KU LEUVEN

fwo

Leveled Isogeny Problems with Hints

PKC '26 - West Palm Beach, USA

S. Das, R. Invernizzi*, P. Kutas, J. Meers

COSIC - KU Leuven

27 may 2026

Outline

1. Introduction
2. Attacks based on Coppersmith method
3. Combinatorial attack

Outline

1. Introduction
2. Attacks based on Coppersmith method
3. Combinatorial attack

Isogeny Problem with Level Structure

- *SIDH attacks:* given $P, Q, \varphi(P), \varphi(Q)$ and $\deg(\varphi)$ can *recover* φ
- countermeasures: publish $\alpha P, \beta Q$ or hide the degree
- *isogeny problem with level structure* [EC:DeFFouPan24]

Leveled Isogeny Problem with Hints

- more generally: fix a subgroup $\Gamma \leq \mathrm{GL}_2(N)$
- publish $M \cdot (\varphi(P), \varphi(Q))^T$ for $M \in \Gamma$
- degree potentially unknown
- hints on $M \rightsquigarrow$ *leveled isogeny problem with hints*

Deriving the equation

- *Weil pairing:* given points (P, Q) of order N and $(\alpha\varphi(P), \beta\varphi(Q))$ it holds

$$e_N(\alpha\varphi(P), \beta\varphi(Q)) = e_N(P, Q)^{\deg(\varphi)\alpha\beta}$$

- e_N is an N-th root of unity (N generally smooth)
- we get an equation

$$\deg(\varphi) \cdot \det(M) - r \equiv 0 \mod N$$

for some known r

Coppersmith Method

Theorem (Coppersmith)

Given a modulus N, a univariate monic polynomial $f(x)$ of degree δ and a bound $X \in \mathbb{N}$, if

$$X < N^{1/\delta}$$

we can compute all $r < X$ such that $f(r) \equiv 0 \bmod N$ in time polynomial in $\log N$ and δ.

Coppersmith Method

- find small roots
- extensive cryptanalytic applications
- can be extended to multivariate polynomials
- requires more setup $\rightsquigarrow$ *automated Coppersmith* [AC:MeeNow23]

Notation

Definition

Let $x \in \mathbb{N}$ be an n-bit integer and $x[i]$ its i-th bit. A k-bit hint for x is a set $\mathbb{H}_{\mathcal{J}}(x) := \{x[j] : j \in \mathcal{J}\}$ where $\#\mathcal{J} = k$.

- most significant bits (MSB): $\mathbb{M}_k(x) := \mathbb{H}(x)$ with $\mathcal{J} = \{1, ..., k\}$.
- least significant bits (LSB): $\mathbb{L}_k(x) := \mathbb{H}(x)$ with $\mathcal{J} = \{n - k + 1, ..., n\}$
- random bits: $\mathbb{R}_k(x) := \mathbb{H}(x)$ with random $\mathcal{J}$ of size k
- continuous bits: $\mathbb{C}_k(x) := \mathbb{H}(x)$ with $\mathcal{J} = \{s + i \bmod n, 0 \leq i < k\}$

Outline

1. Introduction

2. Attacks based on Coppersmith method

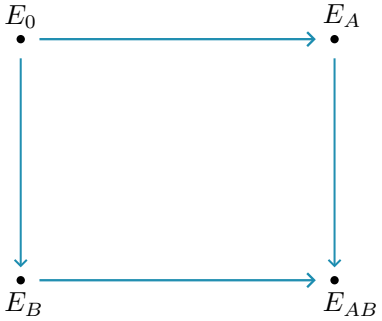
3. Combinatorial attack

Scalar matrices

$$M = \begin{pmatrix} \alpha & \\ & \alpha \end{pmatrix}$$

Applications:

- MSIDH (known degree)
- POKE (4D variant, unknown degree)



Scalar matrices - MSIDH

- public key: $(E', [\alpha]\varphi(P), [\alpha]\varphi(Q))$
- equation:

$$\alpha^2 - 1 \equiv 0 \bmod N$$

Theorem

There exists an efficient adversary agains LIPH with M a scalar matrix given $\mathbb{M}_k(M) = \mathbb{M}_k(\alpha)$ with $k = \lceil (\log N)/2 \rceil$.

