda^{r/4}v^{r/2}(3 - \frac{4\log(Norm(n))}{r\log\lambda})^r\right)$ arithmetic operations to find one element of a given norm, where $r = r_1 + r_2 - 1$, $v$ is a number depending on the number of conjugates of $\xi$ and Furthermore, in the event of $Norm(n)$ being calculated (which need not be so), the attacker will need to calculate the matrix representation $U[b^-1]_E$, and for each possible solution $(x)_E$ of a norm equation solver for $Norm(x) = Norm(n)$, also calculate the matrix representation $[x]_E$, it's Smith Normal Form, only to endevour a decryption. Notice furthermore, that the number of solutions of the norm equation is large —making these attack even less efficient.

An other word on norm attacks. We describe an alternate procedure for calculating $n$. Let $G$ denote the Galois group of the extension $K/\mathbb{Q}$, where $K$ is the normal closure of $\mathbb{Q}[\xi]$. Since the field $\mathbb{Q}[\xi]$ is public and thence $K$ can be calculated; since $[K : \mathbb{Q}]$ might be low, we might suppose that the Galois group $G$ is known to the attacker. However the ring $A = \mathbb{Z}[\xi]$ has an infinite set of units which form a group of order $r_1 + r_2$, and are not computationally easy to calculate. Let $x \in A$ be a solution of $Norm(x) = Norm(n)$. Then there exist a unit $\mu$ of $A$, and a group element $\sigma$ in $G$ such that

$$n = \sigma(\mu \cdot x) \quad .$$

Hence, we can first calculate only one solution $x$ of the equation $Norm(x) = Norm(n)$, and then, use the knowledge of $G$ and the units group, to calculate all the elements

$$\sigma(\mu \cdot Norm(n))$$

of a bounded norm, where $\mu$ is a unit of $A$ and $\sigma \in G$, are candidates for $n$. Furthermore, by [FJP97, Lemma 3.1] it suffices to find a solution of absolute value bounded by $Norm(n)^{1/(r_1+r_2-1)}exp(\sum|\log\epsilon|)$ where the sum is over a set of fundamental units. This procedure has a complexity of $O\left(r^{r/3+4}\lambda^{r/4}v^{r/2}(3 - \frac{4\log(Norm(n))}{r\log\lambda})^r\right)$.

This work was done at the laboratories CORELABS at CORE-S.D.I.

URL `http://www.core-sdi.com`

Florida 141 piso 7
Capital Federal (C1005AAC)
Buenos Aires, Argentina.

# References

[BB84] C. H. Bennett and G. Brassard. Quantum cryptography: public key distribution and coin tosing. In *Proc. Internat. Conf. Computer Systems and Signal Processing*, pages 175–179. Bangalore, 1984.

[Coh96] Henri Cohen. *A Course in Computational Algebraic Number Theory*, volume 138 of *Graduate Texts in Mathematics*. Springer, 1996.

[DH76a] W. Diffie and M.E. Hellman. Multiuser cryptographic techniques. volume 45, pages 109–112. AFIPS Conference Proceedings, 1976.

[DH76b] W. Diffie and M.E. Hellman. New directions in cryptography. *IEEE Transactions of Information Theory*, 22:644–654, 1976.

[Dif92] W. Diffie. The first ten years of public–key cryptography. In *Contemporary Cryptology, The Science of Information Integrity*, pages 135–175. IEEE Press, 1992.

[ElG85] T. ElGammal. A public key cryptosystem and a signature scheme based on the discrete logarithm. *IEEE Transactions of Information Theory*, 31:469–472, 1985.

[FJP97] C. Fieker, A. Jurk, and M. Pohst. On solving relative norm equations in algebraic number fields. *Math. Comput.*, 66:399–410, 1997.

[GJ79] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, 1979.

[Gol99] O. Goldreich. *Modern Cryptography, Probabilistic Proofs and Pseudo–randomness*, volume 17 of *Algorithms and Combinatorics*. Springer, 1999.

[HM91] J. Hafner and K. McCurley. Asymptoticaly fast triangulation of matrices. *SIAM Journal of Computing*, 20:1068–1083, 1991.

[Kob87] N. Koblitz. Elliptic curve cryptosystems. *Mathematics of Computation*, 48:203–209, 1987.

[Lem99] Franz Lemmermeyer. The euclidean algorithm in algebraic numberfields. Update of a survey published by the author in 1995 in Expo. Math. now published on the author's webpage, 1999.

[McE78] R. McEliece. A public–key cryptosystem based on algebriac coding theory. Technical Report 42–44, DNS Progress Report, 1978.

[Mil86] V. Miller. Use of elliptic curves in cryptography. *Lecture Notes in Computer Science*, 13:300–317, 1986.

[PS73] M. S. Paterson and L. J. Stockmeyer. On the number of nonscalar multiplications necessary to evalutate polynomials. *SIAM J. Comput.*, 2:60–66, 1973.

[PZ98] M. Pohst and H. Zassenhaus. *Algorithmic Algebraic Number Theory.* Cambridge Press, 1998.

[RSA78] R.L. Rivest, A. Shamir, and L.M. Adelman. A method for obtaining digital signatures and public–key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[Sch95] Bruce Schneier. *Applied Cryptography, Protocols, Algorithms and Source Code in C (second edition).* John Wiley and sons, 1995.

[Smi61] H.J.S. Smith. On systems of linear indeterminate equations and congruences. *Philosophical transactions of the royal society of london (A)*, 151:293–326, 1861.

[Sti95] D.R. Stinson. *Cryptography: theory and practice.* Discrete Mathematics and its applications. CRC Press, 1995.

[Wie83] S. Wiesner. Conjugate coding. *Sigact News*, 15(1):77–88, 1983.