F4 traces and index calculus on elliptic curves over extension fields

Vanessa VITSE - Antoine JOUX

Université de Versailles Saint-Quentin, Laboratoire PRISM

October 1, 2010

Part I

Index calculus methods

Hardness of ECDLP

ECDLP

Given $P \in E(\mathbb{F}_q)$ and $Q \in \langle P \rangle$, find x such that $Q = [x]P$

Specific attacks on few families of curves:

Transfer methods

- lift to characteristic zero fields: anomalous curves
- transfer to $\mathbb{F}_{q^k}^*$ via pairings: curves with small embedding degree
- Weil descent: transfer from $E(\mathbb{F}_{q^n})$ to $J_{\mathcal{C}}(\mathbb{F}_q)$ where $\mathcal{C}$ is a genus $g \geq n$ curve

Otherwise, only generic attacks

Trying an index calculus approach

- Index calculus usually the best attack of the DLP over finite fields and hyperelliptic curves
- No known equivalent on $E(\mathbb{F}_p)$, p prime
- Feasible on $E(\mathbb{F}_{p^n})$ and better than Weil descent or generic algorithms

Trying an index calculus approach

- Index calculus usually the best attack of the DLP over finite fields and hyperelliptic curves
- No known equivalent on $E(\mathbb{F}_p)$, p prime
- Feasible on $E(\mathbb{F}_{p^n})$ and better than Weil descent or generic algorithms

Basic outline of index calculus method for DLP

1. define a factor base: $\mathcal{F} = \{P_1, \ldots, P_N\}$
2. relation search: for random (a_i, b_i), try to decompose $[a_i]P + [b_i]Q$ as sum of points in $\mathcal{F}$
3. linear algebra step: once $k > N$ relations found, deduce with sparse algebra techniques the DLP of Q

Results

Original algorithm (Gaudry, Diem)

Complexity of DLP over $E(\mathbb{F}_{q^n})$ in $\tilde{O}(q^{2-\frac{2}{n}})$ but with hidden constant exponential in n^2

- faster than generic methods when $n \geq 3$ and $\log q > C.n$
- sub-exponential complexity when $n = \Theta(\sqrt{\log q})$
- impracticable as soon as $n > 4$

Results

Original algorithm (Gaudry, Diem)

Complexity of DLP over $E(\mathbb{F}_{q^n})$ in $\tilde{O}(q^{2-\frac{2}{n}})$ but with hidden constant exponential in n^2

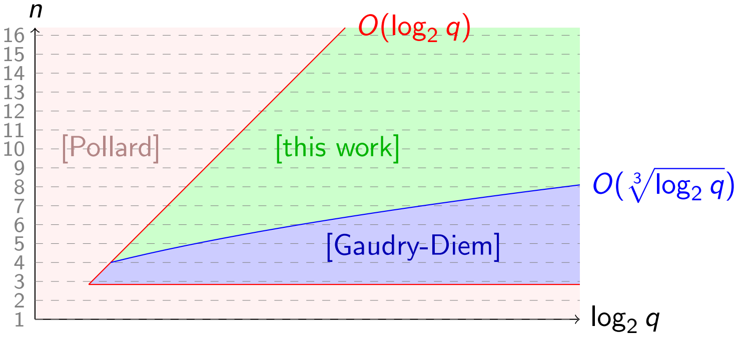
- faster than generic methods when $n \geq 3$ and $\log q > C.n$
- sub-exponential complexity when $n = \Theta(\sqrt{\log q})$
- impracticable as soon as $n > 4$

Our variant

Complexity in $\tilde{O}(q^2)$ but with a better dependency in n

- faster than generic methods when $n \geq 5$ and $\log q \geq 2\omega n$
- faster than Gaudry and Diem's method when $\log q \leq \frac{3-\omega}{2} n^3$
- works for $n = 5$

Comparison of the three attacks of ECDLP over $\mathbb{F}_{q^n}$



Comparison of Pollard's rho method, Gaudry and Diem's attack and our attack for ECDLP over $\mathbb{F}_{q^n}$, $n \geq 1$.

Ingredients of index calculus approaches

Goal

Find at least $\#\mathcal{F}$ decompositions of random combinations $R = [a]P + [b]Q$

What kind of "decomposition" over $E(K)$

Semaev (2004): consider decompositions in a **fixed** number of points of $\mathcal{F}$

$$R = [a]P + [b]Q = P_1 + \ldots + P_m$$

- use the $(m+1)$-th summation polynomial:

$$f_{m+1}(x_R, x_{P_1}, \ldots, x_{P_m}) = 0$$
$$\Leftrightarrow \exists \epsilon_1, \ldots, \epsilon_m \in \{1, -1\}, R = \epsilon_1 P_1 + \cdots + \epsilon_m P_m$$

- Nagao's alternative approach with divisors:
  work with $f \in \mathcal{L}\big((m+1)(\infty) - (R)\big)$ instead

Ingredients of index calculus approaches (2)

Convenient factor base on $E(\mathbb{F}_{q^n})$ – Gaudry (2004)

- Natural factor base: $\mathcal{F} = \{(x, y) \in E(\mathbb{F}_{q^n}) : x \in \mathbb{F}_q\}$, $|\mathcal{F}| \simeq q$
- Weil restriction: decompose along a $\mathbb{F}_q$-linear basis of $\mathbb{F}_{q^n}$

$$f_{m+1}(x_R, x_{P_1}, \ldots, x_{P_m}) = 0 \Leftrightarrow \begin{cases} \varphi_1(x_{P_1}, \ldots, x_{P_m}) = 0 \\ \vdots \\ \varphi_n(x_{P_1}, \ldots, x_{P_m}) = 0 \end{cases} \qquad (\mathcal{S}_R)$$

- Additional trick: symmetrization of the equations

One decomposition trial $\leftrightarrow$ resolution of $\mathcal{S}_R$ over $\mathbb{F}_q$

Ingredients of index calculus approaches (2)

Convenient factor base on $E(\mathbb{F}_{q^n})$ – Gaudry (2004)

