Deterministic Hashing to Elliptic and Hyperelliptic Curves

Mehdi Tibouchi

LORIA, 2010-11-08

Outline

Outline

Elliptic curve cryptography

- F finite field of characteristic > 3 (for simplicity's sake).
- Recall that an elliptic curve over F is the set of points $(x, y) \in F^2$ such that:
$$y^2 = x^3 + ax + b$$
(with $a, b \in F$ fixed parameters), together with a point at infinity.
- This set of points forms an abelian group where the Discrete Logarithm Problem and Diffie-Hellman-type problems are believed to be hard (no attack better than the generic ones).
- Interesting for cryptography: for k bits of security, one can use elliptic curve groups of order $\approx 2^{2k}$, keys of length $\approx 2k$. Also come with rich structures such as pairings.

Outline

Hashing to elliptic curves is a problem

- Many cryptographic protocols (schemes for encryption, signature, PAKE, IBE, etc.) involve representing a certain numeric value, often a hash value, as an element of the group $\mathbb{G}$ where the computations occur.
- For $\mathbb{G} = \mathbb{Z}_n^*$, simply taking the numeric value itself $\bmod n$ is usually appropriate.
- However, if $\mathbb{G}$ is an elliptic curve group, this technique has no obvious counterpart; e.g. one cannot put the value in the x-coordinate of a curve point, because only about $1/2$ of possible x-values correspond to actual points.
- Elliptic curve-specific protocols have been developed to circumvent this problem (ECDSA for signature, Menezes-Vanstone for encryption, ECMQV for key agreement, etc.), but doing so with all imaginable protocols is unrealistic.

The traditional solution

- For k bits of security:
    1. concatenate the hash value with a counter from 0 to $k-1$;
    2. initialize the counter as 0;
    3. if the concatenated value is a valid x-coordinate on the curve, i.e. $x^3 + ax + b$ is a square in F, return one of the two corresponding points; otherwise increment the counter and try again.
- Heuristically, the probability of a concatenated value being valid is $1/2$, so k iterations ensure k bits of security.

Problems with this solution

- A natural implementation does not run in constant time: possible timing attacks (especially for PAKE).
- A constant time implementation (always do k steps, compute the Legendre symbol in constant time) is very inefficient, $O(n^4)$.
- Security is difficult to analyze.

Remark: hashing as $H(m) = h(m)G$ where G is a generator of the elliptic curve group is *not* a good idea.

Outline

Supersingular curves

An elliptic curve shape of particular interest is:

$$y^2 = x^3 + b$$

over a field with q elements, with $q \equiv 2 \pmod 3$.
Admits the following deterministic encoding:

$$f : u \mapsto \left((u^2 - b)^{1/3}, u\right)$$

Such a curve is supersingular. Convenient for pairings, but much less secure than ordinary curves for the same key size (because of the MOV attack).

Shallue-Woestijne-Ulas

First deterministic point construction algorithm on ordinary elliptic curves due to Shallue and Woestijne (ANTS 2006). Later generalized and simplified by Ulas (2007).

Based on Skałba's identity: if $g(x) = x^3 + ax + b$, there are rational functions $X_i(t)$ such that

$$g(X_1(t)) \cdot g(X_2(t)) \cdot g(X_3(t)) = X_4(t)^2$$

Hence, on a finite field, at least one of $g(X_1(t)), g(X_2(t)), g(X_3(t))$ is a square.

Gives a deterministic point construction algorithm, which is efficient if $q \equiv 3 \pmod 4$. Considered for implementation in European e-passports.

Icart

Particularly simple deterministic encoding on ordinary elliptic curves when $q \equiv 2 \pmod 3$, presented by Icart at CRYPTO last year. Generalization of the supersingular case.

Defined as $f : u \mapsto (x, y)$ with

$$x = \left(v^2 - b - \frac{u^6}{27}\right)^{1/3} + \frac{u^2}{3} \qquad y = ux + v \qquad v = \frac{3a - u^4}{6u}$$

This simple idea sparked new research into the subject of deterministic hashing into elliptic curves.

Outline

Questions we solved

The previous constructions do not completely address the problem of constructing "good hash functions" to elliptic curves, and open up a series of related questions.

We solved some of them.

