Relating the ECDLP to Other Curves

Mark Bauer

University of Calgary
Department of Mathematics & Statistics
Centre for Information Security and Cryptography

November 2, 2006

Motivation

Two Fundamental Approaches for solving DLP's:

1. Solve it in the given group.
2. Find an homomorphism to some other group where you can easily solve the DLP.

This talk focusses on the latter.

Pure Algebraist

Problem solved - Fundamental Theorem on Decomposition of Finitely Generated Abelian Groups

Cryptographer

Okay, give me the isomorphism.

Hyperelliptic Curves

$$H : y^2 + hy = f$$

1. $h, f \in \mathbb{F}_q[x]$.
2. deg $h \leq g$, deg $f = 2g + 1$ (or $2g + 2$).
3. Nonsingular over $\overline{\mathbb{F}_q}$.
   i.e. no point satisfying the curve equation and the two partials

$$2y + h = 0 \qquad \text{and} \qquad h'y - f' = 0$$

- $g = 1$: Elliptic Curve - group law for points on the curve.
- $g > 1$: There is no group law for points on the curve.

From a Curve to a Group in 3 easy slides

Function Fields

Category Theorist

$\mathbb{F}_q(C) = Hom_{\mathbb{F}_q}(C, \mathbb{P}^1_{\mathbb{F}_q})$

Everybody Else

These are just rational maps (i.e. rational functions in two variables) defined over $\mathbb{F}_q$ from the curve to $\mathbb{F}_q$

Example

Let H be our hyperelliptic curve defined by $y^2 + hy = f$.
$\mathbb{F}_q(H) \cong \mathbb{F}_q(x)[y]/\langle y^2 + hy - f \rangle$

Curves are really classified by their function fields, not how we write them down.

From a Curve to a Group in 3 easy slides

Divisor Group of a Curve

The free abelian group generated by points on the curve.

- Divisor - A finite formal sum of points, e.g.

$$D = \sum_{P \in C(\overline{\mathbb{F}_q})} m_P P, \;\; m_p \in \mathbb{Z}, \;\; m_P = 0 \text{ for almost all } P.$$

- $Div(C)$ denotes the set/group of all such divisors.
- Divisors defined over $\mathbb{F}_q$ - A divisor that is invariant under the natural action of $Gal(\overline{\mathbb{F}_q}/\mathbb{F}_q)$.
- $Div_{\mathbb{F}_q}(C)$ denotes the set of all such divisors.

This group is too large in two respects.

From a Curve to a Group in 3 easy slides

Two subgroups of $Div_{\mathbb{F}_q}(C)$.

- Degree - we can define the degree of the divisor D to be

$$deg\ D = \sum_{P \in C(\overline{\mathbb{F}_q})} m_P.$$

- $Div^0_{\mathbb{F}_q}(C) = \{D \in Div_{\mathbb{F}_q}(C) | deg\ D = 0\}$
- principal divisor - for $f \in \mathbb{F}_q(C)^*$ we define the divisor

$$(f) = \sum_{P \in C} v_P(f) P$$

  where $v_P(f)$ is the order of vanishing or pole of f at P.
- $Prin_{\mathbb{F}_q}(C) = \{(f) \in Div_{\mathbb{F}_q}(C) | f \in \mathbb{F}_q(C)^*\}$

From a Curve to a Group in 4 slides

Exercise

For $f \in \mathbb{F}_q(C)^*$, $deg\ (f) = 0$.

Corollary

$$Prin_{\mathbb{F}_q}(C) \subseteq Div^0_{\mathbb{F}_q}(C)$$

The quotient group is the object we are integerested in

$$Pic^0_{\mathbb{F}_q}(C) = Div^0_{\mathbb{F}_q}(C)/Prin_{\mathbb{F}_q}(C).$$

Warning

This group is often referred to as the Jacobian, just don't let an algebraic geometer hear you say that.

