Schemi di Firma Digitale Post-Quantum Basati su MinRank

Carlo Sanna

Gruppo di Crittografia e Teoria dei Numeri
Dipartimento di Scienze Matematiche
Politecnico di Torino

MinRank

MinRank è un problema di algebra lineare che è stato definito per la prima volta da Buss, Frandsen e Shallit nel 1999 [1].

MinRank (definizione formale)

- **Parametri:** $q, m, n, k, r \in \mathbb{Z}^+$, con q potenza di un numero primo.
- **Istanza:** $\boldsymbol{M} = (M_0, M_1, \ldots, M_k) \in (\mathbb{F}_q^{m \times n})^{k+1}$.
- **Soluzione:** $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_k) \in \mathbb{F}_q^k$ tale che la matrice

$$E := M_0 + \sum_{i=1}^{k} \alpha_i M_i$$

ha rango minore o uguale a r.

Qui $\mathbb{F}_q$ è un campo finito di q elementi e $\mathbb{F}_q^{m \times n}$ è lo spazio vettoriale delle matrici $m \times n$ con elementi in $\mathbb{F}_q$.

MinRank

MinRank è un problema di algebra lineare che è stato definito per la prima volta da Buss, Frandsen e Shallit nel 1999 [1].

MinRank (definizione formale)

- **Parametri:** $q, m, n, k, r \in \mathbb{Z}^+$, con q potenza di un numero primo.
- **Istanza:** $\boldsymbol{M} = (M_0, M_1, \ldots, M_k) \in (\mathbb{F}_q^{m \times n})^{k+1}$.
- **Soluzione:** $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_k) \in \mathbb{F}_q^k$ tale che la matrice

$$E := M_0 + \sum_{i=1}^{k} \alpha_i M_i$$

ha rango minore o uguale a r.

Qui $\mathbb{F}_q$ è un campo finito di q elementi e $\mathbb{F}_q^{m \times n}$ è lo spazio vettoriale delle matrici $m \times n$ con elementi in $\mathbb{F}_q$.

Perché MinRank?

MinRank è un buon candidato per costruire firme digitali post-quantum per le seguenti ragioni:

- MinRank è un problema NP-completo. Se NP $\neq$ P, come molti congetturano, allora non può esistere un algoritmo per risolvere MinRank in tempo polinomiale.
- Gli algoritmi per risolvere MinRank sono stati studiati intensamente e la loro complessità computazionale è ben compresa [2, 3, 4, 5, 6, 7, 8].
- MinRank è ben noto alla communità crittografica per le sue applicazioni alla crittoanalisi degli schemi multivariati [9, 10, 11, 12].
- MinRank è definito in termini di semplici operazioni di algebra lineare, che è possibile implementare efficientemente e con facilità.

Perché MinRank?

MinRank è un buon candidato per costruire firme digitali post-quantum per le seguenti ragioni:

- MinRank è un problema NP-completo. Se NP $\neq$ P, come molti congetturano, allora non può esistere un algoritmo per risolvere MinRank in tempo polinomiale.
- Gli algoritmi per risolvere MinRank sono stati studiati intensamente e la loro complessità computazionale è ben compresa [2, 3, 4, 5, 6, 7, 8].
- MinRank è ben noto alla communità crittografica per le sue applicazioni alla crittoanalisi degli schemi multivariati [9, 10, 11, 12].
- MinRank è definito in termini di semplici operazioni di algebra lineare, che è possibile implementare efficientemente e con facilità.

Perché MinRank?

MinRank è un buon candidato per costruire firme digitali post-quantum per le seguenti ragioni:

- MinRank è un problema NP-completo. Se NP $\neq$ P, come molti congetturano, allora non può esistere un algoritmo per risolvere MinRank in tempo polinomiale.
- Gli algoritmi per risolvere MinRank sono stati studiati intensamente e la loro complessità computazionale è ben compresa [2, 3, 4, 5, 6, 7, 8].
- MinRank è ben noto alla communità crittografica per le sue applicazioni alla crittoanalisi degli schemi multivariati [9, 10, 11, 12].
- MinRank è definito in termini di semplici operazioni di algebra lineare, che è possibile implementare efficientemente e con facilità.

Perché MinRank?

MinRank è un buon candidato per costruire firme digitali post-quantum per le seguenti ragioni:

- MinRank è un problema NP-completo. Se NP $\neq$ P, come molti congetturano, allora non può esistere un algoritmo per risolvere MinRank in tempo polinomiale.
- Gli algoritmi per risolvere MinRank sono stati studiati intensamente e la loro complessità computazionale è ben compresa [2, 3, 4, 5, 6, 7, 8].
- MinRank è ben noto alla communità crittografica per le sue applicazioni alla crittoanalisi degli schemi multivariati [9, 10, 11, 12].
- MinRank è definito in termini di semplici operazioni di algebra lineare, che è possibile implementare efficientemente e con facilità.

Perché MinRank?

MinRank è un buon candidato per costruire firme digitali post-quantum per le seguenti ragioni:

- MinRank è un problema NP-completo. Se NP $\neq$ P, come molti congetturano, allora non può esistere un algoritmo per risolvere MinRank in tempo polinomiale.
- Gli algoritmi per risolvere MinRank sono stati studiati intensamente e la loro complessità computazionale è ben compresa [2, 3, 4, 5, 6, 7, 8].
- MinRank è ben noto alla communità crittografica per le sue applicazioni alla crittoanalisi degli schemi multivariati [9, 10, 11, 12].
- MinRank è definito in termini di semplici operazioni di algebra lineare, che è possibile implementare efficientemente e con facilità.