- Icart's conjecture: Icart observed that his function did not map to the whole elliptic curve, and conjectured that the image comprised only about 5/8 of all points. Is this true? What about the SWU function?
- In particular if f is Icart's function and h is a random oracle into the base field, $m \mapsto f(h(m))$ is easily distinguished from a random oracle. Can f still be used to construct a random oracle to the curve?
- Extension to hyperelliptic curves: can we construct good hash functions? Note that we should map to the Jacobian variety, not the curve itself!

Outline

Statement

E elliptic curve over $\mathbb{F}_q$, with $q \equiv 2 \pmod 3$, and $f : \mathbb{F}_q \to E(\mathbb{F}_q)$ Icart's deterministic encoding.

Conjecture (Icart)

There exists a universal constant C such that:

$$\left| \# f(\mathbb{F}_q) - \frac{5}{8} \# E(\mathbb{F}_q) \right| \leq C \sqrt{q}$$

Icart's paper presented a heuristic argument to justify the constant $5/8$. The conjecture was proved independently by Farahashi, Shparlinski and Voloch, and by Fouque and T.

A consequence of this conjecture is that f is neither injective nor surjective. However, $(u, v) \mapsto f(u) + f(v)$ is a surjective encoding function for q large enough.

Proof idea I

- A key fact is that u maps to (x, y) under f if and only if:

$$u^4 - 6xu^2 + 6yu - 3a = 0$$

- Hence, the problem is to count the points (x, y) on the curve such that the polynomial $P(u) = u^4 - 6xu^2 + 6yu - 3a$ has at least one root in $\mathbb{F}_q$.
- P can be seen as a polynomial over the function field $\mathbb{F}_q(x, y)$ of E, and the problem is to count places of degree 1 in this function field where the reduction of P has a root.
- Mathematicians have a powerful tool to tackle this kind of problems: the Chebotarev density theorem, which says that the "density" of places at which P reduces into a product of factors of given degrees is determined by the number of permutations with the corresponding cycle decomposition in the Galois group of P.

Proof idea II

At this point, completing the proof is a technical exercise:

- Show that P is an irreducible polynomial with Galois group S_4 (hard part).
- Count the number of permutations in S_4 with a fixed point (there are $1 + 6 + 8 = 15$ of them).
- Deduce that the density of places in $\mathbb{F}_q(x, y)$ at which P has a root is $15/24 = 5/8$.
- Apply an effective version of Chebotarev's density theorem to get the same result with a $O(\sqrt{q})$ error term for places of degree 1 (this gives Icart's conjecture).

In the paper with Fouque, we also show how the technique generalizes to other encoding functions with different Galois groups such as a simplified version of SWU (Galois group D_8, constant $3/8$).

Outline

Statement

Since Icart's function f (or SWU, etc.) only covers a limited fraction of points on the curve, $m \mapsto f(h(m))$ is not a well-behaved hash function: easy to distinguish from a random oracle.

While some schemes may not require randomness or collision resistance, it is desirable in general to have a construction indistinguishable from a random oracle, in the ROM for some $\mathbb{F}_q$-valued hash function h.

Coron and Icart showed it suffices to have an encoding function $F : S \to E(\mathbb{F}_q)$ from some set S, such that F^{-1} is efficiently computable, and that if s is uniformly distributed in S, the distribution of $F(s)$ is *statistically indistinguishable* from uniform in $E(\mathbb{F}_q)$.

Admissible encodings

- An encoding verifying the statistical indistinguishability property is called *admissible* by Coron and Icart (generalization of a previous definition by Boneh-Franklin).
- Using Maurer's indifferentiability framework, they show that if F is admissible, then $m \mapsto F(h(m))$ can be used as a random oracle in the ROM for h.
- An example of such an admissible encoding is $F(u, v) = f(u) + v \cdot G$ with G a generator of the elliptic curve group. The addition of vG acts as a "one-time pad" to mask the irregularities of f, and ensure statistical indistinguishability. Hence

$$H(m) = f(h_1(m)) + h_2(m) \cdot G$$

is a "good" hash function. Also works with SWU, with characteristic 2 counterparts, etc. However, the multiplication makes it slow.