Complexity Notation

Before we talk about complexity of the DLP, we need some notation.

$$L_N[\alpha, \beta] = O\left(exp((\beta + o(1))(\log N)^{\alpha}(\log \log N)^{1-\alpha})\right)$$

- Exponential: $\alpha = 1$, $L_N[1, \beta] = O\left(N^{\beta + o(1)}\right)$
- Polynomial: $\alpha = 0$, $L_N[0, \beta] = O\left((\log N)^{\beta}\right)$
- Subexponential: $0 < \alpha < 1$.

Square Root Algorithms

Generic algorithms can be used to solve any DLP.
Complexity is $L_{|G|}[1, 1/2] = O(\sqrt{|G|})$ (exponential).

Complexity Notation

Before we talk about complexity of the DLP, we need some notation.

$$L_N[\alpha, \beta] = O\left(exp((\beta + o(1))(\log N)^{\alpha}(\log\log N)^{1-\alpha})\right)$$

- Exponential: $\alpha = 1$, $L_N[1, \beta] = O\left(N^{\beta + o(1)}\right)$
- Polynomial: $\alpha = 0$, $L_N[0, \beta] = O\left((\log N)^{\beta}\right)$
- Subexponential: $0 < \alpha < 1$.

Square Root Algorithms

Generic algorithms can be used to solve any DLP.
Complexity is $L_{|G|}[1, 1/2] = O(\sqrt{|G|})$ (exponential).

Complexity Notation

Before we talk about complexity of the DLP, we need some notation.

$$L_N[\alpha, \beta] = O\left(exp((\beta + o(1))(\log N)^{\alpha}(\log\log N)^{1-\alpha})\right)$$

- Exponential: $\alpha = 1,\ L_N[1, \beta] = O\left(N^{\beta + o(1)}\right)$
- Polynomial: $\alpha = 0,\ L_N[0, \beta] = O\left((\log N)^{\beta}\right)$
- Subexponential: $0 < \alpha < 1$.

Square Root Algorithms

Generic algorithms can be used to solve any DLP.
Complexity is $L_{|G|}[1, 1/2] = O(\sqrt{|G|})$ (exponential).

Complexity Notation

Before we talk about complexity of the DLP, we need some notation.

$$L_N[\alpha, \beta] = O\left(exp((\beta + o(1))(\log N)^{\alpha}(\log\log N)^{1-\alpha})\right)$$

- Exponential: $\alpha = 1$, $L_N[1, \beta] = O\left(N^{\beta + o(1)}\right)$
- Polynomial: $\alpha = 0$, $L_N[0, \beta] = O\left((\log N)^{\beta}\right)$
- Subexponential: $0 < \alpha < 1$.

Square Root Algorithms

Generic algorithms can be used to solve any DLP.
Complexity is $L_{|G|}[1, 1/2] = O(\sqrt{|G|})$ (exponential).

Complexity of DLP for curves

Take a curve of genus g defined over $\mathbb{F}_q$. $\#Pic^0_{\mathbb{F}_q}(C) \approx q^g$.

Elliptic Curves - generic algorithms $L_q[1, 1/2] = O(\sqrt{q})$.

Hyperelliptic curves -

index-calculus algorithms $L_{q^g}[1/2, \beta]$, $g > \log q$.

- Adleman-DeMarrais-Huang (1994)
- Müller-Stein-Thiel (1999)
- Enge-Gaudry (2002)

Other curves -

index-calculus algorithms $L_{q^g}[1/3, \beta]$, $g > (\log q)^2$

- Diem - Smooth projective planar curves
- Enge-Gaudry - "$C_{n,d}$" curves.

Complexity of DLP for curves

Take a curve of genus g defined over $\mathbb{F}_q$. $\#Pic^0_{\mathbb{F}_q}(C) \approx q^g$.

Elliptic Curves - generic algorithms $L_q[1, 1/2] = O(\sqrt{q})$.

Hyperelliptic curves -

index-calculus algorithms $L_{q^g}[1/2, \beta]$, $g > \log q$.