- Natural factor base: $\mathcal{F} = \{(x,y) \in E(\mathbb{F}_{q^n}) : x \in \mathbb{F}_q\}$, $|\mathcal{F}| \simeq q$
- Weil restriction: decompose along a $\mathbb{F}_q$-linear basis of $\mathbb{F}_{q^n}$

$$f_{m+1}(x_R, x_{P_1}, \ldots, x_{P_m}) = 0 \Leftrightarrow \begin{cases} \varphi_1(x_{P_1}, \ldots, x_{P_m}) = 0 \\ \vdots \\ \varphi_n(x_{P_1}, \ldots, x_{P_m}) = 0 \end{cases} \qquad (\mathcal{S}_R)$$

- Additional trick: symmetrization of the equations

One decomposition trial $\leftrightarrow$ resolution of $\mathcal{S}_R$ over $\mathbb{F}_q$

Gaudry's original attack

- $m = n$: as many equations as unknowns, $\mathcal{S}_R$ has total degree 2^{n-1}
- Diem: $\mathrm{I}(\mathcal{S}_R)$ has dimension 0 and degree $2^{n(n-1)}$

Example of Gaudry's approach over $\mathbb{F}_{101^3}\left(\simeq\mathbb{F}_{101}[t]/(t^3+t+1)\right)$

- $E : y^2 = x^3 + (44 + 52t + 60t^2)x + (58 + 87t + 74t^2)$, $\#E = 1029583$

base point: $P\left|\begin{array}{l}25+58t+23t^2\\96+69t+37t^2\end{array}\right.$ challenge point: $Q\left|\begin{array}{l}89+78t+52t^2\\14+79t+71t^2\end{array}\right.$

Example of Gaudry's approach over $\mathbb{F}_{101^3}\left(\simeq\mathbb{F}_{101}[t]/(t^3+t+1)\right)$

- $E : y^2 = x^3 + (44 + 52t + 60t^2)x + (58 + 87t + 74t^2)$, $\#E = 1029583$

  base point: $P\left|\begin{array}{l} 25+58t+23t^2 \\ 96+69t+37t^2 \end{array}\right.$ challenge point: $Q\left|\begin{array}{l} 89+78t+52t^2 \\ 14+79t+71t^2 \end{array}\right.$

- random combination of P and Q:

$$R = [658403]P + [919894]Q = \left|\begin{array}{l} 44+57t+55t^2 \\ 8+11t+73t^2 \end{array}\right.$$

Example of Gaudry's approach over $\mathbb{F}_{101^3}\left(\simeq \mathbb{F}_{101}[t]/(t^3+t+1)\right)$

- $E : y^2 = x^3 + (44 + 52t + 60t^2)x + (58 + 87t + 74t^2)$, $\#E = 1029583$

  base point: $P \left| \begin{array}{l} 25+58t+23t^2 \\ 96+69t+37t^2 \end{array} \right.$ challenge point: $Q \left| \begin{array}{l} 89+78t+52t^2 \\ 14+79t+71t^2 \end{array} \right.$

- random combination of P and Q:

$$R = [658403]P + [919894]Q = \left| \begin{array}{l} 44+57t+55t^2 \\ 8+11t+73t^2 \end{array} \right.$$

- compute 4-th summation polynomial with resultant:
  $f_4(X_1, X_2, X_3, X_4) = Res_X\left(f_3(X_1, X_2, X), f_3(X_3, X_4, X)\right)$
  where $f_3 = (X_1 - X_2)^2 X_3^2 - 2((X_1 + X_2)(X_1X_2 + a) + 2b)X_3 + (X_1X_2 - a)^2 - 4b(X_1 + X_2)$

Example of Gaudry's approach over $\mathbb{F}_{101^3}\left(\simeq\mathbb{F}_{101}[t]/(t^3+t+1)\right)$

- $E : y^2 = x^3 + (44+52t+60t^2)x + (58+87t+74t^2)$, $\#E = 1029583$

  base point: $P \left| \begin{array}{l} 25+58t+23t^2 \\ 96+69t+37t^2 \end{array} \right.$ challenge point: $Q \left| \begin{array}{l} 89+78t+52t^2 \\ 14+79t+71t^2 \end{array} \right.$

- random combination of P and Q:

$$R = [658403]P + [919894]Q = \left| \begin{array}{l} 44+57t+55t^2 \\ 8+11t+73t^2 \end{array} \right.$$

- compute 4-th summation polynomial with resultant:
  $f_4(X_1, X_2, X_3, X_4) = Res_X\left(f_3(X_1, X_2, X), f_3(X_3, X_4, X)\right)$
  where $f_3 = (X_1-X_2)^2X_3^2 - 2((X_1+X_2)(X_1X_2+a)+2b)X_3 + (X_1X_2-a)^2 - 4b(X_1+X_2)$

- after partial symmetrisation, solve in $s_1, s_2, s_3 \in \mathbb{F}_{101}$

$$\begin{array}{l} f_4(s_1,s_2,s_3,x_R) = x_R^4 s_2^4 + 93x_R^4 s_1 s_2^2 s_3 \\ \quad +16x_R^4 s_1^2 s_3^2 + \cdots + 94b^3 s_3 = 0 \end{array} \Leftrightarrow \begin{cases} 28s_1^4 + 94s_1^3 s_2 + \cdots + 4s_3 + 69 = 0 \\ 49s_1^4 + 72s_1^3 s_2 + \cdots + 14s_3 + 100 = 0 \\ 32s_1^4 + 97s_1^3 s_2 + \cdots + 50s_3 + 8 = 0 \end{cases}$$

Example of Gaudry's approach over $\mathbb{F}_{101^3}\left(\simeq\mathbb{F}_{101}[t]/(t^3+t+1)\right)$

$$\mathrm{I}(\mathcal{S}_R) = \langle 28s_1^4 + 94s_1^3s_2 + \cdots + 4s_3 + 69, 49s_1^4 + 72s_1^3s_2 + \cdots + 14s_3 + 100, \\ 32s_1^4 + 97s_1^3s_2 + \cdots + 50s_3 + 8\rangle$$

