Secure and Fast Digital Signatures using BCH Codes

Omessaâd HAMDI[*]*, Sami HARARI*[**] *and Ammar BOUALLEGUE*[***],

[(*),(***)]SYSCOM Laboratory, National school of engineering of Tunis, Tunisia
[(*),(**)]SIS Laboratory, South University of Toulon and Var , France

Summary
In this paper, we present new practical digital signature schemes based on chained BCH code. Their safety rests on the well-known Syndrome Decoding problem (SD). We introduce the first practical and secure digital signature scheme using a generator matrix as key. Then, we generate a signature scheme faster than the only practical one based on Niederreiter cryptosystem.

Key words:
Cryptography, Coding theory,Ddigital signature, Public key size.

Introduction

Since the introduction of public key cryptography in the 70's [1], many cryptosystems have been proposed and many cryptographic schemes have been broken. The most used cryptosystems rely on number theory problem like the factorization problem [3] and the discrete logarithm over suitable group [2]. The McEliece cryptosystem [5] and the Neiderreiter variante [6] rely on coding theory, they are ones of the few cryptosystems, which are very secure and which are not broken although they do not rely on number theory. These cryptosystems present many advantages: they are very fast for both encryption and decryption and the best attacks complexity are exponential in the length of the code. These cryptosystems have the drawback to have a large public key which is a generator matrix or a check parity matrix of a long code. Another drawback related to the belief that we can not deduce a digital signature from these public key cryptosystems. In 2001, Courtois, Finiasz and Sendrier [15] introduced the first signature scheme based on McEliece cryptosystem.
Firstly, they have presented a scheme based on McEliece cryptosystem using generator matrix. With the proposed secure parameters, this scheme is impractical. Secondly, they have introduced a short practical signature based on the Neiderreiter variant using the parity check matrix as key. This scheme has the drawback to have a slow signature algorithm.
In this paper, we introduce new performant digital signature schemes based on coding theory similar to those based on McEliece and Niederreiter cryptosystems. The idea of our schemes consists in considering a chained BCH code. The resulting code will be a secret code which will be scrambled and permuted to obtain the public code.

The paper is organized as follows: We, first, define "chained BCH code". Then, we introduce a digital signature scheme using the generator matrix of chained BCH code. Unlike, the scheme based on the McEliece cryptosystem, our scheme has practical parameters and it is performant in terms of public key size and in signature length.
Before concluding, we present the dual version of this scheme. This second scheme permits to skirt the drawback of the signature based on Neiderreiter cryptosystem.

2. Definition

In order to build a system based on chained error correcting codes, we need a family of linear codes with given parameters, that has some "good" cryptographic properties. Each code of this family should have a polynomial complexity decoding algorithm.
The family must be large enough to avoid an exhaustive attack and each code C_i of the family is defined by its generator matrix (respectively a check parity matrix) M_i . The obtained system from chaining these codes called "chained error correcting code " has a resulting generator matrix (respectively a parity check matrix) with the following form:

$$M = \begin{pmatrix} M_1 & & 0 & 0 & 0 \\ 0 & ... & 0 & 0 & 0 \\ ... & & M_k & & ... \\ ... & & ... & ... & 0 \\ 0 & & 0 & 0 & M_l \end{pmatrix}$$

In our case, codewords and syndromes are stocked in tables. To decode a word, we compare it to the table elements and we take the nearest one to this word or the equal syndrome. So, the decoding operation is very fast. It consists, only, in some n length vector comparison.

Manuscript received September 5, 2006.
Manuscript revised October 25, 2006.

3. Signature with chained BCH code (BCHS1)

3.1. Algorithm parameters

Elements of $GF(2^n)$ are called words and elements of C_i are called codewords of C_i. A code is usually defined by its generating matrix. The distance between two words of $GF(2^n)$ is the Hamming distance, that is the number of positions in which they differ. The weight of a word of $GF(2^n)$ is its Hamming distance to zero-words.

Γ denotes a family of linear codes. A code C_i of Γ will be defined by its length n, its dimension k_i and its correction capacity t_i.

In order to obtain an efficient digital signature, we need an algorithm able to compute a signature for any document and a fast verification algorithm to anyone.