- Adleman-DeMarrais-Huang (1994)
- Müller-Stein-Thiel (1999)
- Enge-Gaudry (2002)

Other curves -

index-calculus algorithms $L_{q^g}[1/3, \beta]$, $g > (\log q)^2$

- Diem - Smooth projective planar curves
- Enge-Gaudry - "$C_{n,d}$" curves.

Complexity of DLP for curves

Take a curve of genus g defined over $\mathbb{F}_q$. $\#Pic^0_{\mathbb{F}_q}(C) \approx q^g$.

Elliptic Curves - generic algorithms $L_q[1, 1/2] = O(\sqrt{q})$.

Hyperelliptic curves -

index-calculus algorithms $L_{q^g}[1/2, \beta]$, $g > \log q$.

- Adleman-DeMarrais-Huang (1994)
- Müller-Stein-Thiel (1999)
- Enge-Gaudry (2002)

Other curves -

index-calculus algorithms $L_{q^g}[1/3, \beta]$, $g > (\log q)^2$

- Diem - Smooth projective planar curves
- Enge-Gaudry - "$C_{n,d}$" curves.

Complexity of DLP for curves

Take a curve of genus g defined over $\mathbb{F}_q$. $\#Pic^0_{\mathbb{F}_q}(C) \approx q^g$.

Elliptic Curves - generic algorithms $L_q[1, 1/2] = O(\sqrt{q})$.

Hyperelliptic curves -

index-calculus algorithms $L_{q^g}[1/2, \beta]$, $g > \log q$.

- Adleman-DeMarrais-Huang (1994)
- Müller-Stein-Thiel (1999)
- Enge-Gaudry (2002)

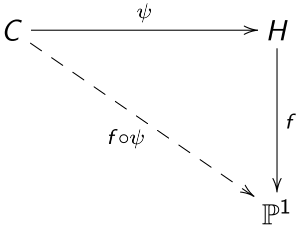
Other curves -

index-calculus algorithms $L_{q^g}[1/3, \beta]$, $g > (\log q)^2$

- Diem - Smooth projective planar curves
- Enge-Gaudry - "$C_{n,d}$" curves.

Maps between curves/function fields

Consider a map $\psi : C \to H$ and $f \in \mathbb{F}_q(H)$.



This gives $f \circ \psi \in \mathbb{F}_q(C)$.
We have an induced map on function fields:

$$\begin{array}{rccc} \psi^* : & \mathbb{F}_q(H) & \longrightarrow & \mathbb{F}_q(C) \\ & f & \mapsto & \psi^*(f) = f \circ \psi \end{array}$$

Co-Norm Map

For a point $P \in H$ Define the divisor $\psi^*(P) \in Div^0_{\mathbb{F}_q}(C)$ by:

- $\psi^*(P) = \sum_{Q \in \psi^{-1}(P)} e_Q Q$,
- e_Q is the order of multiplicity of ψ at Q.

Extend by linearity to get a homomorphism of the divisor groups.

$$\begin{array}{lccc} \psi^* : & Div^0_{\mathbb{F}_q}(H) & \longrightarrow & Div^0_{\mathbb{F}_q}(C) \\ & D = \sum_{P \in H} m_p P & \mapsto & \psi^*(D) = \sum_{P \in H} m_P \psi^*(P) \end{array}$$

We can extend this to a map on principal divisors:

$$\begin{array}{lccc} \psi^* : & Prin_{\mathbb{F}_q}(H) & \longrightarrow & Prin_{\mathbb{F}_q}(C) \\ & (f) & \mapsto & (\psi^*(f)) \end{array}$$

The resulting map $\psi^* : Pic^0_{\mathbb{F}_q}(H) \rightarrow Pic^0_{\mathbb{F}_q}(C)$ is called the Co-Norm Map.