Scalar matrices - POKE (4D variant)

- public key: $(E', [\alpha]\varphi(P), [\alpha]\varphi(Q))$
- unknown degree q
- equation:

$$\alpha^2 q - r \equiv 0 \bmod N$$

Theorem

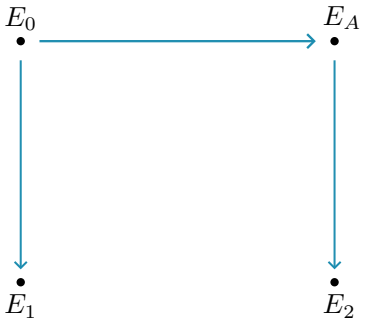
Let $q < N^{6/17}$. There exists an efficient adversary agains LIPH with M a scalar matrix given $\mathbb{M}_k(M) = \mathbb{M}_k(\alpha)$ with $k = \lceil (29 \cdot \log N)/34 \rceil$.

Diagonal matrices

$$M = \begin{pmatrix} \alpha & \\ & \beta \end{pmatrix}$$

Applications:

- FESTA



Diagonal matrices - FESTA

- public key: $(E', [\alpha]\varphi(P), [\beta]\varphi(Q))$
- equation:

$$\alpha\beta \equiv 1 \bmod 2^b$$

Theorem

There exists an efficient adversary against LIPH with M a diagonal matrix given $\mathbb{M}_{2k}(M) = (\mathbb{M}_k(\alpha), \mathbb{M}_k(\beta))$ with $k = \lceil 2b/3 \rceil$.

Outline

1. Introduction

2. Attacks based on Coppersmith method

3. Combinatorial attack

Combinatorial attack - FESTA

- same equation:

$$\alpha\beta \equiv 1 \bmod 2^b$$

- random hints $\mathbb{R}_{b/2}(\alpha), \mathbb{R}_{b/2}(\beta) \rightsquigarrow$ *binary expansion with holes*
- exploit $\alpha\beta \equiv 1 \bmod 2^k$ for all $k \leq b$

Early pruning

$$
\begin{array}{r}
00{*}1 \\
1{**}1 \\
\hline
= \ 1 \mod 2^0
\end{array}
$$

$$Q_0 = \{(1, 1)\}$$

Early pruning

$$
\begin{array}{r}
00{*}1 \\
1{*}{*}1 \\
\hline
= \ 1 \mod 2^1
\end{array}
$$

$$Q_1 = \{(11, 11), (01, 01)\}$$

Early pruning

$$
\begin{array}{r}
00*1 \\
1**1 \\
\hline
= \ 1 \mod 2^2
\end{array}
$$

$$Q_2 = \{(011, 011), (001, 001)\}$$

Early pruning

$$\begin{array}{r} 00{*}1 \\ 1{**}1 \\ \hline = \ 1 \mod 2^3 \end{array}$$

$$Q_3 = \{(0011, 1011)\}$$

Statistical analysis

- Q_k is the list of all solutions at step k
- total complexity: $O(b \cdot \max_k |Q_k|)$
- goal: compute $\max_k \mathbb{E}[|Q_k|] = \mathbb{E}[|Q_b|]$

A dice game

- define $J_k = \log_2(|Q_k|)$
- start with 0 coins, and throw a 4-sided dice (A/B/C/D)
- if A or B, keep coins unchanged
- if C, gain one coin
- if D and you have at least one coin, lose a coin
- J_k is the number of coins after k turns

A dice game

- outcomes after 0 turns: $\{0\}$
- after 1 turn: $\{0, 0, 0, 1\}$
- after 2 turns: $\{0^{10}, 1^5, 2\}$
- ...

Statistical analysis

- $\mathbb{E}[J_k] = O(\sqrt{k})$
- unfortunately, $2^{\mathbb{E}[J_k]} \leq \mathbb{E}[2^{J_k}] = \mathbb{E}[|Q_k|]$ *(Jensen's inequality)*
- indeed $\mathbb{E}[2^{J_k}] \approx 1.125^k$
- on the other hand $\mathbb{P}(J_k < \sqrt{k}) \approx 0.84$
- very practical on concrete instances

Conclusion

- MSIDH: 50% of MSB
- POKE 4D: 86% of MSB
- FESTA: 67% of MSB
- FESTA (combinatorial): 50% random bits, $O(\sqrt{b})$ with 84% probability

Thank you for your attention.
Questions?