FFM Cryptosystem - Finite Field Matrix Cryptosystem - a public-key cryptosystem with encryption and signature schemes based on the hardness of finite field polynomial factorization

Yuri S Villas Boas

December 17, 2023

Abstract

We introduce a cryptosystem based on the difficulty of factorizing polynomials over finite fields. Public key is given by a full-rank Matrix over this field, and private key by its eigenvector decomposition. Cryptosystem allows for encryption and signature schemes, which are covered in the document, as well as preliminary cryptanalysis.

keywords: finite field polynomial factorization, linear algebra over finite field, characteristic polynomial, eigenvector decomposition, public-key cryptosystem, encryption scheme, signature scheme, cryptanalysis.

1 Algebra

Our proposed cryptosystem can be defined by the table below:

Component	**Formula**	**Definition**
Private Key	(R, A, R^{-1})	$R = (\mathbf{r_1}, \cdots, \mathbf{r_n})$ full-rank, with $n \geq 4$, even $A = (a_1\mathbf{e_1}, \cdots, a_n\mathbf{e_n})$, with $B\mathbf{r_i} = a_i\mathbf{r_i} \neq \mathbf{0}$
Public Key	B	$B = RAR^{-1}$
Plaintext	P	$P \in \mathbb{F}/p^{n \times n}$ an ordered basis over $\mathbb{F}/p^n$
Ciphertext	C	$C = PBP^{-1}$
Decription	VR^{-1}	$V = (\mathbf{v_1}, \cdots, \mathbf{v_n})$, where $(C - \mathbf{r_i}I)\,\mathbf{v_i} = \mathbf{0}$
Document	d	$d \in \mathbb{F}/p$
Signature	s	$s = \Pi_{i=1}^{n/2}(x - a_{\pi(i)}^d) \in \mathbb{F}_p[x]$, where $\pi(\cdot) \in S_n$
Verification	$s \vert b_d$	$b = det\left(B^d - xI\right) \in \mathbb{F}_p[x]$

where,

$\mathbb{F}_p$	refers to the finite field of order p;
$\mathbb{F}_p[x]$	refers to the set of polynomials over $\mathbb{F}_p$;
$(\mathbf{m_1}, \cdots, \mathbf{m_n})$	signifies a matrix having $\mathbf{m_i}$ as its ith column vector;
$\mathbf{e_i}$	refers to the ith canonical vector of $\mathbb{F}_p$;
$p \vert q$	means p divides q;
S_n	is the symmetric group of n elements;

2 Computability

Throughout this session we will analyse the **polynomial-reducibility** of objects defined in the previous session. Here, public information (B, C, s, d) is, in fact, known, that is, available to the algorithms.

Definition 2.1. We define[1] the **non-strict preorder** $a \leq_p b$ as: *"b is polynomially reducible to a." namely "There exists a polynomial-time complexity algorithm to compute b from a."* That definition ensues the following **strict preorder** $a <_p b$ given by $a \leq_p b \wedge a \not\geq_p b$, and the **equivalence relation** $a \equiv_p b$ given by $a \leq_p b \wedge a \geq_p b$.

For the nnnn statements below, consider $f(x) \in \mathbb{Z}_p[x]$, given by $f(x) = \Pi_{i=1}^n (x - a_i)$.

Conjecture 2.2. $f(x) \not\leq_p (a_1, \cdots, a_n)$

Conjecture 2.3. For $f(x) \in \mathbb{Z}_p[x]$, given by $f(x) = \Pi_{i=1}^n (x - a_i)$, with $n \geq 2$, $(a_1, \cdots, a_n) \leq_p f(x)$.

Proof. More precisely, we will

($\leq_p$): Just perform the multiplications of the n binomials $(x - a_i)$, totaling $\mathcal{O}(n^2)$ multiplications in $\mathbb{F}_p$.

($\not\geq_p$): Conjectured hardness of polynomial factorization.

QED

Lemma 2.4. $R \equiv_p A$

Proof. linear systems:

($\leq_p$): Have $a_i := (\mathbf{e_{j(i)}}^T B \mathbf{r_i}) * (\mathbf{e_{j(i)}}^T \mathbf{r_i})^{-1}$, for each i and any $j(\cdot)$ for which $\mathbf{e_{j(i)}}^T \mathbf{r_i} \neq 0$ — which is guaranteed to exist by R being full-rank. Namely, use the definition of eigenvectors to calculate each eigenvalues by taking one non-zero entry of $B\mathbf{r_i}$ and finding the 'ratio' of it to the same (non-null) entry of $\mathbf{r_i}$ (meaning, multiply the former by the later's multiplicative inverse in $\mathbb{F}_p$).

($\geq_p$): For each $1 \leq i \leq n$, solve $(B - a_i I)\mathbf{x} = \mathbf{0}$ on $\mathbf{x_i}$ to find $\mathbf{r_i}$ except by a constant scalar factor. Cancel that factor after assertaining it, likewise in the previous step, from the 'ratio' of any non-null entry of $\mathbf{e_i}^T B$ and the correspondent (non-null) entry of $\mathbf{x_i}$.

QED

Conjecture 2.5. Probability of fortuitous ocurrence of any entry whatsoever equal 0 is polynomially bounded by $1/p$.

[1] 'p' in $\leq_p$ stands for *polynomial time*, while 'p' in $\mathbb{F}_p$ is an incognate *prime* number. Those are two totally different concepts and they sharing the same letter is just a notation coincidence.

Intuitively, it seems to be the case that assuming the random elicitation of the private key is, in fact, uniform, and that the hashing algorithm of documents is ideal, and that the specification of vectors of the eigenspace for the plaintext is uniformly random, and that the plaintext itself is uniforrmly random[2], the probabilty of any given entry of the **public key**, a **signature** and a **ciphertext** being 0 is bounded to a power of $1/p$ per instance of usage. Anecdotal observations suggest that such fortuitous occurrences 0's for realistic values of p are, in fact, negligible.

Observation 2.6. One relevant implication of conjecture 2.5 being true, the treatment such possibility is just a matter of formal mathematical correctness, and implementations of this algorithms could be made to simply ignore them, as merely contemplating them could cost more than blindly taking the risk run-time failure, however catastrophic the consequences.

Observation 2.7. Another relevant implication conjecture 2.5 being true is that it to be the case that the difficulty hazardly yielding a 0 entry as described above could be used as a cryptographic scheme of some sort.

Observation 2.8.

3 Complexity

4 Cryptanalysis

[2] That is actually very plausible seeing that public key encryption schemes typically are used to encrypt symmetric session keys represented by arrays of random bits.