Efficient Zero-knowledge Authentication Based on a Linear Algebra Problem MinRank

Nicolas T. Courtois[1,2,3]
courtois@minrank.org

[1] CP8 Crypto Team, SchlumbergerSema, BP 45
36-38 rue de la Princesse, 78430 Louveciennes Cedex, France
[2] SIS, Toulon University, BP 132, F-83957 La Garde Cedex, France
[3] Projet Codes, INRIA Rocquencourt, BP 105, 78153 Le Chesnay - Cedex, France

Official web page: http://www.minrank.org/minrank/

Abstract. A Zero-knowledge protocol provides provably secure entity authentication based on a hard computational problem. Among many schemes proposed since 1984, the most practical rely on factoring and discrete log, but still they are practical schemes based on NP-hard problems. Among them, the problem SD of decoding linear codes is in spite of some 30 years of research effort, still exponential. We study a more general problem called MinRank that generalizes SD and contains also other well known hard problems. MinRank is also used in cryptanalysis of several public key cryptosystems such as birational schemes (Crypto'93), HFE (Crypto'99), GPT cryptosystem (Eurocrypt'91), TTM (Asiacrypt'2000) and Chen's authentication scheme (1996).
We propose a new Zero-knowledge scheme based on MinRank. We prove it to be Zero-knowledge by black-box simulation. An adversary able to fraud for a given MinRank instance is either able to solve it, or is able to compute a collision on a given hash function.
MinRank is one of the most efficient schemes based on NP-complete problems. It can be used to prove in Zero-knowledge a solution to any problem described by multivariate equations. We also present a version with a public key shared by a few users, that allows anonymous group signatures (a.k.a. ring signatures).

Key Words:
Zero-knowledge, identification, entity authentication, MinRank problem, NP-complete problems, multivariate cryptography, rank-distance codes, syndrome decoding (SD), group signatures, ring signatures.

Acknowledgments

I would like to thank prof. Ernst M. Gabidulin, prof. Jacques Patarin, prof. Claus P. Schnorr and dr. Louis Goubin for helpful remarks.
The work described in this paper has been supported by the French Ministry of Research under RNRT Project "Turbo-signatures".

1 Introduction

The general problem we address is the classical problem of interactive entity authentication. It is known since Fiat-Shamir [5] that solving this problem combined with a cryptographic hash function also allows non-interactive authentication, for example digital signatures,

The notion of Zero-knowledge identification has been formalized by Goldwasser, Micali and Rackoff in [18]. In such a scheme a Prover proves his identity to a Verifier. Provided the underlying problem is difficult, we prove that there is no interactive strategy for the Verifier communicating with the Prover, to extract any information whatsoever on the prover's secret. Several such schemes have been proposed since the original Fisher-Micali-Rackoff scheme (1984), and the most practical ones are Fiat-Shamir, Guillou-Quisquater and Schnorr schemes. Unfortunately they rely on problems that are (believed) not NP-hard such as factoring or discrete log. Still there are schemes using an NP-hard problem and still practical, for example PKP by Shamir [31], CLE by Stern [35] or PPP by Pointcheval [27]. However the most interesting schemes are in our opinion the schemes related to coding, as the decoding problem(s) are believed intractable even since the 1970s [2]. There were many proposals [34, 40, 20, 16, 4] and the best of them is the scheme SD by Stern [34, 40]. The simplest decoding problem is the problem of Syndrome Decoding (SD) and consists of finding a small weight vector in an affine subspace of a linear space. Similarly the MinRank problem is a problem of finding a linear (or affine) combination of given matrices that has a small rank. Both problems are NP-hard. Moreover SD have withstood more than 20 years of extensive research on the cryptanalysis of the McEliece cryptosystem [22] and all the known attacks for SD are still exponential, [1, 3, 21, 36, 40]. MinRank in fact contains SD and thus is also probably exponential. It also contains the decoding problem for rank-distance codes of Gabidulin, used in public-key authentication scheme of Chen [4] cryptanalysed in [37, 11], and also used in the public-key encryption scheme GPT [14]. The MinRank problem, not always named so, has many applications in cryptanalysis of various schemes such as Shamir's birational schemes [30, 6, 7] cryptanalysed by Coppersmith, Stern and Vaudenay solving a MinRank with a small rank. Similarly Goubin and Courtois broke the TTM cryptosystem in [19]. In [32] Shamir and Kipnis reduced the cryptanalysis of Hidden Field Equations (HFE) scheme [24] to MinRank.

In the present paper we present a new Zero-knowledge protocol, for MinRank. More precisely we show have to prove in Zero-knowledge an ability to compute (or have) MinRank solutions. We may build instances that have only one solution, and for those it will also be a proof of knowledge. We show that the scheme can also be applied to prove in Zero-knowledge a solution to **any** other problem expressed as a system of multivariate equations over a finite field.

The paper is organized as follows: First we recall the basic requirements of a Zero-knowledge protocol. Then in §3 defines MinRank and studies related hard problems. The §4 shows how to build secure instances for practical use, evaluated with all the 5 attacks currently known for MinRank. In the §5 we describe key generation and setup of the MinRank identification which is described in §6. The following §7 gives proofs of completeness, soundness and Zero-knowledge. Then in §8 we analyse the performance of the scheme and in §8.2 we compare it to other schemes based on NP-complete problems. In Appendix 1 we compute useful probability distributions for ranks of matrices. The Appendix 2 contains various practical improvements to the scheme, notably reducing the fraud probability form 2/3 to 1/2. Finally, the Appendix 3 shows that MinRank allows to achieve authentication and signature, for any small subgroup, of a given group of users sharing the same public key.

2 Zero-knowledge Protocols

An interactive protocol involves two entities/strategies: the Prover (P) and the Verifier (V) that will be two probabilistic Turing machines. The Verifier and prover interact and at the end the Verifier gives an answer: Accept or Refuse.

In known Zero-knowledge protocols, there is a possibility of fraud: a cheater is usually able to answer to some types of questions (for which he was prepared in advance) but not for all of them. The protocols are designed in such a way that an answer to one question gives no information (Zero-knowledge), while answering all the questions is proved to reveal Prover's secret (Soundness). The security is in fact based on the impossibility by the Prover to predict Verifier's questions. If we iterate the protocol, the global fraud probability becomes then as small as we want.

A Zero-knowledge identification scheme should be: **complete**, **sound** and **Zero-knowledge**:

Completeness. The legitimate Prover gets always accepted.