- Gröbner basis of $\mathrm{I}(\mathcal{S}_R)$ for $lex_{s_1>s_2>s_3}$:

$$G = \{s_1 + 33s_3^{63} + 23s_3^{62} + \cdots + 95, s_2 + 80s_3^{63} + 79s_3^{62} + \cdots + 45, \\ s_3^{64} + 36s_3^{63} + 80s_3^{62} + \cdots + 56\}$$

Example of Gaudry's approach over $\mathbb{F}_{101^3}\left(\simeq\mathbb{F}_{101}[t]/(t^3+t+1)\right)$

$$\mathrm{I}(\mathcal{S}_R) = \langle 28s_1^4 + 94s_1^3s_2 + \cdots + 4s_3 + 69, 49s_1^4 + 72s_1^3s_2 + \cdots + 14s_3 + 100, \\ 32s_1^4 + 97s_1^3s_2 + \cdots + 50s_3 + 8\rangle$$

- Gröbner basis of $\mathrm{I}(\mathcal{S}_R)$ for $lex_{s_1>s_2>s_3}$:

$$G = \{s_1 + 33s_3^{63} + 23s_3^{62} + \cdots + 95, s_2 + 80s_3^{63} + 79s_3^{62} + \cdots + 45, \\ s_3^{64} + 36s_3^{63} + 80s_3^{62} + \cdots + 56\}$$

- $V\left(\mathrm{I}(\mathcal{S}_R)\right) = \{(30, 3, 53), (75, 25, 75)\}$

Example of Gaudry's approach over $\mathbb{F}_{101^3}\big(\simeq \mathbb{F}_{101}[t]/(t^3+t+1)\big)$

$$\mathrm{I}(\mathcal{S}_R) = \langle 28s_1^4 + 94s_1^3s_2 + \cdots + 4s_3 + 69, 49s_1^4 + 72s_1^3s_2 + \cdots + 14s_3 + 100, \\ 32s_1^4 + 97s_1^3s_2 + \cdots + 50s_3 + 8\rangle$$

- Gröbner basis of $\mathrm{I}(\mathcal{S}_R)$ for $lex_{s_1>s_2>s_3}$:

$$G = \{s_1 + 33s_3^{63} + 23s_3^{62} + \cdots + 95, s_2 + 80s_3^{63} + 79s_3^{62} + \cdots + 45, \\ s_3^{64} + 36s_3^{63} + 80s_3^{62} + \cdots + 56\}$$

- $V\big(\mathrm{I}(\mathcal{S}_R)\big) = \{(30, 3, 53), (75, 25, 75)\}$
  - $X^3 - 30X^2 + 3X - 53$ irreducible over $\mathbb{F}_{101}[X]$

Example of Gaudry's approach over $\mathbb{F}_{101^3}\left(\simeq\mathbb{F}_{101}[t]/(t^3+t+1)\right)$

$$\mathrm{I}(\mathcal{S}_R) = \langle 28s_1^4 + 94s_1^3s_2 + \cdots + 4s_3 + 69, 49s_1^4 + 72s_1^3s_2 + \cdots + 14s_3 + 100, \\ 32s_1^4 + 97s_1^3s_2 + \cdots + 50s_3 + 8\rangle$$

- Gröbner basis of $\mathrm{I}(\mathcal{S}_R)$ for $lex_{s_1>s_2>s_3}$:

$$G = \{s_1 + 33s_3^{63} + 23s_3^{62} + \cdots + 95, s_2 + 80s_3^{63} + 79s_3^{62} + \cdots + 45, \\ s_3^{64} + 36s_3^{63} + 80s_3^{62} + \cdots + 56\}$$

- $V\left(\mathrm{I}(\mathcal{S}_R)\right) = \{(30,3,53),(75,25,75)\}$
  - $*$ $X^3 - 30X^2 + 3X - 53$ irreducible over $\mathbb{F}_{101}[X]$
  - $*$ $X^3 - 75X^2 + 25X - 75 = (X-4)(X-7)(X-64)$

$$\Rightarrow P_1 \left| \begin{array}{l} 4 \\ 27+34t+91t^2 \end{array}\right. \quad P_2 \left| \begin{array}{l} 7 \\ 58+95t+91t^2 \end{array}\right. \quad P_3 \left| \begin{array}{l} 64 \\ 76+54t+18t^2 \end{array}\right. \quad \text{and } P_1 - P_2 + P_3 = R$$

Example of Gaudry's approach over $\mathbb{F}_{101^3}\left(\simeq\mathbb{F}_{101}[t]/(t^3+t+1)\right)$

$$\mathrm{I}(\mathcal{S}_R) = \langle 28s_1^4 + 94s_1^3s_2 + \cdots + 4s_3 + 69, 49s_1^4 + 72s_1^3s_2 + \cdots + 14s_3 + 100, \\ 32s_1^4 + 97s_1^3s_2 + \cdots + 50s_3 + 8\rangle$$

- Gröbner basis of $\mathrm{I}(\mathcal{S}_R)$ for $lex_{s_1>s_2>s_3}$:

$$G = \{s_1 + 33s_3^{63} + 23s_3^{62} + \cdots + 95, s_2 + 80s_3^{63} + 79s_3^{62} + \cdots + 45, \\ s_3^{64} + 36s_3^{63} + 80s_3^{62} + \cdots + 56\}$$

- $V\left(\mathrm{I}(\mathcal{S}_R)\right) = \{(30, 3, 53), (75, 25, 75)\}$
  - $X^3 - 30X^2 + 3X - 53$ irreducible over $\mathbb{F}_{101}[X]$
  - $X^3 - 75X^2 + 25X - 75 = (X-4)(X-7)(X-64)$

$$\Rightarrow P_1 \Big| {4 \atop 27+34t+91t^2} \quad P_2 \Big| {7 \atop 58+95t+91t^2} \quad P_3 \Big| {64 \atop 76+54t+18t^2} \quad \text{and } P_1 - P_2 + P_3 = R$$