Firme Digitali Post-Quantum Basate su MinRank

Al momento, sono state proposte le seguenti firme digitali post-quantum basate su MinRank:

- Courtois (2001) [13].
- MR-DSS di Bellini, Esser, Sanna e Verbel (2022) [14].
- Adj, Rivera-Zamarripa e Verbel (2022) [15].
- Feneuil's (2022) [16].
- MiRitH di Adj, Barbero, Bellini, Esser, Rivera-Zamarripa, Sanna, Verbel, Zweydinger (coming soon).

Schema di Courtois (1/2)

L'idea di Courtois consiste nel costruire una dimostrazione a conoscenza zero della soluzione $\boldsymbol{\alpha}$ di un'instanza $\boldsymbol{M}$ di MinRank nel modo seguente:

(1) Il prover genera un $\beta \in \mathbb{F}_q^k$ casuale e calcola le matrici

$$N_1 := \sum_{i=1}^{k} \beta_i M_i \quad \text{e} \quad N_2 := M_0 + \sum_{i=1}^{k} (\alpha_i - \beta_i) M_i.$$

(Notare che $N_2 - N_1 = M_0 + \sum_{i=1}^{k} \alpha_i M_i =: E$.)

(2) Il prover genera due matrici casuali invertibili $T \in \mathbb{F}_q^{m \times m}$ e $S \in \mathbb{F}_q^{n \times n}$ e una matrice casuale $X \in \mathbb{F}_q^{m \times n}$, che usa per mascherare N_1 e N_2:

$$Z_1 := TN_1S + X \quad \text{e} \quad Z_2 := TN_2S + X.$$

(Notare che $Z_2 - Z_1 = TES$ ha lo stesso rango di E.)

(3) Il prover invia al verifier gli hash crittografici $h_0 := H(S, T, X)$, $h_1 := Z_1$ e $h_2 := Z_2$.

Schema di Courtois (1/2)

L'idea di Courtois consiste nel costruire una dimostrazione a conoscenza zero della soluzione $\boldsymbol{\alpha}$ di un'instanza $\boldsymbol{M}$ di MinRank nel modo seguente:

(1) Il prover genera un $\boldsymbol{\beta} \in \mathbb{F}_q^k$ casuale e calcola le matrici

$$N_1 := \sum_{i=1}^{k} \beta_i M_i \quad \text{e} \quad N_2 := M_0 + \sum_{i=1}^{k} (\alpha_i - \beta_i) M_i.$$

(Notare che $N_2 - N_1 = M_0 + \sum_{i=1}^{k} \alpha_i M_i =: E$.)

(2) Il prover genera due matrici casuali invertibili $T \in \mathbb{F}_q^{m \times m}$ e $S \in \mathbb{F}_q^{n \times n}$ e una matrice casuale $X \in \mathbb{F}_q^{m \times n}$, che usa per mascherare N_1 e N_2:

$$Z_1 := TN_1S + X \quad \text{e} \quad Z_2 := TN_2S + X.$$

(Notare che $Z_2 - Z_1 = TES$ ha lo stesso rango di E.)

(3) Il prover invia al verifier gli hash crittografici $h_0 := H(S, T, X)$, $h_1 := Z_1$ e $h_2 := Z_2$.

Schema di Courtois (1/2)

L'idea di Courtois consiste nel costruire una dimostrazione a conoscenza zero della soluzione $\boldsymbol{\alpha}$ di un'instanza $\boldsymbol{M}$ di MinRank nel modo seguente:

(1) Il prover genera un $\boldsymbol{\beta} \in \mathbb{F}_q^k$ casuale e calcola le matrici

$$N_1 := \sum_{i=1}^{k} \beta_i M_i \quad \text{e} \quad N_2 := M_0 + \sum_{i=1}^{k} (\alpha_i - \beta_i) M_i.$$

(Notare che $N_2 - N_1 = M_0 + \sum_{i=1}^{k} \alpha_i M_i =: E$.)

(2) Il prover genera due matrici casuali invertibili $T \in \mathbb{F}_q^{m \times m}$ e $S \in \mathbb{F}_q^{n \times n}$ e una matrice casuale $X \in \mathbb{F}_q^{m \times n}$, che usa per mascherare N_1 e N_2:

$$Z_1 := TN_1S + X \quad \text{e} \quad Z_2 := TN_2S + X.$$

(Notare che $Z_2 - Z_1 = TES$ ha lo stesso rango di E.)

(3) Il prover invia al verifier gli hash crittografici $h_0 := H(S, T, X)$, $h_1 := Z_1$ e $h_2 := Z_2$.

Schema di Courtois (1/2)

L'idea di Courtois consiste nel costruire una dimostrazione a conoscenza zero della soluzione $\boldsymbol{\alpha}$ di un'instanza $\boldsymbol{M}$ di MinRank nel modo seguente:

(1) Il prover genera un $\boldsymbol{\beta} \in \mathbb{F}_q^k$ casuale e calcola le matrici

$$N_1 := \sum_{i=1}^{k} \beta_i M_i \quad \text{e} \quad N_2 := M_0 + \sum_{i=1}^{k} (\alpha_i - \beta_i) M_i.$$

(Notare che $N_2 - N_1 = M_0 + \sum_{i=1}^{k} \alpha_i M_i =: E$.)