(Computational) Soundness. An illegitimate Prover will be rejected with some fixed probability. We usually show the Prover that always succeeds can be used to extract the Prover's secret (a knowledge extractor).

Zero-knowledge. It is much stronger that saying the Verifier learns merely nothing about the secret. We demand that no Verifier strategy, can extract any information from the Prover, even in several interactions. It gives provable security against active attacks. Proofs are made by simulation using the Verifier as an oracle, or black-box, and therefore this definition has been called black box (computational) Zero-knowledge, as formalized by Goldreich, and Oren [17]:

Definition 2.0.1 (Black box Zero-knowledge, [17]). A strategy P is told to be black box Zero-knowledge on inputs from S (common input) if there exists an efficient simulating algorithm U so that for every feasible Verifier strategy V, the two following probability ensembles are computationally indistinguishable:

– $\{(P,V)(x)\}_{x\in S} \stackrel{def}{=}$ all the outputs of V when interacting with P on a common input $x \in S$.
– $\{U(V)(x)\}_{x\in S} \stackrel{def}{=}$ the output of U using V as a black box, on $x \in S$.

The definition above is strong and still realistic: all well-known Zero-knowledge protocols are proven in this model.

3 The MinRank Problem

Let M_0; $M_1, \ldots, M_m$ be some $\eta \times n$ matrices over a ring R. The problem MinRank(η, n, m, r, R) is to find a solution $\alpha \in R^m$ such that:

$$Rank(\sum_i \alpha_i M_i - M_0) \leq r.$$

3.1 Related Problems

This version of the MinRank, is a generalized version of one among many NP-complete rank problems studied in [23] and [10]. In our scheme R will be a finite field $GF(q)$.

MinRank over a field can be defined in terms of codes: it is a decoding problem for a kind of subfield subcode of Gabidulin's linear rank-distance code over $GF(q^\eta)$ [13, 11, 37]. Currently one of the two best known attacks to decode rank distance codes is based on MinRank [11, 37]. Therefore MinRank is essential to the security of Chen and GPT public key schemes [14, 4, 11]. MinRank also appears in attacks known on the HFE [32, 8, 10], TTM cryptosystem [19] and Shamir's birational signature scheme [30, 6, 7]. Finally, as we show in §3.3, MinRank contains the SD problem for ordinary codes that underlies the security of McEliece [22] and various identification schemes [34, 40, 16, 20].

MinRank over rings should also be mentioned. MinRank over $\mathbb{Z}$ might be broken by the widely-used LLL algorithm. Indeed, when all the M_i are diagonal of size up to 300×300, the problem is to find a vector in a lattice with a small number of non-zero elements, and this problem is closely related to the well known lattice reduction problem that has numerous applications in cryptography. Still MinRank over $\mathbb{Z}$ is undecidable in general, because it can encode any set of diophantine equations (Tenth Hilbert's problem) [23].

3.2 Encoding NP Problems as MinRank

The problem of proving in Zero-knowledge that a system of equations over a finite field has a solution has already been solved in [12] under RSA **or** DL intractability. Our solution is based on an NP-complete problem.

Theorem 3.2.1 (Determinant Universality, Valiant 1979). Any set of multivariate equations over a ring can be encoded as a determinant of a matrix with entries being constants or variables.

It was first shown by Valiant [38]. For a simpler, and still effective proof see [23]. Both give an **effective** algorithm to encode any set of multivariate polynomial equations as a MinRank. However the size of matrices it gives seems hard to improve, for m equations of degree d with n variables we need matrices of width about mn^d.

From now we always suppose that $R = GF(q)$. Solving multivariate quadratic equations over a field is NP-hard [26], thus:

3.3 MinRank is NP-hard

The proof of [23], however, gives instances of MinRank in which the size of the matrices will be polynomial in the number of matrices. It might seem that MinRank is less secure with m matrices $n \times n$ and m and n being of the same order of magnitude. We are going to show a reduction from an NP-complete problem that gives instances that are known to be hard both in theory in practice, with m, n and r being of the same order of magnitude. We reduce from the Syndrome Decoding problem of a linear error correcting code that is NP-complete. The proof for the case $q = 2$ is to be found in [2], and an extension to the arbitrary field is sketched in [39], page 1764. Let (n, k, d) be an error correcting code. The encoding is trivial: each of the lines of the generating matrix will be put on the diagonal of a $n \times n$ matrix M_i that will have all 0's elsewhere. Similarly M_0 contains the fixed codeword to decode. Solving MinRank with rank r is then equivalent to correcting r errors.

4 MinRank Instances and Attacks

4.1 Preliminary Requirement

The instance of MinRank should be chosen in such a way that the probability it has many solutions (apart from those we might put by construction) should be small. One possible way of achieve this is an explicit reduction from an instance of another problem that has only one solution, as for example in §3.2.

Another way is to choose parameters such that the probability it has a solution, given in Appendix 2, is small, and thus we will be able to build instances with one (constructed) solution that are unlikely to have (m)any more. In this case, as we show in section 10, we need to have

$$m \leq m_{max} \quad \text{with} \quad m_{max} \stackrel{def}{=} \eta n + r^2 - (\eta + n)r + 1$$

4.2 Known Attacks

We assume $\eta \geq n$. [1] There are five attacks known for the problem MinRank. Let ω be the exponent of the Gaussian reduction $2 \leq \omega < 3$, in practice $\omega \simeq 3$.

Exhaustive search. It is $q^m r^\omega$, see [10] for details.

Attacking square MinRank with $r \approx n$. In some cases the exhaustive search may break MinRank [2]. For example we consider a MinRank with m matrices $n \times n$ and with $r = n - s$. Then we have $m_{max} = r^2 + n^2 - 2nr = s^2$. A randomly generated MinRank with such parameters can be solved in about q^{s^2}, which can be quite small. However if the MinRank with $m >> m_{max}$ is generated from a reduction from another problem (see §3.2) having not too many solutions, it is still secure.

[1] The problem is symmetric with respect to transposition of matrices with swapping η and n and by inspection we verify that all the complexities given in the present paper are already given for the better of the two cases.

[2] This attack was suggested to me by prof. Claus P. Schnorr.

Attack Using Sub-matrices. This simple attack works **only if** $r << n$, not the case in this paper, and was first used by Coppersmith, Stern and Vaudenay in [6, 7]. It was then described in details and used in [8, 9].