Coverings of Elliptic Curves (simplified)

Consider $2 \neq \text{ char } \mathbb{F}_q \not| \, n$.

$$H_n : y^2 = x^{3n} + Ax^{2n} + Bx^n + C \qquad E : y^2 = x^3 + Ax^2 + Bx + C$$

$$\begin{array}{cccc} \psi : & H_n & \longrightarrow & E \\ & (\alpha, \beta) & \mapsto & (\alpha^n, \beta) \end{array}$$

We can assume $C \neq 0$, so

- ψ is surjective (over $\overline{\mathbb{F}_q}$).
- H_n has genus $\lfloor \frac{3n-1}{2} \rfloor$.

Oh No!

We've excluded elliptic curves over $\mathbb{F}_3$!

$$\begin{array}{rccc} \psi^* : & Pic^0_{\mathbb{F}_q}(E) & \longrightarrow & Pic^0_{\mathbb{F}_q}(H_n) \\ & (\alpha,\beta) - (\infty) & \mapsto & \left(\sum_{i=1}^{n}(\zeta_n^i \alpha', \beta)\right) - n(\infty) \end{array}$$

where:

- $\alpha' \in \overline{\mathbb{F}_q}$ satisfies $(\alpha')^n = \alpha$,
- $\zeta_n \in \overline{\mathbb{F}_q}$ is a primitive n^{th} root of unity.

Claim

ψ^* is injective.

Step 1. Prove each $\psi^*((\alpha,\beta) - (\infty))$ is a distinct divisor.
This is obvious.

Reduced Divisors

Step 2. Prove each $\psi^*((\alpha,\beta)-(\infty))$ is a reduced divisor.
For n odd, every divisor class is represented by a unique divisor.

$$\left(\sum_{i=1}^{n}(\zeta_n^i\alpha',\beta)\right) - n(\infty)$$

Reduced divisors:

1. $\sum_{P\in H\backslash\{\infty\}} m_P \leq g$;
2. $m_P \geq 0\ \forall P \in H\backslash\{\infty\}$;
3. If $P \neq P^\sigma$, then $m_P > 0 \Rightarrow m_{P^\sigma} = 0$;
4. If $P \neq P^\sigma$, then $m_P \leq 1$.

Our divisor:

1. $\sum_{i=1}^{n} 1 = n \leq \lfloor \frac{3n}{2} \rfloor = g$;
2. $m_P \geq 0\ \forall P \in H\backslash\{\infty\}$;
3. $\beta \neq 0$, then $(\zeta_n^i\alpha',\beta)^\sigma \neq (\zeta_n^j\alpha',\beta)$;
4. $\beta = 0$, then $\alpha \neq 0$ $(\zeta_n^i\alpha',\beta)^\sigma \neq (\zeta_n^j\alpha',\beta)$ for $i \neq j$.

Reduced Divisors

Step 2. Prove each $\psi^*((\alpha, \beta) - (\infty))$ is a reduced divisor.
For n odd, every divisor class is represented by a unique divisor.

$$\left(\sum_{i=1}^{n} (\zeta_n^i \alpha', \beta)\right) - n(\infty)$$

Reduced divisors:

1. $\sum_{P \in H \setminus \{\infty\}} m_P \leq g$;
2. $m_P \geq 0 \ \forall P \in H \setminus \{\infty\}$;
3. If $P \neq P^\sigma$, then $m_P > 0 \Rightarrow m_{P^\sigma} = 0$;
4. If $P \neq P^\sigma$, then $m_P \leq 1$.

Our divisor:

1. $\sum_{i=1}^{n} 1 = n \leq \lfloor \frac{3n}{2} \rfloor = g$;
2. $m_P \geq 0 \ \forall P \in H \setminus \{\infty\}$;
3. $\beta \neq 0$, then $(\zeta_n^i \alpha', \beta)^\sigma \neq (\zeta_n^j \alpha', \beta)$;
4. $\beta = 0$, then $\alpha \neq 0$ $(\zeta_n^i \alpha', \beta)^\sigma \neq (\zeta_n^j \alpha', \beta)$ for $i \neq j$.