- Number of relations needed: $\#\mathcal{F} = 108 \Rightarrow 109$
- Linear algebra $\rightarrow x = 771080$

Complexity estimates of Gaudry-Diem version

Analysis

- Relation step: solve $n!q$ systems
- Each resolution with Gröbner tools has complexity in $\tilde{O}\left(2^{3n(n-1)}\right)$
- Sparse linear algebra in $\tilde{O}(q^2)$
- "Double large prime" variation $\longrightarrow$ overall complexity in

$$\tilde{O}\left(n! 2^{3n(n-1)} q^{2-2/n}\right)$$

- bottleneck: $\mathrm{I}(\mathcal{S}_{\mathcal{R}})$ has degree $2^{n(n-1)}$
  but most solutions not in $\mathbb{F}_q$
- however adding $x^q - x = 0$ not practical for large q

Example of Nagao's approach over $\mathbb{F}_{101^3}$

Instead of using Semaev's summation polynomials,

- consider $\mathcal{L}\big(4(\infty) - (R)\big)$ with basis $\langle x - x_R, y - y_R, x(x - x_R)\rangle$

Example of Nagao's approach over $\mathbb{F}_{101^3}$

Instead of using Semaev's summation polynomials,

- consider $\mathcal{L}\big(4(\infty) - (R)\big)$ with basis $\langle x - x_R, y - y_R, x(x - x_R)\rangle$
- starting from $f(x,y) = x(x - x_R) + \lambda(y - y_R) + \mu(x - x_R)$

  compute $F(x) = f(x,y)f(x,-y)/(x - x_R)$

  $$\rightarrow F(x) = x^3 + (-\lambda^2 + 2\mu - x_R)x^2 + (-x_R\lambda^2 - 2y_R\lambda + \mu^2 - 2x_R\mu)x \\ -\big((x_R^2 + a)\lambda^2 + 2y_R\lambda\mu + x_R\mu^2\big)$$

  roots of F correspond to x-coord. of the P_i in the decomposition of R

Example of Nagao's approach over $\mathbb{F}_{101^3}$

Instead of using Semaev's summation polynomials,

- consider $\mathcal{L}\big(4(\infty) - (R)\big)$ with basis $\langle x - x_R, y - y_R, x(x - x_R)\rangle$
- starting from $f(x, y) = x(x - x_R) + \lambda(y - y_R) + \mu(x - x_R)$
  compute $F(x) = f(x, y)f(x, -y)/(x - x_R)$
  $$\rightarrow F(x) = x^3 + (-\lambda^2 + 2\mu - x_R)x^2 + (-x_R\lambda^2 - 2y_R\lambda + \mu^2 - 2x_R\mu)x \\ -((x_R^2 + a)\lambda^2 + 2y_R\lambda\mu + x_R\mu^2)$$
  roots of F correspond to x-coord. of the P_i in the decomposition of R
- $x(P_i) \in \mathbb{F}_{101} \Rightarrow F \in \mathbb{F}_{101}[x]$
  find $\lambda, \mu \in \mathbb{F}_{101^3}$ such that $\begin{cases} -\lambda^2 + 2\mu - x_R \in \mathbb{F}_{101} \\ -x_R\lambda^2 - 2y_R\lambda + \mu^2 - 2x_R\mu \in \mathbb{F}_{101} \\ (x_R^2 + a)\lambda^2 + 2y_R\lambda\mu + x_R\mu^2 \in \mathbb{F}_{101} \end{cases}$

Example of Nagao's approach over $\mathbb{F}_{101^3}$

Instead of using Semaev's summation polynomials,

- consider $\mathcal{L}\big(4(\infty)-(R)\big)$ with basis $\langle x-x_R, y-y_R, x(x-x_R)\rangle$
- starting from $f(x,y)=x(x-x_R)+\lambda(y-y_R)+\mu(x-x_R)$
  compute $F(x)=f(x,y)f(x,-y)/(x-x_R)$
  $$\to F(x)=x^3+(-\lambda^2+2\mu-x_R)x^2+(-x_R\lambda^2-2y_R\lambda+\mu^2-2x_R\mu)x \\ -\big((x_R^2+a)\lambda^2+2y_R\lambda\mu+x_R\mu^2\big)$$
  roots of F correspond to x-coord. of the P_i in the decomposition of R
- $x(P_i)\in\mathbb{F}_{101}\Rightarrow F\in\mathbb{F}_{101}[x]$
  $$\text{find } \lambda,\mu\in\mathbb{F}_{101^3} \text{ such that } \begin{cases} -\lambda^2+2\mu-x_R\in\mathbb{F}_{101} \\ -x_R\lambda^2-2y_R\lambda+\mu^2-2x_R\mu\in\mathbb{F}_{101} \\ (x_R^2+a)\lambda^2+2y_R\lambda\mu+x_R\mu^2\in\mathbb{F}_{101}\end{cases}$$
- Weil restriction: solve a quadratic polynomial system with 6 var/eq
  check if resulting F splits in linear factors

Remarks on Nagao's approach

Analysis

- differs from Gaudry only in the polynomial system to solve
- actual resolution slower

$\longrightarrow$ not relevant for the elliptic case

Remarks on Nagao's approach

Analysis

- differs from Gaudry only in the polynomial system to solve
- actual resolution slower

$\longrightarrow$ not relevant for the elliptic case

Practical interest

- in the previous example, eliminating λ, μ in
  $\begin{cases} s_1 = -\lambda^2 + 2\mu - x_R \\ s_2 = -x_R\lambda^2 - 2y_R\lambda + \mu^2 - 2x_R\mu \\ s_3 = (x_R^2 + a)\lambda^2 + 2y_R\lambda\mu + x_R\mu^2 \end{cases}$ yields the partially symmetrized summation polynomial $f_4(s_1, s_2, s_3, x_R)$
  $\longrightarrow$ alternate computation of summation polynomials
- can be easily generalised to hyperelliptic curves whereas Semaev cannot