MQ-solving Attacks. Another attack that works only for $r << n$ is due to Shamir and Kipnis [32]. It reduces MinRank to the MQ problem, i.e. to a system of **M**ultivariate **Q**uadratic equations. If $r << n$ the system is overdefined, and surprisingly such a system will be solved in expected polynomial time [32]. Improved algorithms will give roughly about $n^{\mathcal{O}(r)}$, see [10, 11, 8, 33].

Since we will never have $r << n$, both these attacks fail.

The Kernel Attack. is the best attack for the parameter sets we propose. It is due to Louis Goubin and described in [19] with a complexity of $q^{\lceil \frac{m}{n} \rceil r} m^{\omega}$ for $n = \eta$. A more general version described in [10] and [11] gives

$$Min\left(\quad q^{\lceil \frac{m}{n} \rceil r} \quad , \quad q^{\lfloor \frac{m}{n} \rfloor r + (m \bmod n)} \quad \right) \cdot m^{\omega}.$$

For small r there are further improvements described in [11].

The "Big m" Attack This attack designed for $m >> n$ and is described in [11] and [10]. It is trivial and consists of constraining as many entries of the matrix M, as possible to 0. It runs in

$$q^{Max(0, \eta(n-r)-m)} \left(\eta(n-r) \right)^{\omega}.$$

The Syndrome Attack. Another attack for $m >> n$ and is described in [11] and [10]. It is not very practical and gives about

$$q^{Max\left(\frac{\eta n - m - 1}{2}, (\eta+n)r/2 - m - r^2/4\right)} \cdot \mathcal{O}(r\eta n)$$

Hard Instances: All the attacks known for MinRank described above are exponential in general. In a work in progress, [11] it is conjectured that for fixed $\eta = n$ the best security of $q^{\frac{4}{27}n^2}$ is achieved with $r = n/3$ and $m \approx \frac{4}{9}n^2$. If m is fixed, one may also build instances as close as we want to the exhaustive search if we put $n > 3\sqrt{m}$ and as big as possible, and with $r = n - \sqrt{m}$.

4.3 Practical Parameter Choices

We propose six sets of parameters A-F that use square matrices ($\eta = n$) and work either over $GF(2)$ or over $GF(65521)$, the biggest prime that fits in 16 bits. In the following table we compare the complexity of all known attacks described above for A-F, give the communication complexity computed following §12, as well as the probability that it has a solution computed computed in §10.
For comparison we also include two MinRank instances that appear in the Shamir-Kipnis attack on HFE cryptosystem [3] [32], given for the HFE Challenge 1 [24, 9] and for a subsystem of Quartz [4] [25].

Cryptosystem	MinRank identification						HFE	
Parameter set	A	B	C	D	E	F	Chall. 1	Quartz
m	10	10	10	81	121	190	80	103
n	6	7	11	19	21	29	80	103
η	6	7	11	19	21	29	80	103
r	3	4	8	10	10	15	7	8
q	65521	65521	65521	2	2	2	2^{80}	2^{103}
$Pr_\alpha[Rank \leq r]$	0.6	0.6	0.6	0.6	0.6	2^{-6}	$< 2^{-10^5}$	
20×Comm. [Kb]	1.94	2.99	4.86	2.17	2.36	3.13		
Attack								
Brute force	2^{168}	2^{168}	2^{170}	2^{81}	2^{134}	2^{205}	2^{80}	2^{103}
Kernel	$\mathbf{2^{106}}$	$\mathbf{2^{122}}$	$\mathbf{2^{138}}$	$\mathbf{2^{64}}$	$\mathbf{2^{81}}$	$\mathbf{2^{128}}$	2^{577}	2^{844}
Big m	2^{108}	2^{205}	2^{399}	2^{113}	2^{135}	2^{243}	2^{461k}	2^{997k}
Syndrome	2^{118}	2^{312}	2^{1002}	2^{151}	2^{172}	2^{339}	2^{252k}	2^{530k}
Sub-matrices	∞	∞	∞	∞	∞	∞	$\mathbf{2^{97}}$	$\mathbf{2^{114}}$
MQ	∞	∞	∞	∞	∞	∞	2^{152}	2^{188}

5 Setup of MinRank Identification

5.1 Key Setup

The public key are $1 + m$ matrices $\eta \times n$ over a finite field $GF(q)$, $M_0;\ M_1, \ldots, M_m$. Let $r < n$. To generate a random hard [5] instance we pick $1 + m - 1$ (pseudo-)random matrices $M_0;\ M_1, \ldots, M_{m-1}$. We chose a random

[3] The brute force factors given in the table for HFE correspond to the direct brute force attack on HFE itself, not on MinRank that would give much more.

[4] Since it is only a subsystem, an attack on MinRank does not really break Quartz.

[5] The instances of MinRank generated here are such that the matrices, and a linear combination that yields a small rank, are all random and uniformly distributed. It is believed to give hard instances most of the time with respect to all the attacks from section 4.2. It might change if a better way to produce hard instances is known. The same problem is an issue for **any** cryptosystem based on an NP-complete problem: there is a difference between an NP-complete problem in general, and the actual instances in the samplable distribution generated by a finite-length algorithm.

M of rank r and we "adapt" M_m. For this we pick a random $\alpha \in GF(q)^m$ such that $\alpha_m \neq 0$ and M_m is computed as:

$$M_m = (M + M_0 - \sum \alpha_i M_i)/\alpha_m$$

In practice, we generate M and $M_1, \ldots, M_{m-1}$ out of a pseudo-random generator with a seed of 160 bits. It is better to pick all M_i invertible, but it's not necessary. We may use the well-known LU method [6] to generate a deterministic pseudo-random invertible matrix. In order to generate M, first we generate a matrix L which is random invertible matrix $r \times r$, completed with 0's to an $\eta \times n$ matrix. Then a random couple of invertible matrices S and T is applied $M = SLT$, see Lemma 7.0.1.

The secret key It is the solution $\alpha \in GF(q)^n$ such that

$$Rank(\sum \alpha_i \cdot M_i - M_0) = r.$$

Key sizes All the public key is generated out of a pseudo-random generator with a seed of 160 bits, except M_m that is transmitted. The size of the public key is thus only 160 + $n\eta \log_2 q$ bits. The secret key requires only additional $m \log_2 q$ bits to store α.

6 MinRank Identification Scheme

We use a collision-intractable one-way hash function H for commitments that is supposed to be behave as a random oracle. The Prover is going to convince the Verifier of his knowledge of α (and M).