Reduced Divisors

Step 2. Prove each $\psi^*((\alpha, \beta) - (\infty))$ is a reduced divisor.
For n odd, every divisor class is represented by a unique divisor.

$$\left(\sum_{i=1}^{n} (\zeta_n^i \alpha', \beta)\right) - n(\infty)$$

Reduced divisors:

1. $\sum_{P \in H \backslash \{\infty\}} m_P \leq g$;
2. $m_P \geq 0 \; \forall P \in H \backslash \{\infty\}$;
3. If $P \neq P^\sigma$, then $m_P > 0 \Rightarrow m_{P^\sigma} = 0$;
4. If $P \neq P^\sigma$, then $m_P \leq 1$.

Our divisor:

1. $\sum_{i=1}^{n} 1 = n \leq \lfloor \frac{3n}{2} \rfloor = g$;
2. $m_P \geq 0 \; \forall P \in H \backslash \{\infty\}$;
3. $\beta \neq 0$, then $(\zeta_n^i \alpha', \beta)^\sigma \neq (\zeta_n^i \alpha', \beta)$;
4. $\beta = 0$, then $\alpha \neq 0$
   $(\zeta_n^i \alpha', \beta)^\sigma \neq (\zeta_n^j \alpha', \beta)$ for $i \neq j$.

Reduced Divisors

Step 2. Prove each $\psi^*((\alpha,\beta)-(\infty))$ is a reduced divisor.
For n odd, every divisor class is represented by a unique divisor.

$$\left(\sum_{i=1}^{n}(\zeta_n^i\alpha',\beta)\right)-n(\infty)$$

Reduced divisors:

1. $\sum_{P\in H\setminus\{\infty\}} m_P \le g$;
2. $m_P \ge 0\ \forall P\in H\setminus\{\infty\}$;
3. If $P\ne P^\sigma$, then $m_P>0 \Rightarrow m_{P^\sigma}=0$;
4. If $P\ne P^\sigma$, then $m_P\le 1$.

Our divisor:

1. $\sum_{i=1}^{n} 1 = n \le \lfloor \frac{3n}{2}\rfloor = g$;
2. $m_P \ge 0\ \forall P\in H\setminus\{\infty\}$;
3. $\beta\ne 0$, then $(\zeta_n^i\alpha',\beta)^\sigma \ne (\zeta_n^j\alpha',\beta)$;
4. $\beta = 0$, then $\alpha\ne 0$ $(\zeta_n^i\alpha',\beta)^\sigma \ne (\zeta_n^j\alpha',\beta)$ for $i\ne j$.

Reduced Divisors

Step 2. Prove each $\psi^*((\alpha, \beta) - (\infty))$ is a reduced divisor.
For n odd, every divisor class is represented by a unique divisor.

$$\left(\sum_{i=1}^{n} (\zeta_n^i \alpha', \beta) \right) - n(\infty)$$

Reduced divisors:

1. $\sum_{P \in H \backslash \{\infty\}} m_P \leq g$;
2. $m_P \geq 0 \; \forall P \in H \backslash \{\infty\}$;
3. If $P \neq P^\sigma$, then $m_P > 0 \Rightarrow m_{P^\sigma} = 0$;
4. If $P \neq P^\sigma$, then $m_P \leq 1$.

Our divisor:

1. $\sum_{i=1}^{n} 1 = n \leq \lfloor \frac{3n}{2} \rfloor = g$;
2. $m_P \geq 0 \; \forall P \in H \backslash \{\infty\}$;
3. $\beta \neq 0$, then $(\zeta_n^i \alpha', \beta)^\sigma \neq (\zeta_n^j \alpha', \beta)$;
4. $\beta = 0$, then $\alpha \neq 0$ $(\zeta_n^i \alpha', \beta)^\sigma \neq (\zeta_n^j \alpha', \beta)$ for $i \neq j$.

Consider $\psi : H_n \rightarrow E$ with $g = \lfloor \frac{3n}{2} \rfloor \approx \log q$.