Thus, the most BCH codes can not satisfy the first point. In fact, if we consider a random word of length n (n is the length of the chosen code), it is usually chosen at a distance greater than the decoding capacity of this code. In other terms, we can find a word which is not decodable.

One solution to this problem is to obtain an algorithm able to decode any word of the space $GF(2^n)$. In [8], N.Courtois, M.Finiasz and N.Sendrier have introduced a method named *complete decoding* to solve this problem.

Complete decoding consists in finding the nearest codeword to any given word of the space $GF(2^n)$. To achieve this goal, we do not decode at the correction capacity t but at a distance $c = t + \delta$.

In [8], authors evaluate the smallest δ for which the volume of the sphere with radius δ is greater than 2^{n-k}.

$$\delta_{\min} = \min\left\{ \delta \in \mathrm{N} \quad \middle| \quad \sum_{i=0}^{i+\delta} C_n^i > 2^{n-k} \right\} \quad (1)$$

Nextly, we denote by BCH $[n, k_i, t_i]$ a BCH code of length n, of dimension k_i and correction capacity t_i.

The chosen BCH codes set is formed by BCH$[15,11,1]$, BCH$[15,11,2]$ completed by one bit and BCH$[15,11,3]$ completed by two bits using (1).

- *Secret parameters*

- A family Γ of l BCH codes. The chained BCH code is defined by its generator matrix G.
- A secret binary permutation matrix P.
- A secret binary invertible matrix S.

- *Public parameters*

The public key of our scheme consists in:

- A matrix G' defined by:

$$G' = S.G.P$$

The matrix G is obtained from chaining BCH codes. By the algorithm construction, G' is a permuted and scrambled generating matrix.

3.2. Key generation

In our scheme, the public key is deduced from the secret key as described in section 2. In this section, we need only to describe how the secret key and exactly the generating matrix G is obtained.

We consider a family Γ of BCH codes. A code C_i is defined by its length $n = 15$, its dimension $k_i \in \{5,7,11\}$ and its correction capacity t_i. Codes of Γ are chosen randomly.

The generated matrix G results from chained BCH code. So, G is diagonal in blocs whose diagonal is formed by elementary generating matrices of used codes (see section 2).

To hide the G structure, we permute its columns and then it will be multiplied by an invertible matrix to obtain the public key G'.

3.3. Signing a document

In the present section, we describe the signature of a message M by our new scheme.

- *Signing algorithm*

The document M to be signed is given by a binary sequence of length N.

The first step, that a signer must do, is to choose parameters shown in the previous section. Indeed, the signer chooses a random generator of codes, an invertible matrix S and a permutation matrix P.

In the second step, he constructs the generating matrix G of chained linear error correcting code. Then, he calculates the matrix product:

$$G' = S.G.P$$

and publishes this entity G'.

The third stage is the signature: the signer who wants to sign the message M, formed by a binary sequence of length N bits, permutes his message by multiplying it by the inverse of P. Then, he splits his message in words of length n to get a family of words m_i. He applies systematic decoding, that means he looks for the nearest codeword and he keeps the first k_i bits of every m_i (k_i represents dimension of used code to decode m_i).

The concatenated blocks obtained after decoding represent a binary chain of length K which depends on generated codes. This sequence will be multiplied by a matrix S^{-1} to obtain a chain y of length K.

y is the signature of M.

Let's regroup this in an algorithm

- *Parameters:*

- A family of l BCH codes of length $n = 15$ and of dimension $k_i \in \{5,7,11\}$. The resulting matrix G of the chained codes is a diagonal matrix..
- The $N \times N$ binary permutation matrix P.
- The $K \times K$ binary random invertible matrix S.

- *Public key*:

$$G' = S.G.P$$

- *Secret key* :

$$\{S, G, P\}$$

- *Signature*:

M is a document to be signed of length N.

- Compute $\rho = h(M)$
- Compute $x = \rho.P^{-1}$
- Split x in $(x_1, x_2, ..., x_l)$; $x_i, i = 1..l$ are sequences of n bits
- Decode $a_i = dec_{k_i}(x_i)$; $i = 1..l$;
- $a = (a_1, a_2, ..., a_l)$ has a length $K = \sum_i k_i$
- $y = a.S^{-1}$

y is the signature.