(2) Il prover genera due matrici casuali invertibili $T \in \mathbb{F}_q^{m \times m}$ e $S \in \mathbb{F}_q^{n \times n}$ e una matrice casuale $X \in \mathbb{F}_q^{m \times n}$, che usa per mascherare N_1 e N_2:

$$Z_1 := TN_1S + X \quad \text{e} \quad Z_2 := TN_2S + X.$$

(Notare che $Z_2 - Z_1 = TES$ ha lo stesso rango di E.)

(3) Il prover invia al verifier gli hash crittografici $h_0 := H(S, T, X)$, $h_1 := Z_1$ e $h_2 := Z_2$.

Schema di Courtois (1/2)

(4) Il verifier invia al prover una sfida $\mathrm{ch} \in \{0, 1, 2\}$.

(5) Se $\mathrm{ch} = 0$, il prover rivela Z_1, Z_2 e il verifier controlla che

$$H(Z_1) = h_1, \quad H(Z_2) = h_2, \quad \operatorname{rank}(Z_1 - Z_2) \leq r.$$

Se $\mathrm{ch} = 1$, il prover rivela S, T, X e β. Il verifier ricalcola Z_1 e controlla che

$$S, T \text{ sono invertibili}, \quad H(S, T, X) = h_0, \quad H(Z_1) = h_1.$$

Se $\mathrm{ch} = 2$, il prover rivela S, T, X e $\alpha - \beta$. Il verifier procede similmente al caso precedente.

Teorema

La precedente è una dimostrazione a conoscenza zero per MinRank con cheating probability 2/3.

La dimostrazione a conoscenza zero è poi convertita in uno schema di firma digitale utilizzando la trasformata di Fiat–Shamir.

Schema di Courtois (1/2)

(4) Il verifier invia al prover una sfida $\mathsf{ch} \in \{0, 1, 2\}$.

(5) Se $\mathsf{ch} = 0$, il prover rivela Z_1, Z_2 e il verifier controlla che

$$H(Z_1) = h_1, \quad H(Z_2) = h_2, \quad \mathrm{rank}(Z_1 - Z_2) \leq r.$$

Se $\mathsf{ch} = 1$, il prover rivela S, T, X e β. Il verifier ricalcola Z_1 e controlla che

$$S, T \text{ sono invertibili}, \quad H(S, T, X) = h_0, \quad H(Z_1) = h_1.$$

Se $\mathsf{ch} = 2$, il prover rivela S, T, X e $\alpha - \beta$. Il verifier procede similmente al caso precedente.

Teorema

La precedente è una dimostrazione a conoscenza zero per MinRank con cheating probability 2/3.

La dimostrazione a conoscenza zero è poi convertita in uno schema di firma digitale utilizzando la trasformata di Fiat–Shamir.

Schema di Courtois (1/2)

(4) Il verifier invia al prover una sfida $\mathrm{ch} \in \{0, 1, 2\}$.

(5) Se $\mathrm{ch} = 0$, il prover rivela Z_1, Z_2 e il verifier controlla che

$$H(Z_1) = h_1, \quad H(Z_2) = h_2, \quad \mathrm{rank}(Z_1 - Z_2) \leq r.$$

Se $\mathrm{ch} = 1$, il prover rivela S, T, X e β. Il verifier ricalcola Z_1 e controlla che

$$S, T \text{ sono invertibili}, \quad H(S, T, X) = h_0, \quad H(Z_1) = h_1.$$

Se $\mathrm{ch} = 2$, il prover rivela S, T, X e $\alpha - \beta$. Il verifier procede similmente al caso precedente.

Teorema

La precedente è una dimostrazione a conoscenza zero per MinRank con cheating probability 2/3.

La dimostrazione a conoscenza zero è poi convertita in uno schema di firma digitale utilizzando la trasformata di Fiat–Shamir.

Schema di Courtois (1/2)

(4) Il verifier invia al prover una sfida $\mathrm{ch} \in \{0, 1, 2\}$.

(5) Se $\mathrm{ch} = 0$, il prover rivela Z_1, Z_2 e il verifier controlla che

$$H(Z_1) = h_1, \quad H(Z_2) = h_2, \quad \mathrm{rank}(Z_1 - Z_2) \leq r.$$

Se $\mathrm{ch} = 1$, il prover rivela S, T, X e $\boldsymbol{\beta}$. Il verifier ricalcola Z_1 e controlla che

$$S, T \text{ sono invertibili}, \quad H(S, T, X) = h_0, \quad H(Z_1) = h_1.$$

Se $\mathrm{ch} = 2$, il prover rivela S, T, X e $\boldsymbol{\alpha} - \boldsymbol{\beta}$. Il verifier procede similmente al caso precedente.

Teorema

La precedente è una dimostrazione a conoscenza zero per MinRank con cheating probability 2/3.

La dimostrazione a conoscenza zero è poi convertita in uno schema di firma digitale utilizzando la trasformata di Fiat–Shamir.

Schema di Courtois (1/2)

(4) Il verifier invia al prover una sfida $\mathrm{ch} \in \{0, 1, 2\}$.