Since $g \approx \log q$ is large enough, we use our subexponential method to solve the DLP:

$$L_{q^g}[1/2, \beta] = O\left(\exp\left((\beta + o(1))(\log q)^{(2)1/2}(2 \log \log q)^{1/2}\right)\right) >>$$

$$O\left(\exp\left((\beta' + o(1))(\log q)^{1}\right)\right) = L_q[1, \beta'], \quad \forall \beta' > 0$$

We've just created an algorithm that is worse than BRUTE FORCE!!

Using Hyperelliptic covers to solve the ECDLP

Consider $\psi : H_n \rightarrow E$ with $g = \lfloor \frac{3n}{2} \rfloor \approx \log q$.

Since $g \approx \log q$ is large enough, we use our subexponential method to solve the DLP:

$$L_{q^g}[1/2, \beta] = O\left(\exp\left((\beta + o(1))(\log q)^{(2)1/2}(2 \log \log q)^{1/2}\right)\right) >>$$

$$O\left(\exp\left((\beta' + o(1))(\log q)^{1}\right)\right) = L_q[1, \beta'], \quad \forall \beta' > 0$$

We've just created an algorithm that is worse than
BRUTE FORCE!!

More Coverings of Elliptic Curves (simplified)

What do we do generically? i.e. consider
Consider $2 \neq \text{ char } \mathbb{F}_q \not| \ n, m$.

$$C_{m,n} : y^{2m} = x^{3n} + Ax^{2n} + Bx^n + C \qquad E : y^2 = x^3 + Ax^2 + Bx + C$$

$$\begin{array}{cccc} \psi : & C_{m,n} & \longrightarrow & E \\ & (\alpha, \beta) & \mapsto & (\alpha^n, \beta^m) \end{array}$$

Take m, n such that $2m = 3n$ or $2m + 1 = 3n$ and $C \neq 0$.

- ψ is surjective (over $\overline{\mathbb{F}_q}$).
- $C_{m,n}$ is smooth (both affine and projective).
- $C_{m,n}$ has genus $\binom{3n-1}{2}$.

Oh No!

Still missing those elliptic curves over $\mathbb{F}_3$!

Induced Maps

Again, we will get

$$\begin{array}{rccc} \psi^* : & Pic^0_{\mathbb{F}_q}(E) & \longrightarrow & Pic^0_{\mathbb{F}_q}(C_{m,n}) \\ & (\alpha, \beta) - (\infty) & \mapsto & \left(\sum_{i=1}^{n} \sum_{j=1}^{m} (\zeta_n^i \alpha', \zeta_m^j \beta') \right) - \psi^{-1}(\infty) \end{array}$$

where:

- $\alpha', \beta' \in \overline{\mathbb{F}_q}$ satisfy $(\alpha')^n = \alpha$ and $(\beta')^m = \beta$.
- $\zeta_n, \zeta_m \in \overline{\mathbb{F}_q}$ are primitive n^{th} and m^{th} roots of unity.

Question

Is ψ^* injective?

- We have no notion of reduced divisors for these curves.

"Sometimes you have to roll a 6 the hard way"

The better half - Norm map

So far, we've only used the contravariance of the *Hom* functor.

$$\begin{array}{rccc} \psi_* : & Div^0_{\mathbb{F}_q}(C_{m,n}) & \longrightarrow & Div^0_{\mathbb{F}_q}(E) \\ & D = \sum_{P \in C_{m,n}} m_p P & \mapsto & \psi_*(D) = \sum_{P \in C_{m,n}} m_P \psi(P) \end{array}$$

We can extend this to a map on principal divisors:

$$\begin{array}{rccc} \psi_* : & Prin_{\mathbb{F}_q}(C_{m,n}) & \longrightarrow & Prin_{\mathbb{F}_q}(E) \\ & (f) & \mapsto & (N_{\mathbb{F}_q(C_{m,n})/\psi^*(\mathbb{F}_q(E))}(f)) \end{array}$$

