Cryptographic Applications of Hyperelliptic Function Fields

Michael J. Jacobson, Jr.

jacobs@cpsc.ucalgary.ca

UNIVERSITY OF CALGARY

UNCG Summer School in Computational Number Theory 2016: Function Fields

Cryptography in Hyperelliptic Function Fields

Public-key cryptography: secret key exchange and digital signatures

Many widely-used protocols use arithmetic in a finite cyclic group G that should satisfy:

- efficient arithmetic (eg. non-adjacent form for exponentiation)
- discrete logarithm problem seems difficult

We have seen that $G = Cl^0(F)$ can be a good candidate (especially genus 1 and 2)

- efficient arithmetic (eg. non-adjacent form for exponentiation)
- discrete logarithm problem seems difficult

Diffie-Hellman Key Exchange

Public system information: generator P of G of prime order n

- A computes aP (a random in $[1, n-1|$) and sends to B
- B computes bP (b random in $[1, n-1]$) and sends to A
- A and B compute $K = a(bP) = b(aP) = abP$

Adversary's goal: find the secret key K given P, aP, bP

- Equivalent to Diffie-Hellman problem
- DLP in G must be hard (necessary, not known whether this is sufficient)

Real World Security (2 Examples)

When using elliptic curves, group elements must be verified as being on the given curve:

- Arithmetic on $E : y^2 = x^3 + Ax + B$ does *not* require the use of B
- Malicious participant in Diffie-Hellman protocol can send a point P' with *small order* on an elliptic curve $E : y^2 = x^3 + Ax + B'$.
- Partner's secret scalar can be computed modulo the order of P' exhaustively

aP (respectively bP) must be authenticated as coming from A (respectively B)

- otherwise, man-in-the-middle attack (intercept message, replace with attacker's own) completely breaks this

Digital Signatures

Digital signature: a means by which the recipient of a message can authenticate the identity of the sender. It should have two properties:

1. Only the sender can produce his signature.
2. *Anyone*, including an arbitrator, should be easily able to verify the validity of the signature.

Important application of public-key cryptography:

- User generates a pair of keys, one public (known to everyone) and one private
- Use private key to generate signatures (only user can do this!)
- Use public key to verify signatures (anyone can do this!)

One example: Digital Signature Algorithm (DSA)

DSA Signature Generation

Public information:

- generator P of elliptic curve $E(\mathbb{F}_q)$, $n = |E(\mathbb{F}_q)|$
- public cryptographic hash function H (hashes messagese to $[1, \dots, n-1]$)

Signer's input: private key $d \in [1, n-1]$, message m

1. Select $k \in [1, n-1]$ at random.
2. Compute $kP = (x_1, y_1)$ and convert x_1 to an integer $\overline{x}_1$.
3. Compute $r = \overline{x}_1 \bmod n$. If $r = 0$ go to Step 1.
4. Compute $e = H(m)$.
5. Compute $s = k^{-1}(e + dr) \bmod n$. If $s = 0$ then go to Step 1.
6. Return signature (r, s).

DSA Signature Verification

Verifier's input: public key $Q = dP$, signature (r, s)

1. Verify that r and s are integers in the interval $[1, n-1]$. If any verification fails return "reject."
2. Compute $e = H(m)$.
3. Compute $w = s^{-1} \bmod n$.
4. Compute $u_1 = ew \bmod n$ and $u_2 = rw \bmod n$.
5. Compute $X = u_1 P + u_2 Q$.
6. If $X = \infty$ return "reject."
7. Convert the x-coordinate x_1 of X to an integer $\overline{x}_1$; compute $v = \overline{x}_1 \bmod n$.
8. If $v = r$ return "accept," otherwise return "reject."

Why This Works

Idea of verification: should have $X = kP$ if the signature is valid.

A legitimate signature has $s \equiv k^{-1}(e + dr) \pmod{n}$. Thus

$$
\begin{aligned}
k &\equiv s^{-1}(e + dr) \pmod{n} \\
&\equiv s^{-1}e + s^{-1}dr \pmod{n} \\
&\equiv we + wrd \pmod{n} \\
&\equiv u_1 + u_2 d \pmod{n}
\end{aligned}
$$

and thus $X = u_1 P + u_2 Q = (u_1 + u_2 d)P = kP$.