(5) Se $\mathrm{ch} = 0$, il prover rivela Z_1, Z_2 e il verifier controlla che

$$H(Z_1) = h_1, \quad H(Z_2) = h_2, \quad \operatorname{rank}(Z_1 - Z_2) \leq r.$$

Se $\mathrm{ch} = 1$, il prover rivela S, T, X e $\boldsymbol{\beta}$. Il verifier ricalcola Z_1 e controlla che

$$S, T \text{ sono invertibili}, \quad H(S, T, X) = h_0, \quad H(Z_1) = h_1.$$

Se $\mathrm{ch} = 2$, il prover rivela S, T, X e $\boldsymbol{\alpha} - \boldsymbol{\beta}$. Il verifier procede similmente al caso precedente.

Teorema

La precedente è una dimostrazione a conoscenza zero per MinRank con cheating probability 2/3.

La dimostrazione a conoscenza zero è poi convertita in uno schema di firma digitale utilizzando la trasformata di Fiat–Shamir.

Schema di Courtois (1/2)

(4) Il verifier invia al prover una sfida $\mathrm{ch} \in \{0, 1, 2\}$.

(5) Se $\mathrm{ch} = 0$, il prover rivela Z_1, Z_2 e il verifier controlla che

$$H(Z_1) = h_1, \quad H(Z_2) = h_2, \quad \operatorname{rank}(Z_1 - Z_2) \leq r.$$

Se $\mathrm{ch} = 1$, il prover rivela S, T, X e $\boldsymbol{\beta}$. Il verifier ricalcola Z_1 e controlla che

$$S, T \text{ sono invertibili}, \quad H(S, T, X) = h_0, \quad H(Z_1) = h_1.$$

Se $\mathrm{ch} = 2$, il prover rivela S, T, X e $\boldsymbol{\alpha} - \boldsymbol{\beta}$. Il verifier procede similmente al caso precedente.

Teorema

La precedente è una dimostrazione a conoscenza zero per MinRank con cheating probability $2/3$.

La dimostrazione a conoscenza zero è poi convertita in uno schema di firma digitale utilizzando la trasformata di Fiat–Shamir.

MR-DSS

MR-DSS (MinRank Digital Signature Scheme) è basato sull'idea originale di Courtois, ma la cheating probability della dimostrazione a conoscenza zero è ridotta da 2/3 ad 1/2 utilizzando i **sigma protocol with helper** introdotti da Beullens [17].

Un sigma protocol with helper è una particolare dimostrazione a conoscenza zero in cui è presente una terza parte (fittizia) detta **helper**. L'helper si occupa di generare alcune informazioni ausiliarie, che non dipendono dal segreto, che sono trasmesse parzialmente al verifier.

Ogni sigma protocol with helper può essere convertito in una dimostrazione a conoscenza zero rimuovendo l'helper con una particolare tecnica di **cut-and-choose**.

In MR-DSS, l'helper è utilizzato per gestire il caso $ch = 0$, riducendo così la cheating probability a 1/2.

MR-DSS

MR-DSS (MinRank Digital Signature Scheme) è basato sull'idea originale di Courtois, ma la cheating probability della dimostrazione a conoscenza zero è ridotta da 2/3 ad 1/2 utilizzando i **sigma protocol with helper** introdotti da Beullens [17].

Un sigma protocol with helper è una particolare dimostrazione a conoscenza zero in cui è presente una terza parte (fittizia) detta **helper**. L'helper si occupa di generare alcune informazioni ausiliarie, che non dipendono dal segreto, che sono trasmesse parzialmente al verifier.

Ogni sigma protocol with helper può essere convertito in una dimostrazione a conoscenza zero rimuovendo l'helper con una particolare tecnica di **cut-and-choose**.

In MR-DSS, l'helper è utilizzato per gestire il caso $\mathrm{ch} = 0$, riducendo così la cheating probability a 1/2.

MR-DSS

MR-DSS (MinRank Digital Signature Scheme) è basato sull'idea originale di Courtois, ma la cheating probability della dimostrazione a conoscenza zero è ridotta da 2/3 ad 1/2 utilizzando i **sigma protocol with helper** introdotti da Beullens [17].

Un sigma protocol with helper è una particolare dimostrazione a conoscenza zero in cui è presente una terza parte (fittizia) detta **helper**. L'helper si occupa di generare alcune informazioni ausiliarie, che non dipendono dal segreto, che sono trasmesse parzialmente al verifier.

Ogni sigma protocol with helper può essere convertito in una dimostrazione a conoscenza zero rimuovendo l'helper con una particolare tecnica di **cut-and-choose**.

In MR-DSS, l'helper è utilizzato per gestire il caso $ch = 0$, riducendo così la cheating probability a 1/2.

MR-DSS

MR-DSS (MinRank Digital Signature Scheme) è basato sull'idea originale di Courtois, ma la cheating probability della dimostrazione a conoscenza zero è ridotta da 2/3 ad 1/2 utilizzando i **sigma protocol with helper** introdotti da Beullens [17].

Un sigma protocol with helper è una particolare dimostrazione a conoscenza zero in cui è presente una terza parte (fittizia) detta **helper**. L'helper si occupa di generare alcune informazioni ausiliarie, che non dipendono dal segreto, che sono trasmesse parzialmente al verifier.

Ogni sigma protocol with helper può essere convertito in una dimostrazione a conoscenza zero rimuovendo l'helper con una particolare tecnica di **cut-and-choose**.

In MR-DSS, l'helper è utilizzato per gestire il caso $\mathrm{ch} = 0$, riducendo così la cheating probability a 1/2.