Joux-V. approach

Decompositions into $m = n - 1$ points

- compute the n-th summation polynomial (instead of $n+1$-th) with partially symmetrized resultant
- solve $\mathcal{S}_R$ with $n-1$ var, n eq and total degree 2^{n-2}
- $(n-1)!q$ expected numbers of trials to get one relation

Joux-V. approach

Decompositions into $m = n - 1$ points

- compute the n-th summation polynomial (instead of $n+1$-th) with partially symmetrized resultant
- solve $\mathcal{S}_R$ with $n-1$ var, n eq and total degree 2^{n-2}
- $(n-1)!q$ expected numbers of trials to get one relation

Computation speed-up

1. $\mathcal{S}_R$ is overdetermined and $\mathrm{I}(\mathcal{S}_\mathcal{R})$ has very low degree
   - resolution with a *degrevlex* Gröbner basis
   - no need to change order (FGLM)
2. Speed up computations with "F4 traces"

Overall complexity in $\tilde{O}\big((n-1)!2^{\omega(n-1)(n-2)}e^{\omega n}q^2\big)$

A toy example over $\mathbb{F}_{101^3}\left(\simeq \mathbb{F}_{101}[t]/(t^3+t+1)\right)$

- E, P and Q as before, random combination of P and Q:

$$R = [357347]P + [488870]Q = \left| \begin{array}{l} 6+63t+58t^2 \\ 11+97t+95t^2 \end{array} \right.$$

A toy example over $\mathbb{F}_{101^3}\left(\simeq\mathbb{F}_{101}[t]/(t^3+t+1)\right)$

- E, P and Q as before, random combination of P and Q:

$$R = [357347]P + [488870]Q = \left| \begin{array}{l} 6+63t+58t^2 \\ 11+97t+95t^2 \end{array} \right.$$

- use 3-rd "symmetrized" Semaev polynomial and Weil restriction:

$$\begin{aligned}
& (s_1^2 - 4s_2)x_R^2 - 2(s_1(s_2+a)+2b)x_R + (s_2-a)^2 - 4bs_1 = 0 \\
\Leftrightarrow\quad & (83t+89t^2)s_1^2 + (89+76t+86t^2)s_1s_2 + (5+98t+45t^2)s_1 \\
& +s_2^2 + (13+69t+29t^2)s_2 + 8 + 96t + 51t^2 = 0 \\
\Leftrightarrow\quad & \begin{cases} 89s_1s_2 + 5s_1 + s_2^2 + 13s_2 + 8 = 0 \\ 83s_1^2 + 76s_1s_2 + 98s_1 + 69s_2 + 96 = 0 \\ 89s_1^2 + 86s_1s_2 + 45s_1 + 29s_2 + 51 = 0 \end{cases}
\end{aligned}$$

A toy example over $\mathbb{F}_{101^3}\left(\simeq \mathbb{F}_{101}[t]/(t^3+t+1)\right)$

$$\mathrm{I}(\mathcal{S}_R) = \langle 89s_1s_2 + 5s_1 + s_2^2 + 13s_2 + 8,\\ 83s_1^2 + 76s_1s_2 + 98s_1 + 69s_2 + 96,\\ 89s_1^2 + 86s_1s_2 + 45s_1 + 29s_2 + 51\rangle$$

- Gröbner basis of $\mathrm{I}(\mathcal{S}_R)$ for *degrevlex*$_{s_1>s_2}$: $G = \{s_1 + 89, s_2 + 49\}$

A toy example over $\mathbb{F}_{101^3}\left(\simeq\mathbb{F}_{101}[t]/(t^3+t+1)\right)$

$$\begin{aligned}\mathrm{I}(\mathcal{S}_R) = \langle 89s_1s_2 + 5s_1 + s_2^2 + 13s_2 + 8,\\ 83s_1^2 + 76s_1s_2 + 98s_1 + 69s_2 + 96,\\ 89s_1^2 + 86s_1s_2 + 45s_1 + 29s_2 + 51\rangle\end{aligned}$$

- Gröbner basis of $\mathrm{I}(\mathcal{S}_R)$ for *degrevlex*$_{s_1>s_2}$: $G = \{s_1 + 89, s_2 + 49\}$
- $V\big(\mathrm{I}(\mathcal{S}_R)\big) = \{(12, 52)\} \rightarrow X^2 - 12X + 52 = (X - 46)(X - 67)$

A toy example over $\mathbb{F}_{101^3}\left(\simeq\mathbb{F}_{101}[t]/(t^3+t+1)\right)$

$$\begin{aligned}\mathrm{I}(\mathcal{S}_R) = \langle 89s_1s_2 + 5s_1 + s_2^2 + 13s_2 + 8,\\ 83s_1^2 + 76s_1s_2 + 98s_1 + 69s_2 + 96,\\ 89s_1^2 + 86s_1s_2 + 45s_1 + 29s_2 + 51\rangle\end{aligned}$$

- Gröbner basis of $\mathrm{I}(\mathcal{S}_R)$ for *degrevlex*$_{s_1>s_2}$: $G = \{s_1 + 89, s_2 + 49\}$
- $V\left(\mathrm{I}(\mathcal{S}_R)\right) = \{(12, 52)\} \to X^2 - 12X + 52 = (X - 46)(X - 67)$

$$\Rightarrow P_1 \left|\begin{array}{l} 46 \\ 29{+}55t{+}56t^2 \end{array}\right. \quad P_2 \left|\begin{array}{l} 67 \\ 20{+}8t{+}59t^2 \end{array}\right. \quad \text{and } P_1 + P_2 = R$$

A toy example over $\mathbb{F}_{101^3}\left(\simeq\mathbb{F}_{101}[t]/(t^3+t+1)\right)$

$$\begin{aligned}\mathrm{I}(\mathcal{S}_R) = \langle 89s_1s_2 + 5s_1 + s_2^2 + 13s_2 + 8,\\ 83s_1^2 + 76s_1s_2 + 98s_1 + 69s_2 + 96,\\ 89s_1^2 + 86s_1s_2 + 45s_1 + 29s_2 + 51\rangle\end{aligned}$$