The Prover chooses two random invertible matrices S, T that are $\eta \times \eta$ and $n \times n$, and a totally random $\eta \times n$ matrix X. We call $\overline{STX}$ the triple (S, T, X). Then, he picks a random combination β_1 of the M_i:

$$N_1 = \sum \beta_{1i} \cdot M_i$$

He puts and $N_2 = M + M_0 + N_1$ and uses his secret expression of M to get:

$$N_2 = \sum \beta_{2i} \cdot M_i$$

We have $\beta_2 - \beta_1 = \alpha$, but each of β_i (taken separately) is random and uniformly distributed. Each of the N_i is just a random combination of the M_i.

[6] This method is known to give a slight bias, but it seems easy tor repair for example by multiplying a few such matrices and permuting columns.

One Round of Affine MinRank Identification:

1. The Prover sends to the Verifier:

$$\xrightarrow{\hspace{6cm}}$$
$$H(\overline{STX}),\ H(TN_1S+X),\ H(TN_2S+X-TM_0S)$$

2. The Verifier chooses a query $\mathcal{Q} \in \{0,1,2\}$ and sends $\mathcal{Q}$ to the Prover.

$$\xleftarrow{\hspace{6cm}}$$
$$\mathcal{Q} \in \{0,1,2\}$$

3. If $\mathcal{Q} = 0$ the Prover gives the following values:

$$\xrightarrow{\hspace{6cm}}$$
$$(TN_1S+X),\ (TN_2S+X-TM_0S)$$

Verification $\mathcal{Q} = \mathbf{0}$: The Verifier accepts if
$H(TN_1S+X) and H(TN_2S+X-TM_0S)$ are correct and if

$$(TN_2S+X-TM_0S)-(TN_1S+X)=TMS$$

is indeed a matrix of rank r.

3' If $\mathcal{Q} = 1,2$ the Prover reveals:

$$\xrightarrow{\hspace{6cm}}$$
$$\overline{STX},\ \beta_{\mathcal{Q}}$$

Verification $\mathcal{Q} = \mathbf{1,2}$: The Verifier checks if S and T are invertible and $H(\overline{STX})$ is correct. Then he computes

$$TN_{\mathcal{Q}}S = \sum \beta_{\mathcal{Q}i}\ \ TM_iS$$

and verifies $H(TN_1S+X)$ or $H(TN_2S+X-TM_0S)$.

6.1 Completeness

It is clear that a legitimate Prover that knows α always succeeds.

6.2 Soundness

We will show that a false Prover is rejected with probability $\frac{1}{3}$. Let C (Charlie or the Cheater), be an expected polynomial time Turing machine. We suppose that there is such a false Prover C that can answer all the questions $\mathcal{Q}$. In fact the proof below shows that such a Prover **will either be able to compute a collision for H, or be able to solve the given instance of the NP-complete problem MinRank** [7].

[7] Here it can be just any instance of MinRank, however in the practical authentication the public key is generated in a specific way, see note 1 on the bottom of page 7.

Proof: C commits (with H) to the values of TN_1S+X and TN_2S+X. For $\mathcal{Q}=1$ and 2 he proves that he has indeed generated them in the form $X+T(\sum\beta_{1i}M_i)S$ and $X+T(\sum\beta_{2i}M_i)S$. In both cases we verify $H(\overline{STX})$ and we are certain that he used the same X, S and T. Finally when $\mathcal{Q}=0$ we will verify the rank of the following matrix is indeed r:

$$\Big(T(\sum\beta_{2i}M_i)S - TM_0S + X\Big) - \Big(T(\sum\beta_{1i}M_i)S + X\Big) =$$
$$= \sum_{i=1}^{m}(\beta_{2i}-\beta_{1i})\cdot TM_iS - TM_0S$$

When $\mathcal{Q}=1$ or 2 we check that S and T are invertible, thus

$$\sum_{i=1}^{m}(\beta_{2i}-\beta_{1i})\cdot M_i - M_0$$

is also of rank r. Thus the Prover knows a solution to MinRank $\alpha=(\beta_2-\beta_1)$, i.e. either the secret key α or an equivalent one. □

One can see that the fraud probability for several rounds is:

$$Pr_{fraud} = \left(\frac{2}{3}\right)^{\#\text{rounds}}.$$

For details and an improvement to $\left(\frac{1}{2}\right)^{\#\text{rounds}}$ see 11.2 and 11.3.

7 Black-box Zero-knowledge of MinRank

Let the Prover strategy P be a probabilistic average polynomial time Turing machine. We suppose that H is a random function (oracle).The simplicity of MinRank makes very easy to show it is Zero-knowledge.

- In cases $\mathcal{Q}=1,2$ we only disclose random unrelated variables S, T, $\beta_{\mathcal{Q}}, X$.
- The case $\mathcal{Q}=0$: disclosing (TN_1S+X) and (TN_2S-TM_0S+X) is equivalent to disclosing (TN_1S+X) and their difference $TN_2S-TM_0S-TN_1S=TMS$.

  Since X is completely random, (TN_1S+X) is a random matrix independent from TMS. As for TMS, we show that it is a uniformly distributed matrix of rank r:

Lemma 7.0.1. Let M be a $\eta\times n$ matrix of rank r. Let S and T be two uniformly distributed random invertible matrices $\eta\times\eta$ and $n\times n$. Then TMS is uniformly distributed among all $\eta\times n$ matrices of rank r.

Proof sketch: All the $\eta\times n$ matrices M of rank r are equivalent modulo invertible variable changes and can be written as:

$$M = S'\cdot\begin{pmatrix} Id_{r\times r} & 0_{r\times(n-r)} \\ 0_{(\eta-r)\times r} & 0_{(\eta-r)\times(n-r)}\end{pmatrix}\cdot T'$$

7.1 The Exact Proof of Zero-knowledge by Simulation

We construct a simulator U with oracle access to V, see Def. 2.0.1:

1. $U(V)$ chooses a random query $\mathcal{Q} = 1, 2$. He will prepare to answer to questions 0 **and** $\mathcal{Q}$.
2. He chooses $N = \sum \delta_i M_i$ with a random δ.
3. He picks up $\overline{STX} = (S, T, X)$ with invertible S and T.
4. He picks up a random matrix R of rank r.
5. Let $N_{\mathcal{Q}} = N$ and $N_{3-\mathcal{Q}} = N + (-1)^{\mathcal{Q}+1}(R + M_0)$. Now $N_2 - N_1 = R + M_0$.
6. He asks for Verifier's query on his commitment:
$$\mathcal{Q}' = V\left(H(\overline{STX}),\ H(TN_1S + X),\ H(TN_2S - TM_0S + X)\right) \in \{0, 1, 2\}.$$
7. He repeats steps 1-6 about 2 times (**rewinding**), until he does get one of the two queries he has prepared to answer:
$$\mathcal{Q}' \in \{0, \mathcal{Q}\}$$
8. If $\mathcal{Q}' = 0$ the simulator $U(V)$ reveals $(TN_2S + X - TM_0S)$ and $(TN_1S + X)$ with indeed a difference TRS of rank r.