- *Control algorithm*

To control the validity of the signature, the verifier receives the message M, its signature y and the sum of correction capacities of used codes. He multiplies y by the resulting generator matrix G' to get a binary chain b of length N and he compares the distance between b and $h(M)$ to the sum of the used codes capacity correction.

The signature is valid if the following condition is satisfied:

$$d_H(b, h(M)) < \sum_i c_i$$

3.4. Implementation

In this section, the key generation, signature and verification algorithms will be implemented and discussed in more details.

- *Key Generation*

This is certainly the most complicated part of the algorithm. It is certainly less critical than the signature and verification, but the nature of the operations (e.g. inversion of a $K \times K$ matrix) requires careful coding if a reasonable performance is expected. The key generation involves the following steps:

- Select a random $N \times N$ permutation matrix P and construct the invertible scramble matrix S as follows: $S = S_1 \times S_2$, where S_1 is a lower triangular matrix over $GF(2)$ with random entries and S_2 is an upper triangular matrix over $GF(2)$ with random entries and with diagonal elements equal to 1. The inverse S^{-1} is easily computed as $S^{-1} = S_2^{-1} \times S_1^{-1}$
- Generate randomly the family of BCH codes.

The public key G' is computed as $S.G.P$, and the secret key consists of S, G and P.

- *Signature cost*

The signature algorithm is very fast. It consists in:

- One multiplication of a binary vector of length N by a matrix $N \times N$. Therefore it requires $N^2/2$ binary operations.
- One multiplication of a binary vector of length K by a matrix $K \times K$. Therefore it requires $K^2/2$ binary operations.
- l decoding operations which has a cost of $K2^{k_i}$ binary operations.

- *Verification cost*

The verification algorithm is very fast since it consists in a multiplication of a K length binary vector by a matrix $K \times N$. Therefore, it requires $K \cdot N / 2$ binary operations.

3.5 Performances

Our scheme is determinist to sign and to verify the validity of the signature. Indeed, it permits to sign all messages since the signer, who possesses the secret key (S, G, P), can sign any message after completing BCH$[15,7,2]$ and BCH$[15,5,3]$.

Moreover, our algorithm permits to verify every signatures for every message by anyone possessing the public key G'.

Indeed, the verifier calculates

$$y.G' = y.S.G.P = a.G.P$$

Then, he codes the text a to get a text x'. By multiplying x' by P, the distance between the obtained message and the original text must be inferior to the completed codes correction capacities sum.

In our implementation, we have chosen to sign a binary sequence of length 1000 with BCH $[15,5,c=1]$, BCH $[15,7,c=3]$ and BCH $[15,5,c=5]$ (c is the correction capacity of completed code calculated with (1)).

The obtained signature is of overage length $K = 500$ bits.

The table 1 summarizes the efficiency of our scheme BCHS1 compared with McEliece signature.

Tab 1: Signature with chained BCH code BCHS1 compared to the one based on the one based on McEliece cryptosystem

Signature	McEliece	BCHS1
Data size	65536	1000
Signature length	65392	500
Key size (bit)	4292069312	500000

4. Signature with dual version of chained BCH Code (BCHS2)

As the first scheme, we consider a set of l BCH codes of length $n = 15$ and of dimension $k_i \in \{5,7,11\}$. The set of l BCH codes will be chained to obtain a large code of length $N = \sum_{i=1}^{l} n$ and dimension $K = N - \sum_{i=1}^{l} k_i$.

4.1. Signature parameters

- *Secret parameters*

- A family of l BCH codes.

The parity check matrix of each code is stocked in a large matrix H which we call parity check matrix of chained BCH code (see section 2).

- A binary permutation matrix P over $GF(2)$.
- A binary invertible matrix S over $GF(2)$.

- *Public parameters*

The public parameter used to validate the signature is the public matrix H' deduced from the secret parameters $\{S, H, P\}$. It is a parity check matrix of an equivalent code of chained BCH code.

We need, also, a hash function with an output of length $N - K$.

4.2. Signature algorithm

- *Signing a document*

The message M to be signed is a binary sequence.