The resulting map is called the norm map:

$$\psi_* : Pic^0_{\mathbb{F}_q}(C_{m,n}) \to Pic^0_{\mathbb{F}_q}(E)$$

Composing the Norm and co-Norm maps

$$Pic^0_{\mathbb{F}_q}(E) \xrightarrow{\psi^*} Pic^0_{\mathbb{F}_q}(C_{m,n}) \xrightarrow{\psi_*} Pic^0_{\mathbb{F}_q}(E)$$

$$Pic^0_{\mathbb{F}_q}(E) \xrightarrow[\psi_* \circ \psi^*]{} Pic^0_{\mathbb{F}_q}(E)$$

$$P \xmapsto{\psi^*} \sum_{Q \in \psi^{-1}(P)} e_Q Q \xmapsto{\psi^*} \left(\sum_{Q \in \psi^{-1}(P)} e_Q\right) P$$

Note: $\sum_{Q \in \psi^{-1}(P)} e_Q = deg\, \psi = [\mathbb{F}_q(C_{m,n}) : \mathbb{F}_q(E)] = mn$.
Hence, $\psi_* \circ \psi^* = [deg\ \psi]$ on $Pic^0_{\mathbb{F}_q}(E)$.

Condition for ψ^*

- Assume our DLP is in a subgroup of prime order l.
- If $\gcd(mn, l) = 1$, then the DLP is preserved.

Using $L[1/3]$ Algorithms to solve ECDLP

This time we use $\psi : C_{m,n} \to E$ with $g = \binom{3n-1}{2} \approx (\log q)^2$:

Since $mn \approx g \approx (\log q)^2$, $\gcd(mn, l) = 1$ and we can use Diem's algorithm:

$$L_{q^g}[1/3, \beta] = O\left(\exp\left((\beta + o(1))(\log q)^{(3)1/3}(3 \log\log q)^{2/3}\right)\right) >>$$

$$O\left(\exp\left((\beta' + o(1))(\log q)^1\right)\right) = L_q[1, \beta'], \quad \forall \beta' > 0$$

Same thing as before!!

Comments:

- We can do the same for $C_{n,d}$ curves and then use Enge-Gaudry.
- We can use these same tricks to map between other curves.

Using $L[1/3]$ Algorithms to solve ECDLP

This time we use $\psi : C_{m,n} \to E$ with $g = \binom{3n-1}{2} \approx (\log q)^2$:

Since $mn \approx g \approx (\log q)^2$, $\gcd(mn, l) = 1$ and we can use Diem's algorithm:

$$L_{q^g}[1/3, \beta] = O\left(\exp\left((\beta + o(1))(\log q)^{(3)1/3}(3 \log\log q)^{2/3}\right)\right) >>$$

$$O\left(\exp\left((\beta' + o(1))(\log q)^1\right)\right) = L_q[1, \beta'], \quad \forall \beta' > 0$$

Same thing as before!!

Comments:

- We can do the same for $C_{n,d}$ curves and then use Enge-Gaudry.
- We can use these same tricks to map between other curves.

Dividing Lines

New subexponential algorithm for solving DLP's

- Run-time $L_{q^g}[\alpha, \beta]$.
- $g \geq (\log q)^{\delta} \Rightarrow \log q^g \geq (\log q)^{1+\delta}$.

Again, find some embedding of our ECDLP into the new curve.

Exponential:

$$L_{q^g}[\alpha, \beta] = O\left(\exp\left((\beta + o(1))(\log q)^{(1+\delta)\alpha}((1+\delta)\log\log q)^{1-\alpha}\right)\right) >>$$

$$O\left(\exp\left((\beta' + o(1))(\log q)^{(1+\delta)\alpha}(\log\log q)^{1-(1+\delta)\alpha}\right)\right) = L_q[(1+\delta)\alpha, \beta']$$

$$(1+\delta)\alpha \geq 1 \Rightarrow \delta \geq \frac{1-\alpha}{\alpha}$$

Dividing Lines

New subexponential algorithm for solving DLP's

- Run-time $L_{q^g}[\alpha, \beta]$.
- $g \geq (\log q)^{\delta} \Rightarrow \log q^g \geq (\log q)^{1+\delta}$.