MiRitH

MiRitH (**Min**Rank **i**n **t**he **H**ead) è basato su due idee:

(1) Il **modello di Kipnis–Shamir**, ovvero se esiste $K \in \mathbb{F}_q^{r \times (n-r)}$ tale che

$$M_0^{\mathrm{L}} + \sum_{i=1}^{k} \alpha_i M_i^{\mathrm{L}} = -\left(M_0^{\mathrm{R}} + \sum_{i=1}^{k} \alpha_i M_i^{\mathrm{R}} \right) K \qquad (\star)$$

allora α è una soluzione di MinRank. Qui A^{L} (risp. A^{R}) è la matrice costituita dalla prime $n-r$ (risp. r) colonne della matrice A.

(2) Dare una dimostrazione a conoscenza zero dell'identità matriciale $(\star)$ utilizzando la tecnica di **Multi-Party Computation in the head**, ideata da Ishai et al. [18], la quale permette di trasformare protocolli di multy-party computation in dimostrazioni a conoscenza zero.

MiRitH (**Min**Rank **i**n **t**he **H**ead) è basato su due idee:

(1) Il **modello di Kipnis–Shamir**, ovvero se esiste $K \in \mathbb{F}_q^{r \times (n-r)}$ tale che

$$M_0^{\mathrm{L}} + \sum_{i=1}^{k} \alpha_i M_i^{\mathrm{L}} = - \left(M_0^{\mathrm{R}} + \sum_{i=1}^{k} \alpha_i M_i^{\mathrm{R}} \right) K \qquad (\star)$$

allora $\boldsymbol{\alpha}$ è una soluzione di MinRank. Qui A^{L} (risp. A^{R}) è la matrice costituita dalla prime $n - r$ (risp. r) colonne della matrice A.

(2) Dare una dimostrazione a conoscenza zero dell'identità matriciale $(\star)$ utilizzando la tecnica di **Multi-Party Computation in the head**, ideata da Ishai et al. [18], la quale permette di trasformare protocolli di multy-party computation in dimostrazioni a conoscenza zero.

MiRitH

MiRitH (**Min**Rank **i**n **t**he **H**ead) è basato su due idee:

(1) Il **modello di Kipnis–Shamir**, ovvero se esiste $K \in \mathbb{F}_q^{r \times (n-r)}$ tale che

$$M_0^{\mathrm{L}} + \sum_{i=1}^{k} \alpha_i M_i^{\mathrm{L}} = -\left(M_0^{\mathrm{R}} + \sum_{i=1}^{k} \alpha_i M_i^{\mathrm{R}} \right) K \qquad (\star)$$

allora $\boldsymbol{\alpha}$ è una soluzione di MinRank. Qui A^{L} (risp. A^{R}) è la matrice costituita dalla prime $n-r$ (risp. r) colonne della matrice A.

(2) Dare una dimostrazione a conoscenza zero dell'identità matriciale $(\star)$ utilizzando la tecnica di **Multi-Party Computation in the head**, ideata da Ishai et al. [18], la quale permette di trasformare protocolli di multy-party computation in dimostrazioni a conoscenza zero.

Fenuil's (1/2)

Fenuil ha ideato due schemi di firma digitale basati su MinRank. Entrambi utilizzano multi-party computation in the head.

I dettagli sono piuttosto tecnici e menzioniamo solo due cose:

(1) Il primo schema utilizza la **rank decomposition**, ovvero il fatto che la matrice E ha rango minore o uguale a r se e solo se esistono $X \in \mathbb{F}_q^{m \times r}$ e $Y \in \mathbb{F}_q^{r \times n}$ tali che $E = XY$.

(2) Il secondo schema usa dei particolari polinomi. Fissata una base di $\mathbb{F}_{q^n}$ su $\mathbb{F}_q$, identifichiamo le colonne di $E \in \mathbb{F}_q^{m \times n}$ con gli elementi di $\mathbb{F}_{q^n}$ e definiamo il polinomio

$$L_E(X) := \prod_{u \in \mathrm{columnspace}(E)} (X - u) \in \mathbb{F}_{q^n}[X]$$

dove $u \in \mathbb{F}_{q^n}$ varia nello spazio delle colonne di E. [...]

Fenuil's (1/2)

Fenuil ha ideato due schemi di firma digitale basati su MinRank. Entrambi utilizzano multi-party computation in the head.

I dettagli sono piuttosto tecnici e menzioniamo solo due cose:

(1) Il primo schema utilizza la **rank decomposition**, ovvero il fatto che la matrice E ha rango minore o uguale a r se e solo se esistono $X \in \mathbb{F}_q^{m \times r}$ e $Y \in \mathbb{F}_q^{r \times n}$ tali che $E = XY$.

(2) Il secondo schema usa dei particolari polinomi. Fissata una base di $\mathbb{F}_{q^n}$ su $\mathbb{F}_q$, identifichiamo le colonne di $E \in \mathbb{F}_q^{m \times n}$ con gli elementi di $\mathbb{F}_{q^n}$ e definiamo il polinomio

$$L_E(X) := \prod_{u \in \mathrm{columnspace}(E)} (X - u) \in \mathbb{F}_{q^n}[X]$$

dove $u \in \mathbb{F}_{q^n}$ varia nello spazio delle colonne di E. [...]

Fenuil's (1/2)

Fenuil ha ideato due schemi di firma digitale basati su MinRank. Entrambi utilizzano multi-party computation in the head.