Efficient indifferentiable hashing with Icart

- Since the "easy" admissible encoding is slow, we proposed the following much more efficient solution:

$$F(u, v) = f(u) + f(v)$$

- We know as a corollary of Icart's conjecture that this is surjective, but we can also prove statistical indistinguishability with some algebraic geometry machinery.
- Basic idea: for some given point ϖ on E, the set of (u, v) in the affine plane such that $F(u, v) = \varpi$ forms an algebraic curve of bounded genus, that will usually be irreducible.
- In that case, the Hasse-Weil bound ensures that:

$$F^{-1}(\varpi) = q + O(\sqrt{q})$$

  giving admissibility.
- Making the idea work involves beautiful algebraic geometry (such as intersection theory on the surface $C \times C$, where C is the quartic covering of E defined by the polynomial P from the previous section).

Efficient indifferentiable hashing, general case

- Previous geometric method: works well for Icart's function, but difficult to generalize (for SWU, multiple components with complicated interplay; in higher genus, simply horrible).
- We recently proposed a much simpler technique based on character sums. We call an encoding $f : \mathbb{F}_q \to E(\mathbb{F}_q)$ well-distributed when for any nontrivial character χ of $E(\mathbb{F}_q)$:

$$\left| \sum_{u \in \mathbb{F}_q} \chi\big(f(u)\big) \right| \leq B\sqrt{q}$$

- Completely formal to show that if f is well-distributed, $(u, v) \mapsto f(u) + f(v)$ is admissible: write down the statistical distance.
- Relatively easy to show that a given deterministic encoding is well-distributed: the character sum can be interpreted as an Artin character sum on the covering curve C, which is bounded by $(2g_C + 2)\sqrt{q}$ according to a theorem by Weil (corollary of the Riemann hypothesis for curves).

Outline

A simple encoding to hyperelliptic curves

- The first deterministic point-encoding function to hyperelliptic curves of a very special shape, $y^2 = x^{2g+1} + ax + b$ was proposed by Ulas, as a generalization of the Shallue-van de Woestijne technique.
- More recently, Kammerer, Lercier and Renault proposed several Icart-like encoding functions to hyperelliptic curves of somewhat complicated but more general shape.
- We proposed a much simpler encoding function to the family of odd hyperelliptic curves $H : y^2 = g(x)$ where g is an odd polynomial, over $\mathbb{F}_q$, $q \equiv 3 \pmod 4$. This encoding has many nice properties.
- Easy to describe: for any $t \in \mathbb{F}_q$, one of $g(t)$ or $g(-t)$ is a square; define the point $f(t)$ as $y^2 = g(\pm t)$ accordingly, and set x such that $f(-t) = -f(t)$.
- This encoding is very simple to compute, and is (almost) a bijection $f : \mathbb{F}_q \to H(\mathbb{F}_q)$. In particular, it is admissible.

Encoding and hashing to the Jacobian

- The group used in hyperelliptic curve cryptography is the Jacobian J of the curve: it is this group that we should seek to encode or hash to.
- Hashing at least is easy. All previously mentioned encodings to hyperelliptic curves H are also well-distributed, in the sense that for all nontrivial characters χ of $J(\mathbb{F}_q)$:

$$\left| \sum_{u \in \mathbb{F}_q} \chi\big(f(u)\big) \right| \leq B\sqrt{q}$$

- Admissibility of $(u_1, \dots, u_s) \mapsto f(u_1) + \cdots + f(u_s)$ again follows formally, as soon as s is greater that the genus g of H.
- Our encoding to odd hyperelliptic curves allows a different construction: an injective encoding to the Jacobian. Take $(u_1, \dots, u_g) \mapsto f(u_1) + \cdots + f(u_g)$, from the set of tuples such that $u_1 < \cdots < u_g$ and $u_i + u_j \neq 0$. This is injective and reaches a fraction of aboug $1/g!$ points of $J(\mathbb{F}_q)$. Necessary for e.g. El Gamal encryption.

Outline

Some open problems

- Encoding to some missing types of curves: Baretto-Naehrig elliptic curves, more hyperelliptic curves...
- Bounded leakage. It is easy to distinguish the output of the whole Icart's function from a uniform distribution. And the same is true with just the x-coordinate. However, if one only has the top half bits of x, the output is uniform. At which point between these two extremes can a distinguisher still work?
- Injective deterministic encodings: they are probably even more useful than hash functions, but have only been constructed on a few curves. Extend this to at least ordinary elliptic curves. A proper formalization of desired properties would be desirable.
- Impossibility results in generic groups.

Summary

- Hashing and encoding to (hyper)elliptic curves are problems worth looking into.
- Some good candidates are known, but there is still a lot of work to do.
- Plenty of nice problems, from pure mathematics to applied crypto.

Thank you!