- Gröbner basis of $\mathrm{I}(\mathcal{S}_R)$ for *degrevlex*$_{s_1>s_2}$: $G = \{s_1 + 89, s_2 + 49\}$
- $V\left(\mathrm{I}(\mathcal{S}_R)\right) = \{(12, 52)\} \to X^2 - 12X + 52 = (X - 46)(X - 67)$

  $\Rightarrow P_1 \left| \begin{array}{l} 46 \\ 29{+}55t{+}56t^2 \end{array} \right. \quad P_2 \left| \begin{array}{l} 67 \\ 20{+}8t{+}59t^2 \end{array} \right.$ and $P_1 + P_2 = R$
- Number of relations needed: $\#\mathcal{F} = 108 \Rightarrow 109$
- Linear algebra $\to x = 771080$

Summary

Comparison between the three approaches

	Gaudry-Diem	Nagao	Joux-V.
nb of points	$m = n$	$m = n$	$m = n - 1$
decomp. trials	$n!q$	$n!q$	$(n-1)!q^2$
features	deg 2^{n-1}	deg 2	deg 2^{n-2}
of $\mathcal{S}_R$	n eq/var	$n(n-1)$ eq/var	n eq, $n-1$ var
$deg(\mathrm{I}(\mathcal{S}_{\mathcal{R}}))$	$2^{n(n-1)}$	$2^{n(n-1)}$	0 (1 exceptionally)
complexity	$n!2^{3n(n-1)}q^{2-2/n}$	$n!2^{2\omega n(n-1)}q^{2-2/n}$	$(n-1)!2^{\omega(n-1)(n-2)}e^{\omega n}q^2$

Part II

F4 traces

Gröbner basis: a tool for polynomial system solving

$\mathrm{I} = \langle f_1, \ldots, f_r \rangle \subset \mathbb{K}[X_1, \ldots, X_n]$ ideal

Gröbner basis

$G = \{g_1, \ldots, g_s\} \subset \mathrm{I}$ is a Gröbner basis of I if

$$\langle LT(g_1), \ldots, LT(g_s) \rangle = LT(\mathrm{I})$$

Buchberger's algorithm

- S-polynomial: $f_1, f_2 \in \mathbb{K}[X_1, \ldots, X_n]$

$$S(f_1, f_2) = \frac{LM(f_1) \vee LM(f_2)}{LT(f_1)} f_1 - \frac{LM(f_1) \vee LM(f_2)}{LT(f_2)} f_2$$

- Buchberger's theorem:

  $G = \{g_1, \ldots, g_s\}$ Gröbner basis $\Leftrightarrow \overline{S(g_i, g_j)}^G = 0$ for all i, j
- Buchberger's algorithm: compute iteratively the remainder by G of every possible S-polynomials and add it to G

Techniques for resolution of polynomial systems

F4: efficient implementation of Buchberger's algorithm

- linear algebra to reduce a large number of critical pairs $(lcm, u_1, f_1, u_2, f_2)$ where $lcm = LM(f_1) \vee LM(f_2)$, $u_i = \frac{lcm}{LM(f_i)}$
- selection strategy (e.g. lowest total degree lcm)
- at each step construct a Macaulay-style matrix containing
  - products $u_i f_i$ coming from the selected critical pairs
  - polynomials from preprocessing phase

$$\begin{array}{cc} & \text{monomial } m \\ & \downarrow \\ \text{polynomial } P \longrightarrow & \begin{pmatrix} & \vdots & \\ \cdots\cdots & \mathrm{coeff}(P, m) & \cdots\cdots \\ & \vdots & \end{pmatrix} \end{array}$$

Macaulay-style matrix

Techniques for resolution of polynomial systems

Standard Gröbner basis algorithms

1. F4 algorithm ('99)
   - fast and complete reductions of critical pairs
   - drawback: many reductions to zero
2. F5 algorithm ('02)
   - elaborate criterion $\longrightarrow$ skip unnecessary reductions
   - drawback: incomplete polynomial reductions

- multipurpose algorithms
- do not take advantage of the common shape of the systems
- knowledge of a prior computation

  $\longrightarrow$ no more reduction to zero in F4 ?

Specifically devised algorithms

Outline of our F4 variant

1. `F4Precomp`: on the first system
   - at each step, store the list of all involved polynomial multiples
   - reduction to zero $\rightarrow$ remove well-chosen multiple from the list
2. `F4Remake`: for each subsequent system
   - no queue of untreated pairs
   - at each step, pick directly from the list the relevant multiples

Former works

- Idea originating from CRT computation of GB over $\mathbb{Q}$
- Traverso 88: precise definition of *Gröbner traces* for the Buchberger algorithm, but behaviour analysis restricted to the rational case

Analysis of F4Remake

"Similar" systems

- parametric family of systems: $\{F_1(y), \ldots, F_r(y)\}_{y \in \mathbb{K}^\ell}$ where $F_1, \ldots, F_r \in \mathbb{K}[Y_1, \ldots, Y_\ell][X_1, \ldots, X_n]$
- $\{f_1, \ldots, f_r\} \subset \mathbb{K}[\underline{X}]$ random instance of this parametric family

Generic behaviour

1. "compute" the GB of $\langle F_1, \ldots, F_r \rangle$ in $\mathbb{K}(\underline{Y})[\underline{X}]$ with F4 algorithm
2. $f_1, \ldots, f_r$ behaves generically if during the GB computation with F4
   - same number of iterations
   - at each step, same new leading monomials $\longrightarrow$ similar critical pairs

Analysis of F4Remake

"Similar" systems

- parametric family of systems: $\{F_1(y), \ldots, F_r(y)\}_{y \in \mathbb{K}^\ell}$
  where $F_1, \ldots, F_r \in \mathbb{K}[Y_1, \ldots, Y_\ell][X_1, \ldots, X_n]$
- $\{f_1, \ldots, f_r\} \subset \mathbb{K}[\underline{X}]$ random instance of this parametric family