I dettagli sono piuttosto tecnici e menzioniamo solo due cose:

(1) Il primo schema utilizza la **rank decomposition**, ovvero il fatto che la matrice E ha rango minore o uguale a r se e solo se esistono $X \in \mathbb{F}_q^{m \times r}$ e $Y \in \mathbb{F}_q^{r \times n}$ tali che $E = XY$.

(2) Il secondo schema usa dei particolari polinomi. Fissata una base di $\mathbb{F}_{q^n}$ su $\mathbb{F}_q$, identifichiamo le colonne di $E \in \mathbb{F}_q^{m \times n}$ con gli elementi di $\mathbb{F}_{q^n}$ e definiamo il polinomio

$$L_E(X) := \prod_{u \in \mathrm{columnspace}(E)} (X - u) \in \mathbb{F}_{q^n}[X]$$

dove $u \in \mathbb{F}_{q^n}$ varia nello spazio delle colonne di E. [...]

[...] allora si dimostra che $L_E(X)$ è della forma

$$L_E(X) = X^{q^r} + \sum_{i=0}^{r-1} c_i X^{q^i},$$

che lo rende comodo per la MPC, poiché l'isomorfismo di Frobenius $X \mapsto X^q$ è una funzione lineare su $\mathbb{F}_{q^n}$. Inoltre, E ha rango minore o uguale a r se e solo se le colonne di E sono zeri di $L_E(U)$.

Confronto degli Schemi

Gli schemi a confronto per la scelta dei parametri $q = 16$, $m = n = 16$, $k = 142$, $r = 4$, che corrisponde ad un livello di sicurezza di $\lambda = 128$ bits.

Schema	Firma (kB)
Courtois'	28,5
MR-DSS	26,4
MiRitH	5,6
Fenuil (RD)	7,1
Fenuil (LP)	5,5

Nota: Per ogni schema si è scelta la variante che minimizza la dimensione della firma, senza tenere conto della velocità di firma e verifica.

Generazione delle Chiavi (1/3)

Nello schema originale di Courtois, la chiave pubblica $\boldsymbol{M}$ e la chiave segreta $\boldsymbol{\alpha}$ sono generate nel modo seguente:

(1) $M_1, \ldots, M_k \in \mathbb{F}_q^{m \times n}$ sono generate da un seme casuale $\text{seed}_{\text{pk}} \in \{0,1\}^\lambda$ utilizzando un generatore pseudocasuale crittograficamente sicuro.

(2) $E \in \mathbb{F}_q^{m \times n}$ di rango r e $\alpha_1, \ldots, \alpha_k \in \mathbb{F}_q$ sono generati da un seme casuale $\text{seed}_{\text{sk}} \in \{0,1\}^\lambda$ utilizzando un generatore pseudocasuale crittograficamente sicuro.

(3) $M_0 := E - \sum_{i=1}^{k} \alpha_i M_i$.

La chiave pubblica può essere ricostruita da $(M_0, \text{seed}_{\text{pk}})$ e dunque occupa $mn \log_2 q + \lambda$ bits, mentre la chiave segreta è generata da seed_{sk} e perciò occupa λ bits.

Generazione delle Chiavi (1/3)

Nello schema originale di Courtois, la chiave pubblica $\boldsymbol{M}$ e la chiave segreta $\boldsymbol{\alpha}$ sono generate nel modo seguente:

(1) $M_1, \ldots, M_k \in \mathbb{F}_q^{m \times n}$ sono generate da un seme casuale $\text{seed}_{\text{pk}} \in \{0,1\}^\lambda$ utilizzando un generatore pseudocasuale crittograficamente sicuro.

(2) $E \in \mathbb{F}_q^{m \times n}$ di rango r e $\alpha_1, \ldots, \alpha_k \in \mathbb{F}_q$ sono generati da un seme casuale $\text{seed}_{\text{sk}} \in \{0,1\}^\lambda$ utilizzando un generatore pseudocasuale crittograficamente sicuro.

(3) $M_0 := E - \sum_{i=1}^{k} \alpha_i M_i$.

La chiave pubblica può essere ricostruita da $(M_0, \text{seed}_{\text{pk}})$ e dunque occupa $mn \log_2 q + \lambda$ bits, mentre la chiave segreta è generata da seed_{sk} e perciò occupa λ bits.

Generazione delle Chiavi (1/3)

Nello schema originale di Courtois, la chiave pubblica $\boldsymbol{M}$ e la chiave segreta $\boldsymbol{\alpha}$ sono generate nel modo seguente:

(1) $M_1, \ldots, M_k \in \mathbb{F}_q^{m \times n}$ sono generate da un seme casuale $\text{seed}_{\text{pk}} \in \{0,1\}^\lambda$ utilizzando un generatore pseudocasuale crittograficamente sicuro.

(2) $E \in \mathbb{F}_q^{m \times n}$ di rango r e $\alpha_1, \ldots, \alpha_k \in \mathbb{F}_q$ sono generati da un seme casuale $\text{seed}_{\text{sk}} \in \{0,1\}^\lambda$ utilizzando un generatore pseudocasuale crittograficamente sicuro.

(3) $M_0 := E - \sum_{i=1}^{k} \alpha_i M_i$.

La chiave pubblica può essere ricostruita da $(M_0, \text{seed}_{\text{pk}})$ e dunque occupa $mn \log_2 q + \lambda$ bits, mentre la chiave segreta è generata da seed_{sk} e perciò occupa λ bits.