- Compute $\rho = h(M)$.
- Compute $a = S^{-1}.P$.
- Split the sequence a in $n - k_i$ length blocs a_i
- If a_i is a syndrome then stock the correspondent error b_i of length n in the message b.

Else modify one bit and try to code, if it is a syndrome then stock the modified bit in a sequence σ' else remodify, again, another bit, etc...

- Compute $y = P^{-1}.b$
- Compute $\sigma = S.\sigma'$

The Signature is formed by (y, σ).

- *Verification algorithm*

In the verification step, we have the message M , the signature (y, σ).

- Compute $\rho = h(M)$.

- Compute $\rho' = H'.y^t$.
- Compute $\varsigma = \sigma + \sigma'$.

The signture is valid if $\varsigma = \rho$

Let's regroup the signature algorithm in a flow diagram:

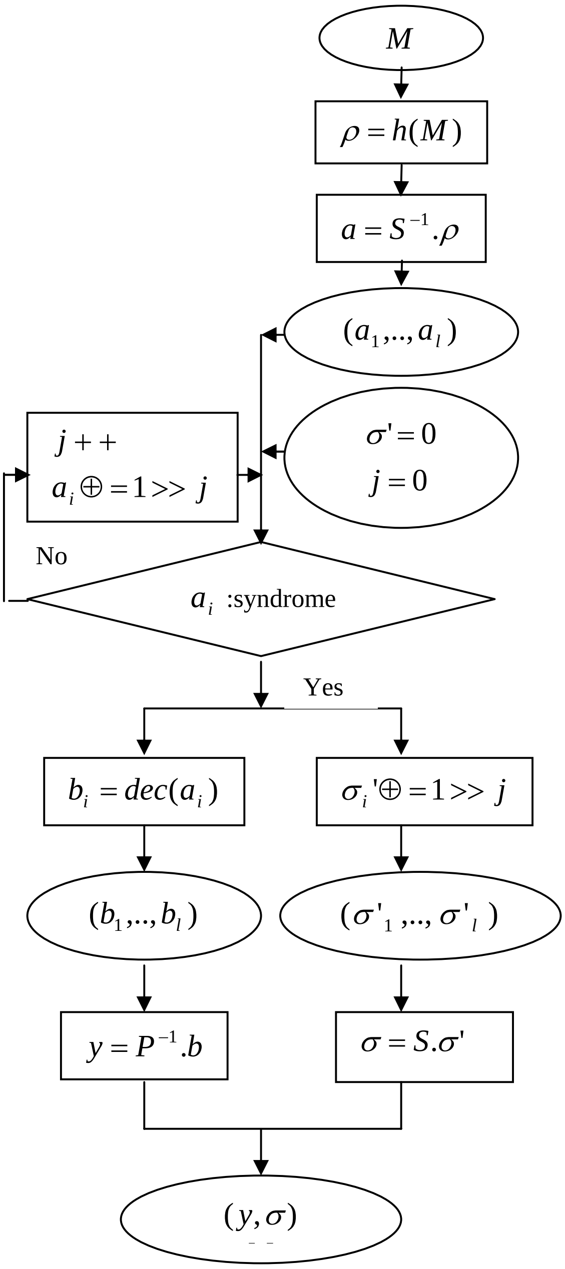

Fig 1: Signature with dual version of chained BCH code

4.3. Implementation aspects

- *Signature cost*

The complexity of the signature is based specially on decoding with the chained BCH code.

The decoding operation is done by blocs using elementary codes. It consists in testing if an $(n-k_i)$ tuples is a syndrome.

The BCH code of length $n=15$ and of dimension $k_i = 11$ correcting $t_i = 1$ error is a perfect code so all $(n-k_i)$ tuples are syndromes.

The BCH code of length $n=15$ and of dimension $k_i = 7$ correcting $t_i = 2$ errors (or of dimension $k_i = 5$ correcting $t_i = 3$ errors) has $C_n^{t_i}$ syndromes.