Generic behaviour

1. "compute" the GB of $\langle F_1, \ldots, F_r \rangle$ in $\mathbb{K}(\underline{Y})[\underline{X}]$ with F4 algorithm
2. $f_1, \ldots, f_r$ behaves generically if during the GB computation with F4
   - same number of iterations
   - at each step, same new leading monomials $\rightarrow$ similar critical pairs

F4Remake computes successfully the GB of $f_1, \ldots, f_r$
if the system behaves generically

Algebraic condition for generic behaviour

1. Assume $f_1, \ldots, f_r$ behaves generically until the $(i-1)$-th step
2. At step i, F4 constructs
   - M_g =matrix of polynomial multiples at step i for the parametric system
   - M =matrix of polynomial multiples at step i for $f_1, \ldots, f_r$

Algebraic condition for generic behaviour

1. Assume $f_1, \ldots, f_r$ behaves generically until the $(i-1)$-th step
2. At step i, F4 constructs
   - M_g =matrix of polynomial multiples at step i for the parametric system
   - M =matrix of polynomial multiples at step i for $f_1, \ldots, f_r$
3. Reduced row echelon form of M_g and M

$$
\begin{array}{rc}
 & \overbrace{\hphantom{A_{g,0}\,A_{g,0}}}^{LT(M)} \hphantom{A_{g,1}A_{g,1}A_{g,1}A_{g,1}} \\
s\left\{\vphantom{\begin{matrix} A \\ A \end{matrix}}\right. &
\left(\begin{array}{c|c}
\begin{matrix} & A_{g,0} \\ 0 & \end{matrix} & A_{g,1} \\
\hline
A_{g,3} & A_{g,2}
\end{array}\right)
\end{array}
\qquad
\left(\begin{array}{c|c}
\begin{matrix} & A_0 \\ 0 & \end{matrix} & A_1 \\
\hline
A_3 & A_2
\end{array}\right)
$$

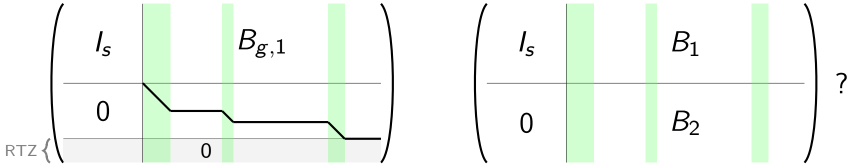
Algebraic condition for generic behaviour

1. Assume $f_1, \ldots, f_r$ behaves generically until the $(i-1)$-th step
2. At step i, F4 constructs
   - M_g =matrix of polynomial multiples at step i for the parametric system
   - M =matrix of polynomial multiples at step i for $f_1, \ldots, f_r$
3. Reduced row echelon form of M_g and M

$$\left(\begin{array}{c|c} I_s & B_{g,1} \\ \hline 0 & B_{g,2} \end{array}\right) \qquad \left(\begin{array}{c|c} I_s & B_1 \\ \hline 0 & B_2 \end{array}\right)$$

Algebraic condition for generic behaviour

1. Assume $f_1, \ldots, f_r$ behaves generically until the $(i-1)$-th step
2. At step i, F4 constructs
   - M_g =matrix of polynomial multiples at step i for the parametric system
   - M =matrix of polynomial multiples at step i for $f_1, \ldots, f_r$
3. Reduced row echelon form of M_g and M



Algebraic condition for generic behaviour

1. Assume $f_1, \ldots, f_r$ behaves generically until the $(i-1)$-th step
2. At step i, F4 constructs
   - M_g =matrix of polynomial multiples at step i for the parametric system
   - M =matrix of polynomial multiples at step i for $f_1, \ldots, f_r$
3. Reduced row echelon form of M_g and M

$$\left(\begin{array}{c|c|c} I_s & 0 & C_{g,1} \\ \hline 0 & I_\ell & C_{g,2} \\ \hline 0 & 0 & 0 \end{array}\right) \qquad \left(\begin{array}{c|c|c} I_s & & B'_1 \\ \hline 0 & B & B'_2 \end{array}\right) ?$$

Algebraic condition for generic behaviour

1. Assume $f_1, \ldots, f_r$ behaves generically until the $(i-1)$-th step
2. At step i, F4 constructs
   - M_g =matrix of polynomial multiples at step i for the parametric system
   - M =matrix of polynomial multiples at step i for $f_1, \ldots, f_r$
3. Reduced row echelon form of M_g and M

$$\left(\begin{array}{c|c|c} I_s & 0 & C_{g,1} \\ \hline 0 & I_\ell & C_{g,2} \\ \hline 0 & 0 & 0 \end{array}\right) \qquad \left(\begin{array}{c|c|c} I_s & & B'_1 \\ \hline 0 & B & B'_2 \end{array}\right)$$

$f_1, \ldots, f_r$ behaves generically at step $i \Leftrightarrow B$ has full rank

Probability of success

Heuristic assumption

B matrices are uniformly random over $\mathcal{M}_{n,\ell}(\mathbb{F}_q)$

- makes sense for $\mathcal{S}_R$ arising from index calculus
- not always valid, but generic behaviour can often be deduced for the first stages of F4

Probability estimates over $\mathbb{F}_q$

Under heuristic assumption:

$$\text{Proba}(\{f_1, \ldots, f_r\} \text{ behaves generically}) \geq c(q)^{n_{step}}$$

- n_{step} = nb of steps during F4 computation for the parametric system
- $c(q) = \prod_{i=1}^{\infty}(1 - q^{-i}) \xrightarrow[q \to \infty]{} 1$

Experimental results: index calculus on $E(\mathbb{F}_{p^5})$

$\lvert p \rvert_2$	est. failure proba.	F4Remake	F4 (Joux-V.)	F4 (Magma)
8 bits	0.11	2.844	5.903	9.660
16 bits	4.4×10^{-4}	3.990	9.758	9.870
25 bits	2.4×10^{-6}	4.942	16.77	118.8
32 bits	5.8×10^{-9}	8.444	24.56	1046

Times in seconds, using a 2.6 GHz Intel Core 2 Duo processor.
Precomputation done in 8.963 s on an 8-bit field.