8' If $\mathcal{Q}' = \mathcal{Q}$ the simulator $U(V)$ reveals $\overline{STX}$ and δ, that were indeed used to construct the committed $TN_{\mathcal{Q}}S + X[-TM_0S]$.

8 Performance of the Scheme

8.1 Communication Complexity

We assume that hash values are computed with SHA-1. Thus we need $3 \cdot 160 + 2$ bits for the first two passes.
We note that the values of $\overline{STX} = (S, T, X)$ does not need to be transmitted, they are in practice generated using a pseudorandom generator out of a seed of 160 bits, using the method we described in §5.1 to generate pseudorandom invertible matrices S and T. [8]
The last pass requires $2n\eta \log_2 q$ bits in the case $\mathcal{Q} = 0$. In the two other cases it requires $160 + m \log_2 q$ bits. The weighted average bit complexity for the whole scheme is $3 \cdot 160 + 2 + \frac{2}{3} \cdot 160 + \frac{2}{3}(n\eta + m) \log_2 q$.

This is to be multiplied by the number of rounds which is ≥ 35 for the round fraud probability of 2/3. In the Appendix 3 we show how achieve 1/2 instead (which will require only 20 rounds) and present several other improvements. Our best scheme (cf. 12 and 12.1) gives a communication complexity as low as :

$$\text{Comm. [in bits]} = 2 \cdot 160 + \left(4 \cdot 160 + 8 + \frac{n\eta + m}{2} \log_2 q\right) \cdot \#\text{rounds}$$

[8] Such modifications make the security depend on an additional assumption. It seems to be a quite weak and plausible assumption. For example here (S, T, X) should be indistinguishable from random.

8.2 Comparison with Other Schemes

The following table compares different Zero-knowledge protocols based on NP-complete problems based on previous work of Pointcheval [28].

	PKP Shamir	SD Stern	Chen [4] Chen	CLE Stern	PPP Pointcheval	MinRank (A) Author
matrix	16 x 34	256 x 512	32 x 16	24 x 48	101 x 117	6 x 6
field	$\mathbb{F}_{251}$	$\mathbb{F}_2$	$\mathbb{F}_{65535}$	$\mathbb{F}_{257}$	$\mathbb{F}_2$	$\mathbb{F}_{65521}$
passes	5	3	5	3/5	3/5	3
impersonation probability	$\frac{1}{2}$	$\frac{2}{3}$	$\frac{1}{2}$	$\frac{2}{3}/\frac{1}{2}$	$\frac{3}{4}/\frac{2}{3}$	$\frac{2}{3}/\frac{1}{2}$
rounds	20	35	20	35/20	48/35	35/20
impersonation global	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
public key [bits]	272	256	256	80	149	735
secret key [bits]	128	512	512	80	117	160
best attack	2^{60}	2^{70}	2^{53}	2^{73}	2^{61}	$\mathbf{2^{106}}$
bits send/round	665	954	1553	940/824	896/1040	1075/694
global [Kbytes]	**1.62**	4.08	3.79	4.01/**2.01**	5.25/4.44	4.6/**1.94**

9 Conclusion and Perspectives

We described a new MinRank authentication scheme. It is proven Zero-knowledge and relies on a linear algebra problem MinRank. This NP-hard problem contains in a very natural way some famous problems such as Syndrome Decoding. Both these problems are believed hard on average and all the known algorithms are exponential.

It is possible to use MinRank to prove in Zero-knowledge a knowledge of a solution for any problem expressed as a set of multivariate equations over a finite field (see 3.2). However, the encoding will not always be practical.

Among known schemes based on NP-complete problems MinRank is one of the most efficient, though several schemes are not much worse.

MinRank also allows to share the public key among several users in such a way that any small subgroup can identify itself or produce signatures.

References

1. Alexander Barg: *Handbook of coding theory, Chapter 7: Complexity Issues in Coding Theory;* North Holland, 1999.
2. E.R. Berlekamp, R.J. McEliece, H.C.A. van Tilborg: *On the inherent intractability of certain coding problems*; IEE Trans. Inf. Th., IT-24(3), pp. 384-386, May 1978.
3. Anne Canteaut, Florent Chabaud: *A new algorithm for finding minimum-weight words in a linear code: application to McEliece's cryptosystem and to BCH Codes of length 511;*