The probability to have a decodable $(n-k_i)$ tuples is:

$$P = \frac{C_n^{t_i}}{2^{n-k_i}} \approx \frac{1}{2}$$

So, for each bloc of length $n-k_i$ with $k_i \in \{5,7,11\}$ is the dimension of the $i^{'th}$ used code, we need $\frac{n-k_i}{2}$ of random bit modification to obtain a decodable $n-k_i$ tuples. Then, the complexity of signature is:

$$(N-K)(\frac{n-k_i}{2}C_n^{t_i} + (N-K)) + N$$

- *Verification cost*

The verification cost is the complexity of the syndrome determination from the error.

So, the verification requires

$$(c+2)(N-K)$$

binary operations with $c = \sum_{i=1}^{l} c_i$

4.4. Security

There are two broad types of attacks:

- *Decoding attack: Information Set decoding (IS)*

Our schemes are based on the well known problem NP-complete:

Syndrome Decoding (SD)
Instance : Let H be a $(n-k)\times n$ binary matrix , s a binary vector of size $n-k$ and an integer p .
Question : Is there a binary vector x of size n of weight smaller or equal to p such that $Hx^t = s$?

The most efficient algorithms in our case are based on the information set decoding. A first analysis was done by Lee and Brickell in [12], Stern in [13] , Leon in [11] and at last by Canteaut and Chabaud in [14] which is the most efficient one.
Consider a binary code of length n, of dimension k and of correction capacity t , if one uses information set decoding, one chooses a random set of k columns, an error is decodable when its support doesn't meet the k random columns. The probability for an error to be decodable is then $P_{dec} = \frac{C_{n-k}^{t}}{C_{n}^{t}}$.
Then the estimated work factor WF to find a word of weight t can be estimated as follow:

$$WF = \frac{P(k)}{P_{dec}}$$

where $P(k)$ corresponds to the cost of Gaussian elimination, $P(k)$ can be first thought as a cost in $O(k^3)$.

- *Structural attack*

The complexity of this attack on our signature scheme public key can be measured by searching exhaustively for all possible combination of permutation (N !), secret code (3^l) and invertible matrix (0.29×2^K). Then, for each secret key, one has to test wether this key is the good one.
In the case of our scheme, the complexity of this attack can be increased due to the fact that the secret code is formed by three BCH codes. We can apply permutations to the elementary BCH codes to increase their number.

Tab 2 New signatures schemes BCHS1 and BCHS2 compared to the one based on Neiderreiter cryptosystem

Signature	**Neiderreiter**	**BCHS1**	**BCHS2**
Data Size	144	1000	900
Signature length	**132**	500	1350
Key size (bit)	2^{23}	$\mathbf{2^{19}}$	$\mathbf{2^{19}}$
Signature cost	2^{33} *	2^{20} **	$\mathbf{2^{16}}$ **
Verification cost	$\mathbf{2^{10}}$	2^{18}	2^{16}
IS attack workfactor +	2^{80}	2^{88}	2^{85}
Structural attack	2^{119}	2^{106}	2^{95}

*: the document is hashed *t!* times,
**: the document is hashed only one time
+:the IS attack workfactor is the one of the Canteaut and Chabaud algorithm.

The signature BCHS2 has the following proporties compared to the signature based on Niederreiter cryptosystem:
- It has a signature algorithm faster 157000 times,
- It has a public key size smaller 23 times,
- It has a verification algorithm slower 42 times,
- It has a signature longer 10 times

5. A variant that improves the security

In this variant, we will introduce a "random" sequence in the signature to increase the security of the previous scheme. The principle consists in transforming a syndrome to another syndrome by modifying some random bits. As calculated for the previous scheme, the probability to obtain a decodable $n-k_i$ tuples for a code c_i of length n and dimension k_i is 1/2. This modification requires $\frac{n-k_i}{2}$ attempt and if we re-sign the same text, we have a probability of 2^{-60} to obtain the same signature.
The new scheme algorithm is as follow:
- Compute $\rho = h(M)$.
- Compute $a = S^{-1}.\rho$.
- Split the sequence a in $n-k_i$ length blocs a_i

- If a_i is a syndrome then modify it in another syndrome by modifying a random bit and stock the modified bit in a sequence σ' and, then, store the correspondent error b_i of length n in the message b.

Else modify a bit until obtaining a syndrome and transform it to another syndrome then stock the modified bit in the sequence σ'.

- Compute $y = P^{-1}.b$
- Compute $\sigma = S.\sigma'$

The Signature is formed by (y, σ).