Experimental results: index calculus on $E(\mathbb{F}_{p^5})$

$\lvert p\rvert_2$	est. failure proba.	F4Remake	F4 (Joux-V.)	F4 (Magma)
8 bits	0.11	2.844	5.903	9.660
16 bits	4.4×10^{-4}	3.990	9.758	9.870
25 bits	2.4×10^{-6}	4.942	16.77	118.8
32 bits	5.8×10^{-9}	8.444	24.56	1046

Times in seconds, using a 2.6 GHz Intel Core 2 Duo processor.
Precomputation done in 8.963 s on an 8-bit field.

Comparison with F5

- both algorithms eliminate all reductions to zero, but
- F5 computes a much larger GB:
  17249 labeled polynomials against **2789** with F4
- signature condition in F5 $\rightarrow$ redundant polynomials

Part III

Application to the Static Diffie-Hellman Problem

Static Diffie-Hellman problem

Observation

Semaev's decomposition into a factor base leads to an oracle-assisted solution of SDHP

Static Diffie-Hellman problem

Given G a finite group, $P, Q \in G$ s.t. $Q = [d]P$ where d secret, and a challenge $X \in G$, compute $[d]X$.

Oracle-assisted SDHP: G finite group and d secret integer

- Initial learning phase: the attacker has access to an oracle which outputs $[d]Y$ for any Y in G
- After a number of oracle queries, the attacker has to compute $[d]X$ for a previously unseen challenge X

Solving SDHP over $G = E(\mathbb{F}_{q^n})$

$\mathcal{F} = \{P \in E(\mathbb{F}_{q^n}) : P = (x_p, y_p), x_p \in \mathbb{F}_q\}$

- Learning phase: ask the oracle to compute $Q = [d]P$ for each $P \in \mathcal{F}$
- Given a challenge X,

1. pick a random integer r coprime with $|G|$ and compute $[r]X$
2. check if $[r]X$ can be written as a sum of m points of $\mathcal{F}$:
   $[r]X = \pm P_1 \pm P_2 \pm \cdots \pm P_m$
3. if $[r]X$ is not decomposable, go back to step 1;
   else output $Y = [s]\left(\sum_{i=1}^{m}[d]P_i\right)$ where $s = r^{-1} \bmod |G|$.

Remarks

- only one decomposition is needed $\rightarrow$ no linear algebra step
  but the q oracle queries are the bottleneck
- Granger (2010): balance the two stages by reducing the factor base à la Harley

An interesting target (joint work with R. Granger)

IPSEC Oakley key determination protocol 'well known group' 3 curve

$\mathbb{F}_{2^{155}} = \mathbb{F}_2[u]/(u^{155} + u^{62} + 1)$ $\qquad G = E(\mathbb{F}_{2^{155}})$ where

$E : y^2 + xy = x^3 + (u^{18} + u^{17} + u^{16} + u^{13} + u^{12} + u^9 + u^8 + u^7 + u^3 + u^2 + u + 1)$

$\#G = 12 * 3805993847215893016155463826195386266397436443$

Remarks

- $\mathbb{F}_{2^{155}} = \mathbb{F}_{(2^{31})^5} \longrightarrow$ curve known to be theoretically weaker than curves over comparable size prime fields
- decomposition as sum of 5 points not realisable
  $\longrightarrow$ Gaudry's approach doesn't work on this curve
- we show that an actual attack with our approach is feasible

Results

Algebraic attack features

For each decomposition trial:

- associate to $[r]X$ the overdetermined symmetrized system $\mathcal{S}_r = \{\varphi_1, \ldots, \varphi_5\} \subset \mathbb{F}_{2^{31}}[s_1, \ldots, s_4]$ of total degree 8
- solve $\mathcal{S}_r$ in $\mathbb{F}_{2^{31}}$ with degrevlex Gröbner basis computation

Expected number of decomposition tests: $4!2^{31} \simeq 5.10^{10}$

Results

Algebraic attack features

For each decomposition trial:

- associate to $[r]X$ the overdetermined symmetrized system $\mathcal{S}_r = \{\varphi_1, \ldots, \varphi_5\} \subset \mathbb{F}_{2^{31}}[s_1, \ldots, s_4]$ of total degree 8
- solve $\mathcal{S}_r$ in $\mathbb{F}_{2^{31}}$ with degrevlex Gröbner basis computation

Expected number of decomposition tests: $4!2^{31} \simeq 5.10^{10}$

Timings

- Magma (V2.15-15): each decomposition trial takes about 1 sec

Results

Algebraic attack features

For each decomposition trial:

- associate to $[r]X$ the overdetermined symmetrized system $\mathcal{S}_r = \{\varphi_1, \ldots, \varphi_5\} \subset \mathbb{F}_{2^{31}}[s_1, \ldots, s_4]$ of total degree 8
- solve $\mathcal{S}_r$ in $\mathbb{F}_{2^{31}}$ with degrevlex Gröbner basis computation

Expected number of decomposition tests: $4!2^{31} \simeq 5.10^{10}$

Timings

- Magma (V2.15-15): each decomposition trial takes about 1 sec
- F4Variant + dedicated optimizations of arithmetic and linear algebra $\rightarrow$ only **22.95 ms** per test on a 2.93 GHz Intel Xeon processor ($\simeq$ 400$\times$ faster than results in odd characteristic)

Conclusion

- A variant of the index calculus method on elliptic curves over small degree extension fields
- F4 traces: a new tool for Gröbner basis computations
  $\rightarrow$ useful as soon as one needs to solve several systems with similar shapes
- Our variant still to slow to threaten DLP on curves with current level of security but efficient on non-standard problems
- In particular, feasible attack on the 'Well Known Group' 3 Oakley curve:
  $\rightarrow$ oracle-assisted SDHP solvable in ≤ 2 weeks with 1000 processors after a learning phase of 2^{30} oracle queries

F4 traces and index calculus on elliptic curves over extension fields

Vanessa VITSE - Antoine JOUX

Université de Versailles Saint-Quentin, Laboratoire PRISM

October 1, 2010