4. Kefei Chen: *A new identification algorithm.* Cryptography Policy and algorithms conference, vol. 1029, LNCS, Springer-Verlag, 1996.
5. Amos Fiat, Adi. Shamir: *How to prove yourself: Practical solutions to identification and signature problems.* In Advances in Cryptology, Crypto '86, pp. 186-194, Springer-Verlag, 1987.
6. Don Coppersmith, Jacques Stern, Serge Vaudenay: *Attacks on the birational permutation signature schemes*; Crypto 93, Springer-Verlag, pp. 435-443.
7. Don Coppersmith, Jacques Stern, Serge Vaudenay, *The Security of the Birational Permutation Signature Schemes*, in Journal of Cryptology, 10(3), pp. 207-221, 1997.
8. Nicolas Courtois: *The security of Hidden Field Equations (HFE)*; Cryptographers' Track Rsa Conference 2001, San Francisco 8-12 April 2001, LNCS2020, Springer-Verlag.
9. The HFE cryptosystem home page: `http://hfe.minrank.org`.
10. Nicolas Courtois: *The security of cryptographic primitives based on multivariate algebraic problems: MQ, MinRank, IP, HFE*; PhD thesis, September 25th 2001, Paris 6 University, France. Mostly in French. Available at `http://www.minrank.org/phd.pdf`
11. Nicolas Courtois and Ernst M. Gabidulin.: *Security of cryptographic schemes based on rank problems*; work in progress.
12. Ronald Cramer, Ivan Damgård: *Zero-Knowledge Proofs for Finite Field Arithmetic or: Can Zero-Knowledge be for Free?* Crypto'98, LNCS 1642, pp. 424-441, Springer Verlag. See `http://www.brics.dk/RS/97/27/`
13. Ernst M. Gabidulin. *Theory of codes with maximum rank distance.* Problems of Information Transmission, 21:1-12, 1985.
14. Ernst M. Gabidulin, A. V. Paramonov, O. V. Tretjakov: *Ideals over a Non-Commutative Ring and their Applications in Cryptology. Eurocrypt 1991, pp. 482-489.*
15. *Ernst M. Gabidulin, Alexei V. Ourivski: Modified GPT PKC with Right Scrambler.* WCC 2001, Paris, France, Daniel Augot and Claude Carlet Editor.
16. Marc Girault: *A (non-practical) three pass identification protocol using coding theory;* Advances in cryptology, AusCrypt'90, LNCS 453, pp. 265-272.
17. Oded Goldreich, Y. Oren. Definitions and properties of Zero-knowledge proof systems. Journal of Cryptology 1994, vol.7, no.1, pp.1-32.
18. S. Goldwasser, S. Micali and C. Rackoff, *The knowledge Complexity of interactive proof systems*; SIAM Journal of computing, 1997, Vol. 6, No.1, pp.84.
19. Louis Goubin, Nicolas Courtois *Cryptanalysis of the TTM Cryptosystem*; Advances of Cryptology, Asiacrypt'2000, 3-9 December 2000, Kyoto, Japan, Springer-Verlag.
20. Sami Harari. *A new authentication algorithm.* In Coding Theory and Applications, volume 388, pp.204-211, LNCS, 1989.
21. P. J. Lee and E. F. Brickell. *An observation on the security of McEliece's public-key cryptosystem;* In Advances in Cryptology , Eurocrypt'88, LNCS 330, pp. 275–280. Springer-Verlag, 1988.
22. R.J. McEliece: *A public key cryptosystem based on algebraic coding theory*; DSN Progress Report42-44, Jet Propulsion Laboratory, 1978, pp. 114-116.
23. Jeffrey O. Shallit, Gudmund S. Frandsen, Jonathan F. Buss: *The Computational Complexity of Some Problems of Linear Algebra problems*, BRICS series report, Aaarhus, Denmark, RS-96-33, available on the net http://www.brics.dk/RS/96/33/.

24. Jacques Patarin: *Hidden Fields Equations (HFE) and Isomorphisms of Polynomials (IP): two new families of Asymmetric Algorithms*; Eurocrypt'96, Springer Verlag, pp. 33-48.
25. Jacques Patarin, Louis Goubin, Nicolas Courtois: Quartz, *128-bit long digital signatures*; Cryptographers' Track Rsa Conference 2001, San Francisco 8-12 April 2001, LNCS2020, Springer-Verlag.
26. Jacques Patarin, Louis Goubin, Nicolas Courtois, + papers of Eli Biham, Aviad Kipnis, T. T. Moh, et al.: *Asymmetric Cryptography with Multivariate Polynomials over a Small Finite Field;* known as 'orange script', compilation of papers with added material. Available from JPatarin@slb.com.
27. David Pointcheval: *A new Identification Scheme Based on the Perceptrons Problem*; In Advances in Cryptology, Proceedings of Eurocrypt'95, LNCS 921, pp.319-328, Springer-Verlag.
28. David Pointcheval: *Les preuves de connaissance et leurs preuves de sécurité*, PhD thesis, December 1996, Caen University, France.
29. Ronald R. Rivest, Adi Shamir and Yael Tauman: *How to leak a secret* ; preprint, available at `http://www.theory.lcs.mit.edu/ rivest/publications.html`.
30. Adi Shamir: *Efficient signature schemes based on birational permutations*; Crypto'93, Springer-Verlag, pp1-12.
31. Adi Shamir: *An efficient Identification Scheme Based on Permuted Kernels*, In Advances in Cryptology, Proceedings of Crypto'89, LNCS 435, pp.606-609, Springer-Verlag.
32. Adi Shamir, Aviad Kipnis: *Cryptanalysis of the HFE Public Key Cryptosystem*; In Advances in Cryptology, Proceedings of Crypto'99, Springer-Verlag, LNCS.
33. Nicolas Courtois, Adi Shamir, Jacques Patarin, Alexander Klimov, *Efficient Algorithms for solving Overdefined Systems of Multivariate Polynomial Equations*, In Advances in Cryptology, Eurocrypt'2000, LNCS 1807, Springer-Verlag, pp. 392-407.
34. Jacques Stern: *A new identification scheme based on syndrome decoding*; In Advances in Cryptology, Proceedings of Crypto'93, LNCS 773, pp.13-21, Springer-Verlag.
35. Jacques Stern: *Designing identification schemes with keys of short size*; In Advances in Cryptology, Proceedings of Crypto'94, LNCS 839, pp.164-73, Springer-Verlag.
36. Jacques Stern: *A method for finding codewords of small weight;* Coding Theory and Applications, LNCS 434, pp.173-180, Springer-Verlag.
37. Jacques Stern, Florent Chabaud: *The cryptographic security of the syndrome decoding problem for rank distance codes*. In Advances in Cryptology, Asiacrypt'96, LNCS 1163, pp. 368-381, Springer-Verlag.
38. L.G. Valiant: *Completeness classes in algebra*. In Proc. Eleventh Ann. ACM Symp. Theor. Comp., pp. 249-261, 1979.
39. Alexander Vardy: *The intractability of computing the minimum distance of a code; IEEE Transactions on Information Theory, Nov 1997, Vol.43, No. 6; pp. 1757-1766.*
40. *Pascal Véron, Problème SD, Opérateur Trace, Schémas d'Identification et Codes de Goppa*; PhD thesis in french, l'Université de Toulon et du Var, France, july 1995.

10 Appendix 1 - Probability Distribution of Ranks

Following [13] the probability that a random matrix $\eta \times n$ is of rank r is

$$P(\eta, n, r) = \frac{(q^n - 1) \cdot \ldots \cdot (q^n - q^{r-1})}{(q^r - 1) \cdot \ldots \cdot (q^r - q^{r-1})} \cdot \frac{(q^\eta - 1) \cdot \ldots \cdot (q^\eta - q^{r-1})}{q^{\eta n}}.$$

If $r \leq min(n, \eta)$ it is non-zero, and when all the $n, \eta, r \to \infty$ we get the following approximation:

$$p(\eta, n, r) \;\simeq\; \mathcal{O}(q^{(\eta+n)r - r^2 - \eta n})$$

The probability that a random matrix $\eta \times n$ is of rank $> r$ is about:

$$(1 - \sum_{s=0}^{r} q^{(\eta+n)s - s^2 - \eta n}) \approx (1 - q^{(\eta+n)r - r^2 - \eta n})$$