Again, find some embedding of our ECDLP into the new curve.

Subxponential:

$$L_{q^g}[\alpha, \beta] = O\left(\exp\left((\beta + o(1))(\log q)^{(1+\delta)\alpha}((1+\delta)\log\log q)^{1-\alpha}\right)\right) <<$$

$$O\left(\exp\left(o(1)(\log q)^{(1+\delta)\alpha+\epsilon}(\log\log q)^{1-(1+\delta)\alpha-\epsilon}\right)\right) = L_q[(1+\delta)\alpha+\epsilon, 0]$$

$$(1+\delta)\alpha < 1 \Rightarrow \delta < \frac{1-\alpha}{\alpha}$$

Review of Index-Calculus Algorithms

Index-calculus

The computational mathematician's answer to the fundamental decomposition of finitely generated abelian groups.

Three basic steps.

1. Construct a factor base;
2. Collect relations;
3. Linear algebra.

For a factor base B, we basically compute the kernel of

$$\phi : \mathbb{Z}^{|B|} \longrightarrow G$$

And explicitly compute

$$\mathbb{Z}^{|B|} / ker\ \phi \cong G$$

Factor Bases

Typical Factor Base

All (or positive proportion) of points defined over $\mathbb{F}_{q^k}$ for all $k < B$.

Probability of finding relation with factor base:

- Probability of finding smooth polynomial of bounded degree with smoothness bound B.

Smaller factor base:

- Let θ be the proportion of $\mathbb{F}_q$-points in factor base;
- Probability that n random $\mathbb{F}_q$-points are in factor base is θ^n.
- Requires $(1/\theta)^n$ such divisors to find one with desired property.

Question

What is one fundamental requirement on the size of the factor base to achieve a subexponential algorithm?

Subexponential factor bases

Fundamental requirement for subexponential index calculus method for ECDLP:

- Size of factor base has to be subexponential in q.

Take factor base of size $L_q[\alpha, \beta]$ with $\alpha < 1$.

What is the probability that a point over $\mathbb{F}_q$ is in factor base?

$$\frac{\#\text{ points in factor base}}{\text{total }\#\text{ of points}} = \frac{L_q[\alpha, \beta]}{L_q[1, 1]}.$$

How many tries to find one such point? $L_q[1, 1]/L_q[\alpha, \beta]$.

Problem

$L_q[1, 1]/L_q[\alpha, \beta]$ dominates $L_q[\alpha', \beta']$ for any $\alpha' < 1$.

NOT subexponential. Answer: Toast.

Subexponential factor bases

Fundamental requirement for subexponential index calculus method for ECDLP:

- Size of factor base has to be subexponential in q.

Take factor base of size $L_q[\alpha, \beta]$ with $\alpha < 1$.

What is the probability that a point over $\mathbb{F}_q$ is in factor base?

$$\frac{\#\text{ points in factor base}}{\text{total }\#\text{ of points}} = \frac{L_q[\alpha, \beta]}{L_q[1, 1]}.$$

How many tries to find one such point? $L_q[1, 1]/L_q[\alpha, \beta]$.

Problem

$L_q[1, 1]/L_q[\alpha, \beta]$ dominates $L_q[\alpha', \beta']$ for any $\alpha' < 1$.

NOT subexponential. Answer: Toast.

Conclussions

1. Find all classes of curves that admit a subexponential algorithm satisfying $\delta \geq \frac{1-\alpha}{\alpha}$ and:
   - Run-time $L_{q^g}[\alpha, \beta]$;
   - $g \geq (\log q)^\delta$.
2. Develop an analogous result for Number Fields? (well, this is easy, but is it worth anything)
3. Can we prove that these maps are injective (enough) when we don't know the group orders involved?
4. Build a better mouse trap - i.e. a fundamentally different index calculus algorithm