The complexity of this algorithm is:

$$(N-K)((n-k_i)C_n^{t_i} + (N-K)) + N$$

which is very close to the one of the previous scheme. With this algorithm, we preserve the same performances and we improve the security.

6. Conclusion

In this paper, we have defined a new signature scheme based on the well known Syndrome Decoding problem SD. Our schemes consist in chaining a family of BCH codes with various dimensions. The resulting matrix of chaining these codes forms the trapdoor of our algorithm.

The main contribution in this paper consists, firstly, in generating a practical digital signature using generator matrix and, secondly, in introducing another scheme faster than the only practical and secure signature scheme based on coding theory.

Our schemes offer, also, a fast and short signature with less public key size than the signature based on McEliece and Neiderreiter public key cryptosystem.

References

[1] W.Deffie and M.E. Hellman, New directions in cryptography, IEEE Transactions on information theory, vol 22, N°.6, pp.644-654 , 1976.

[2] T.ElGamal, A public-key cryptosystem and a signature scheme based on discrete logarithms, Advances in Cryptography, Crypto’84, pp.10-18, 1985.

[3] R.L. Rivest, A. Shamir and L.M. Adleman, A method for obtaining digital signatures and public key cryptosystems, Communications of the ACM, Vol.21, No.2, pp.120- 126, 1978.

[4] E.R. Berlekamp, R.J. McEliece, and H.C.A. van Tilborg, On the inherent intractability of certain coding problems, IEEE Transactions on Information Theory, Vol.24, No.3, pp.384-386 ,1978.

[5] R.J. McEliece. A public-key cryptosystem based on algebraic coding theory; DSN Prog. Rep., Jet Propulsion Laboratory, California Inst. Technol., Pasadena, CA, pp. 114-116,January 1978.

[6] H.Neiderreiter. Knapsack-type crytosystems and algebraic coding theory. Prob. Contr. Inform. Theory, 15(2):157-166, 1986.

[7] S.harari, Anew authentification algorithm, Proceeding of Coding Theory and Applications, Lectures Notes in Computers Sciences 388, pp91-105 , 1988.

[8] N.Courtois, M.Finiasz, N.Sendrier. How to acheive a McEliece based digital signature scheme. Rapport de recherche INRIA n°4118, ISSN 0249-6399, Fevirier 2001.

[9] P. Veron. A fast identification scheme. In IEEE Symposium on Information Theory, Canada, 1995.

[10] Jacques Stern. A new identification scheme based on syndrome decoding. In Douglas R. Stinson, editor, Advances in Cryptology-CRYPTO ’93, volume 773 of Lecture Notes in Computer Science, pages 13-21. Springer-Verlag, 22-26 August 1993.

[11] J.S.Leon, A probabilistic algorithm for computing minimum weights of large error-correcting codes. IEEE Trans. Inform. Theory, IT-34(5), pp. 1354-1359, 1988.

[12] P.J.Lee and E.F. Brikell, An observation on the security of McEliece’s public key cryptosystem, Lecture notes in Computer Science , Advances in Cryptology, Eurocrypt’88, pp. 275-280 , 1989.

[13] J.stern, A method for finding codewords of small weight, Coding Theory and Applications, Lecture Notes in Computer Science 434, pp. 173-180.

[14] A.Canteaut and H.Chabbane, A further improvement of the worfactor in an attempt at breaking Mceliece’s cryptosystem, Proceeding of Eurocode’94, Inria, pp. 163-167.

[15] N. Courtois, M. Finiasz, and N. Sendrier, "How to achieve a McEliece-based digital signature scheme", In C. Boyd, editor, Asiacrypt 2001, volume 2248 of LNCS, pages 157-174. Springer-Verlag, 2001.

Hamdi Omessa$ad was born in 1978 in Gabès, Tunisia. She received the degree in electrical engineering from National School of Engineering of Tunis, Tunisia, in 2002, and the postgraduate research degree in telecommunications, in 2003, from National School of Engineering of Tunis, Tunisia, in 2003.

She is preparing her PHD in National School of Engineering of Tunis, Tunisia, and in South University of Toulon and Var, France. Her actual research interests involves cryptography and network security.