There are $\frac{q^m - 1}{q-1}$ non-collinear combinations α of the M_i. The probability that all of them give $Rank(\sum_i \alpha_i M_i - M_0) > r$ with $r \leq min(n, \eta)$ is about:

$$Pr_\alpha[Rank \leq r](\eta, n, r) = 1 - (1 - q^{(\eta+n)r - r^2 - \eta n})^{\frac{q^m - 1}{q-1}}$$

We want to evaluate the value m_{max} such that for $m \leq m_{max}$ we expect to have solutions for a random MinRank, and such that for $m \approx m_{max}$, we expect to have one solution on average. Therefore:

$$(q^{m_{max}} - 1)/(q - 1) \cdot \mathcal{O}(q^{(\eta+n)r - r^2 - \eta n}) \approx 1$$

$$m_{max} \stackrel{def}{=} \eta n + r^2 - (\eta + n)r + 1$$

11 Appendix 2 - Achieving Fraud Probability 1/2

We present a technique to achieve the fraud probability 1/2 instead of 2/3. It has the following interesting features:

- It requires additional assumption (of type one-wayness of a function).
- Should this assumption fail, the scheme is still at least as secure as before, only with a worse impersonation probability.

The principle of the "trick" is to replace some random choices by a deterministic procedure so that they are still random but cannot be chosen. We add an additional "verifiable" requirement on generation of some values, and thus we eliminate some fraud scenarios (but not others). Then we modify the probabilities of different questions in order to balance the probabilities for the remaining fraud scenarios.

We consider any Zero-knowledge protocol in which a Prover picks up 2 values β_1 and β_2 such that $\beta_2 - \beta_1 = \alpha$ is a given (usually secret) value. Usually we will generate β_1 at random and compute β_2, which enables fraud scenarios in which the adversary may chose a value for one out of β_1, β_2. We want to avoid this. Let F be a function with a following properties:

(1) It is very hard to compute an inverse $F^{-1}(y)$ for a given random y.
(2) It is very easy to compute two solutions x and x' such that $F(x') - F(x)$ is a given value Δy and $x' = x + \Delta x$ with a given constant Δx.

Example 1: $F : x \mapsto x^2 \bmod N$, N being an RSA modulus. The inversion problem (1) is as hard as factoring.

Example 2: $F : GF(q)^n \to GF(q)^n$ is a set of random quadratic equations over a finite field. The inversion problem (1) is called MQ, is NP-hard very difficult in practice [26, 33].

In both examples, (2) is a linear problem easily solved.

We note that each of the above examples is applied with an operation '+' that belongs to a different group. Only the first example can be used for MinRank, as our '+' will be the component-by-component addition in the finite field.

11.1 Application to MinRank Scheme

Let $F : GF(q)^{n\eta} \to GF(q)^{n\eta}$ be a public fixed random set of quadratic equations. In the modified MinRank scheme, the Prover picks up two 160-bit seeds Z and $\overline{STX}$. Let $\Delta y = \text{Expand}(Z)$ and $(S, T, X) = \text{Expand}(\overline{STX})$ be the output of a pseudo-random generator. He solves

$$(S) \begin{cases} F(T(\sum \beta_{2i} M_i)S - TM_0S + X) - F(T(\sum \beta_{1i} M_i)S + X) = \text{Expand}(Z) \\ \beta_2 - \beta_1 = \alpha \end{cases}$$

The first equation becomes linear in β_1 after substitution of $\beta_2 = \beta_1 + \alpha$. He gets m linear equations with m variables β_{1i}. If there is no solution (β_1, β_2) found, he tries again with a new Z.

11.2 Verification that the Prover Follows the Scenario

If $\mathcal{Q} = 0$, the Prover will send an additional value Z. The Verifier will check that $F(TN_2S + X) - TM_0S - F(TN_1S + X) = \text{Expand}(Z)$. In the previous version of MinRank scheme possible fraud scenarios were:

01 Try to be able to answer $\mathcal{Q} = 0$ and 1.
It is easy to produce two matrices, seemingly $T(\sum \beta_{1i} M_i)S + X$ and $(T(\sum \beta_{2i} M_i)S - TM_0S + X)$, such that only one of them is really constructed in such a form, and the other is adjusted to get a difference of rank r.

02 Try to be able to answer $\mathcal{Q} = 0$ and 2 in the same way.

12 Try to be able to answer $\mathcal{Q} = 1$ and 2: We pick up any $\overline{STX}, \beta_1, \beta_2$ and produce a genuine $T(\sum \beta_{\mathcal{Q}i} M_i)S + X[-TM_0S]$.

0 Try to be able to answer $\mathcal{Q} = 0$ only. For this we just give any matrices that have a difference with rank r.

1 Try to be able to answer $\mathcal{Q} = 1$ only. For this we produce $T(\sum \beta_{1i} M_i)S + X$ in the required form.

2 Try to be able to answer $\mathcal{Q} = 2$ only. As above.

The new version excludes the scenarios (01) and (02). Let us see why on the example of scenario (01). We assume that a false Prover wants to answer $\mathcal{Q} = 0$ and 1. He may try the following possibilities:

a. Since S, T and X are always obtained as $\text{Expand}(\overline{STX})$, if we cheat and have not selected them in this way, we are only able to answer $\mathcal{Q} = 0$.

b. He may try to pick up β_1. Since F is one way (the NP-chard problem MQ), he will be unable to produce a matrix R such that $F(Q)-F(T(\sum \beta_{1i}M_i)S+X) = \text{Expand}(Z)$.
c. Another way is to try find R of rank r and write the $n\eta$ equations with m variables $(\sum \beta_{2i}M_i - M_0) - (\sum \beta_{1i}M_i) = R$. However to find a solution is hard because $\alpha = \beta_2 - \beta_1$ would allow him to solve an instance of MinRank.

Still an adversary has the capacity to answer all possible questions separately: fraud scenarios (0), (1) and (2).

11.3 Resulting Changes in the Protocol

Now we may modify the probabilities. The question $\mathcal{Q} = 0$ is asked with probability 1/2 and $\mathcal{Q} = 1, 2$ with probability 1/4 each. The following table shows the probabilities of success for all fraud scenarios.

Fraud scenario	0	1	2	01	02	12	012
Pr[Success] before	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{2}{3}$	$\frac{2}{3}$	$\frac{2}{3}$	0
now	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	0	0	$\frac{1}{2}$	0

A false Prover is detected with probability 1/2. Now only 20 instead of 35 rounds are needed to achieve the security of 10^{-6}.