Generazione delle Chiavi (1/3)

Nello schema originale di Courtois, la chiave pubblica $\boldsymbol{M}$ e la chiave segreta $\boldsymbol{\alpha}$ sono generate nel modo seguente:

(1) $M_1, \ldots, M_k \in \mathbb{F}_q^{m \times n}$ sono generate da un seme casuale $\text{seed}_{\text{pk}} \in \{0,1\}^\lambda$ utilizzando un generatore pseudocasuale crittograficamente sicuro.

(2) $E \in \mathbb{F}_q^{m \times n}$ di rango r e $\alpha_1, \ldots, \alpha_k \in \mathbb{F}_q$ sono generati da un seme casuale $\text{seed}_{\text{sk}} \in \{0,1\}^\lambda$ utilizzando un generatore pseudocasuale crittograficamente sicuro.

(3) $M_0 := E - \sum_{i=1}^{k} \alpha_i M_i$.

La chiave pubblica può essere ricostruita da $(M_0, \text{seed}_{\text{pk}})$ e dunque occupa $mn \log_2 q + \lambda$ bits, mentre la chiave segreta è generata da seed_{sk} e perciò occupa λ bits.

Generazione delle Chiavi (1/3)

Nello schema originale di Courtois, la chiave pubblica $\boldsymbol{M}$ e la chiave segreta $\boldsymbol{\alpha}$ sono generate nel modo seguente:

(1) $M_1, \ldots, M_k \in \mathbb{F}_q^{m \times n}$ sono generate da un seme casuale $\text{seed}_{\text{pk}} \in \{0,1\}^\lambda$ utilizzando un generatore pseudocasuale crittograficamente sicuro.

(2) $E \in \mathbb{F}_q^{m \times n}$ di rango r e $\alpha_1, \ldots, \alpha_k \in \mathbb{F}_q$ sono generati da un seme casuale $\text{seed}_{\text{sk}} \in \{0,1\}^\lambda$ utilizzando un generatore pseudocasuale crittograficamente sicuro.

(3) $M_0 := E - \sum_{i=1}^{k} \alpha_i M_i$.

La chiave pubblica può essere ricostruita da $(M_0, \text{seed}_{\text{pk}})$ e dunque occupa $mn \log_2 q + \lambda$ bits, mentre la chiave segreta è generata da seed_{sk} e perciò occupa λ bits.

Generazione delle Chiavi (2/3)

In MR-DSS, il metodo precedente è migliorato riducendo la dimensione della chiave pubblica a $(mn - k)\log_2 q + \lambda$ bits. Il miglioramento consiste nell'utilizzare solo matrici $M_0, \ldots, M_k$ in una forma canonica, che non riduce la difficoltà di MinRank.

MiRitH e gli schemi di Feneuil, non considerano miglioramenti nella dimensione della chiave pubblica.

Infine, Di Scala e Sanna (2023) [19] hanno ottenuto un metodo di generazione delle chiavi che riduce la dimensione della chiave pubblica a $(m(n - r) - k)\log_2 q + \lambda$ bits. L'idea è di utilizzare il modello di Kipnis–Shamir in combinazione a una forma canonica per $M_0, \ldots, M_k$.

Generazione delle Chiavi (2/3)

In MR-DSS, il metodo precedente è migliorato riducendo la dimensione della chiave pubblica a $(mn - k)\log_2 q + \lambda$ bits. Il miglioramento consiste nell'utilizzare solo matrici $M_0, \ldots, M_k$ in una forma canonica, che non riduce la difficoltà di MinRank.

MiRitH e gli schemi di Feneuil, non considerano miglioramenti nella dimensione della chiave pubblica.

Infine, Di Scala e Sanna (2023) [19] hanno ottenuto un metodo di generazione delle chiavi che riduce la dimensione della chiave pubblica a $(m(n - r) - k)\log_2 q + \lambda$ bits. L'idea è di utilizzare il modello di Kipnis–Shamir in combinazione a una forma canonica per $M_0, \ldots, M_k$.

Generazione delle Chiavi (2/3)

In MR-DSS, il metodo precedente è migliorato riducendo la dimensione della chiave pubblica a $(mn - k)\log_2 q + \lambda$ bits. Il miglioramento consiste nell'utilizzare solo matrici $M_0, \ldots, M_k$ in una forma canonica, che non riduce la difficoltà di MinRank.

MiRitH e gli schemi di Feneuil, non considerano miglioramenti nella dimensione della chiave pubblica.

Infine, Di Scala e Sanna (2023) [19] hanno ottenuto un metodo di generazione delle chiavi che riduce la dimensione della chiave pubblica a $(m(n - r) - k)\log_2 q + \lambda$ bits. L'idea è di utilizzare il modello di Kipnis–Shamir in combinazione a una forma canonica per $M_0, \ldots, M_k$.

Generazione delle Chiavi (3/3)

I metodi di generazione delle chiavi a confronto (la dimensione della chiave segreta è sempre λ bits).

Sicurezza (bits)	Parametri					Chiave pubblica (bits)		
λ	q	m	n	k	r	Courtois	MR-DSS	D. & S.
128	16	16	16	142	4	1,152	584	328
192	16	19	19	167	6	1,636	968	512
256	16	22	22	254	6	2,192	1,176	648

Grazie per l'attenzione!