Note: We obtained a more efficient authentication scheme with an added computational assumption based on the NP-hard problem MQ. This problem is believed very hard [33], but if it wasn't then the scenarios (01) and (02) will be possible again and the fraud probability will be 3/4. The MinRank scheme will remain secure, but with worse fraud probability, or equivalently, it will require more iterations.

11.4 Further Improvements

First we remark that if $\mathcal{Q} = 0$, it is not necessary at all to transmit the two values $TN_2S - TM_0S + X$ and $TN_1S + X$. In fact it is enough to transmit their difference TMS and Z that is already among the values that are transmitted. The values of $TN_2S - TM_0S + X$ and $TN_1S + X$ can be then recovered by the Verifier that has to solve a system similar to (11.1.(S)). We saved a transfer of one matrix $\eta \times n$.

Another improvement is to use only one seed $\overline{STXZ}$ with:

$$(S, T, X, Z) = \text{Expand}(\overline{STXZ})$$

12 The Modified Version MinRank-v2

Now we integrate all improvements in order to have a general view. The prover chooses a random seed of 160-bits $\overline{STXZ}$. Let

$$(S, T, X, Z) = \text{Expand}(\overline{STXZ})$$
$$\Delta y = \text{Expand}(Z)$$

Now the Prover solves:

$$(S)\begin{cases} F(T(\sum\beta_{2i}M_i)S - TM_0S + X) - F(T(\sum\beta_{1i}M_i)S + X) = \text{Expand}(Z) \\ \beta_2 - \beta_1 = \alpha \end{cases}$$

If there is no solution, (β_1, β_2), we try again a small number of times. with a different seed $\overline{STXZ}$. Then in each round of authentication:

1. The Prover sends to the Verifier:

$$\xrightarrow{\hspace{8em}}$$
$$H(\overline{STXZ}),\ H(TN_1S + X),\ H(TN_2S + X - TM_0S)$$

2. The Verifier chooses a query $\mathcal{Q}$, such that $\mathcal{Q} = 0$ with probability $1/2$, and $\mathcal{Q} \in \{1, 2\}$ with probability $1/4$ each. He sends $\mathcal{Q}$ to the Prover.

$$\xleftarrow{\hspace{8em}}$$
$$\mathcal{Q} \in \{0, 1, 2\}$$

3. If $\mathcal{Q} = 0$, the Prover gives the following values:

$$\xrightarrow{\hspace{8em}}$$
$$TMS,\ Z$$

Verification $\mathcal{Q} = \mathbf{0}$: The Verifier will compute the $(TN_1S + X)$ and $(TN_2S + X - TM_0S)$, see 11.4 Then he will accept if $H(TN_1S + X)$ and $H(TN_2S + X - TM_0S)$ are correct, and if $Rank(TMS) = r$.

3' In the case $\mathcal{Q} = 1, 2$, the Prover reveals:

$$\xrightarrow{\hspace{8em}}$$
$$\overline{STXZ},\ \beta_{\mathcal{Q}}$$

Verification $\mathcal{Q} = \mathbf{1}, \mathbf{2}$: The Verifier checks if S and T are invertible and if $H(\overline{STXZ})$ is correct. Then he computes

$$TN_{\mathcal{Q}}S = \sum \beta_{\mathcal{Q}i}\ \ TM_iS$$

and verifies the correctness of $H(TN_1S + X)$ or $H(TN_2S + X - TM_0S)$.

12.1 Improvements in the Communications

As in 8.1 we compute the communication complexity of the new version. By inspection we see that it becomes:

$$\left(3 \cdot 160 + 2 + \frac{n\eta + m}{2} \log_2 q\right) \cdot \#\text{rounds}$$

Remark: The value of 160 bits for a length of seeds and commitments is appropriate for the security level of 2^{80} and should be increased otherwise. For example for a security level 2^{SF} we should use $2SF$ bits. So we get

$$\left(6SF + 2 + \frac{n\eta + m}{2} \log_2 q\right) \cdot \#\text{rounds}$$

Chaining random seeds. It is also possible to save on the size of random seeds used in the scheme and use one single seed A_0 of $2SF$ bits for the whole scheme. Each time we compute a seed A_i as the following:

$$A_i = H(A_0||i||b_1, \ldots, b_7)$$

with an appropriate length hash function and with 7 random bits b_i, as the seed $\overline{STXZ} = A_i$ will only work in sec. 12 with a probability different than 1. Thus we may try again for b_i in order to have a working seed. With $2^7 = 128$ tries we have a negligible probability to never find an appropriate seed. The main seed A_0 is only given **at the end**, after all rounds of authentication, and only then all the verifications are carried. Now, with the exception of A_0, each round requires only $4SF+7+2+\frac{n\eta+m}{2}\log_2 q$ bits. Thus we get a communication complexity of

$$2SF + \left(4SF + 9 + \frac{n\eta + m}{2}\log_2 q\right) \cdot \#\text{rounds}.$$

13 Appendix 3 - Multi-user Setting

It is easy to produce almost totally random instances of MinRank with several users, each of which has one solution to MinRank and no information about other solutions. We pick $1+m$ [pseudo-]random matrices $M_0; M_1, \ldots, M_m$. Each user i has the right to pick up a matrix U_i such that $U_i - M_0$, plus some randomly chosen linear combination of the $M_1 \ldots M_m$, has a small rank. It can be done for an unlimited (in practice) number of users. Then the set of matrices: $M_0; M_1, \ldots, M_m;$ with the $\{U_i | i \in G\}$ is the public key for any small [9] subgroup G.

13.1 Ring Signatures with MinRank

A well known method (see [5]) that transforms a Zero-knowledge protocol into a signature scheme will also apply to MinRank. This in turn can be combined with the above multi-user setting. We obtain an anonymous group signature scheme known as **a ring signature scheme** [29], with the following properties:

- Each group member signs with his own private key (no shared secrets).
- He may sign on behalf on **any** subgroup of users that contains himself.
- There is no central authority.
- The user within the group that signs is anonymous (inside the group).
- Security is based on the NP-hard problem MinRank.
- At any moment we may introduce a new user and remove a user.
- Selective repudiation of signatures: introducing a new user U' and invalidating his public key can be used as a mean to repudiate all signatures made with this user included in the subgroup. The repudiation is controlled by the person who knows the secret key of U' and publishes it.

[9] Here the total number of matrices m can be very big: attacks such as the "big m attack" described in §4.2 or in [11, 10] will only apply to a smaller m', the maximum cardinal of a subgroup used.