Sommario della presentazione:

- MinRank
- Perché MinRank?
- Firme Digitali Post-Quantum basate su MinRank
  - Courtois'
  - MR-DSS
  - MiRitH
  - Feneuil's
- Generazione delle Chiavi

(Riferimenti bibliografici nelle pagine successive.)

References I

[1] J. F. Buss, G. S. Frandsen, and J. O. Shallit.
The computational complexity of some problems of linear algebra.
J. Comput. System Sci., 58(3):572–596, 1999.

[2] M. Bardet and M. Bertin.
Improvement of algebraic attacks for solving superdetermined MinRank instances.
Lecture Notes in Comput. Sci., 13512:107–123, 2022.

[3] M. Bardet, P. Briaud, M. Bros, P. Gaborit, and J.-P. Tillich.
Revisiting algebraic attacks on MinRank and on the rank decoding problem.
Cryptology ePrint Archive, Paper 2022/1031, 2022.
`https://eprint.iacr.org/2022/1031`.

[4] M. Bardet, M. Bros, D. Cabarcas, P. Gaborit, R. Perlner, D. Smith-Tone, J.-P. Tillich, and J. Verbel.
Improvements of algebraic attacks for solving the rank decoding and MinRank problems.
In *Advances in cryptology—ASIACRYPT 2020. Part I*, volume 12491 of *Lecture Notes in Comput. Sci.*, pages 507–536. Springer, Cham, 2020.

[5] J.-C. Faugère, F. Levy-dit Vehel, and L. Perret.
Cryptanalysis of MinRank.
In *Advances in cryptology—CRYPTO 2008*, volume 5157 of *Lecture Notes in Comput. Sci.*, pages 280–296. Springer, Berlin, 2008.

[6] J.-C. Faugère, M. Safey El Din, and P.-J. Spaenlehauer.
Computing loci of rank defects of linear matrices using Gröbner bases and applications to cryptology.
In *ISSAC 2010—Proceedings of the 2010 International Symposium on Symbolic and Algebraic Computation*, pages 257–264. ACM, New York, 2010.

[7] J. Verbel, J. Baena, D. Cabarcas, R. Perlner, and D. Smith-Tone.
On the complexity of "superdetermined" MinRank instances.
In *Post-quantum cryptography*, volume 11505 of *Lecture Notes in Comput. Sci.*, pages 167–186. Springer, Cham, 2019.

References II

[8] P. Gaborit, O. Ruatta, and J. Schrek.
On the complexity of the rank syndrome decoding problem.
IEEE Trans. Inform. Theory, 62(2):1006–1019, 2016.

[9] L. Bettale, J.-C. Faugère, and L. Perret.
Cryptanalysis of HFE, multi-HFE and variants for odd and even characteristic.
Des. Codes Cryptogr., 69(1):1–52, 2013.

[10] A. Kipnis and A. Shamir.
Cryptanalysis of the HFE public key cryptosystem by relinearization.
In *Advances in cryptology—CRYPTO '99 (Santa Barbara, CA)*, volume 1666 of *Lecture Notes in Comput. Sci.*, pages 19–30. Springer, Berlin, 1999.

[11] W. Beullens.
Improved cryptanalysis of UOV and Rainbow.
In *Advances in cryptology—EUROCRYPT 2021. Part I*, volume 12696 of *Lecture Notes in Comput. Sci.*, pages 348–373. Springer, Cham, 2021.

[12] C. Tao, A. Petzoldt, and J. Ding.
Efficient key recovery for all HFE signature variants.
In *Advances in cryptology—CRYPTO 2021. Part I*, volume 12825 of *Lecture Notes in Comput. Sci.*, pages 70–93. Springer, Cham, 2021.

[13] N. T. Courtois.
Efficient zero-knowledge authentication based on a linear algebra problem MinRank.
In *Advances in cryptology—ASIACRYPT 2001 (Gold Coast)*, volume 2248 of *Lecture Notes in Comput. Sci.*, pages 402–421. Springer, Berlin, 2001.

[14] E. Bellini, A. Esser, C. Sanna, and J. Verbel.
MR-DSS – Smaller MinRank-based (ring-)signatures.
Lecture Notes in Comput. Sci., 13512:144–169, 2022.

References III

[15] G. Adj, L. Rivera-Zamarripa, and J. Verbel.
MinRank in the Head: Short signatures from zero-knowledge proofs.
Cryptology ePrint Archive, Paper 2022/1501, 2022.
https://eprint.iacr.org/2022/1501.

[16] T. Feneuil.
Building MPCitH-based signatures from MQ, MinRank, Rank SD and PKP.
Cryptology ePrint Archive, Paper 2022/1512, 2022.
https://eprint.iacr.org/2022/1512.

[17] W. Beullens.
Sigma protocols for MQ, PKP and SIS, and fishy signature schemes.
In *Advances in cryptology—EUROCRYPT 2020. Part III*, volume 12107 of *Lecture Notes in Comput. Sci.*, pages 183–211. Springer, Cham, [2020] ©2020.

[18] Y. Ishai, E. Kushilevitz, R. Ostrovsky, and A. Sahai.
Zero-knowledge from secure multiparty computation.
STOC '07, page 21–30, New York, NY, USA, 2007. Association for Computing Machinery.

[19] A. J. Di Scala and C. Sanna.
Smaller public keys for minrank-based schemes.
ArXiv, 2023.
https://arxiv.org/abs/2302.12447.