Security of ECDSA

Adversary should not be able to forge a valid signature for any message.

Necessary (not sufficient!) conditions:

1. intractability of ECDLP,
2. secure hash function

Other issues:

1. k must be unpredictable (can recover private key if k is known)
2. k must never be re-used (can recover private key otherwise)

Bilinear Pairings

Recall Tate-Lichtenbaum pairing: let E be an elliptic curve over $\mathbb{F}_q$ such that E has a point of order n and $n \mid q-1$. There exists an efficiently-computatable pairing

$$\tau_n : E(\mathbb{F}_q)[n] \times E(\mathbb{F}_q)/nE(\mathbb{F}_q) \to \mu_n \subseteq \mathbb{F}_{q^k}$$

Properties:

- bilinear, i.e., $\tau_n(aP, bQ) = \tau_n(P, Q)^{ab}$
- $\tau_n(P, P) = \zeta_n \in \mathbb{F}_{q^k}$ (primitive nth root of unity)

Many uses in cryptographic protocols. Examples:

- Boneh/Franklin (2001): ID-based cryptography
- Boneh/Lynn/Shacham (2004): short signatures
- *many* others

E.g. Tripartite Key Exchange

Joux (2000): three participants can obtain a shared secret key in just one round of communication

Public system information: generator P of $E(\mathbb{F}_q)$ of prime order n

- Participants A, B, and C each choose random integers a, b, and c coprime to n and respectively compute and broadcast $P_A = aP$, $P_b = bP$, and $P_C = cP$.
- A computes key as $k = \tau_n(P_b, P_c)^a = \tau_n(bP, cP)^a = \tau_n(P, P)^{abc} = \zeta_n^{abc}$
- B computes $k = \tau_n(P_a, P_c)^b$
- C computes $k = \tau_n(P_a, P_b)^c$

Security

Need the DLP to be hard in both $E(\mathbb{F}_q)$ *and* $\mathbb{F}_{q^k}$

Recent years have seen many advances in solving the DLP in $\mathbb{F}_{q^k}$

- Joux/Granger/Kleinjung/Zumbrägel (2014): quasi-polynomial time for small characteristic
- Kim/Barbelescu (2016): advances for $\mathbb{F}_{p^k}$, p a large prime

Consequences:

- Can't use characteristic 2
- Security of commonly-used curves in odd characteristic is being reassessed

May need new curves over bigger fields, and even faster curve/pairing arithmetic to compensate - on-going research!

Finding Cryptographically-Suitable Function Fields

For DSA, the order of G is required (also for Diffie-Hellman, to check security properties)

Two approaches:

1. Generate random function fields, compute (and test) class number.
2. Construct function fields with a given class number

Class Number Computation

Very efficient for elliptic curves (point counting):

- p-adic methods (odd char: Satoh 2000, char 2: AGM, Mestre 2000)
- SEA algorithm (Schoof 1985, Atkin/Elkies 1990's) — polynomial-time

Good computation results for special types of more general curves (some hyperelliptic, Picard, radical cubic, superelliptic, ...)

Hard in general, especially for odd characteristic (even genus 2!)

Class Number Computation: Higher Genus

Small characteristic (i.e. p-adic) methods based on Satoh & Mestre

- Monsky-Washnitzer cohomology (Kedlaya 2001)
- Deformation theory (Lauder 2004, 2006)
- Canonical lifts (Satoh 2003)

Some adaptations to medium and larger characteristic

Large characteristic methods based on SEA

- Pila 1990, Couveignes 1996, Adleman/Huang 2001

Generic algorithms (baby step giant step, Pollard kangaroo, using Euler products)

Index calculus methods — compute the class *group* as well

Elliptic Curve Constructions over $\mathbb{F}_q$

Curves with prescribed group order over finite fields (Bröker 2007)

- heavily use theory of *complex multiplication*

Pairing-friendly curves (low embedding degree)

- Supersingular curves
- Constructions for ordinary curves, eg. Barreto/Naehrig (2005), Miyaji/Nakabayashi/Takano (2001)

Curves with many rational points — $\mathbb{F}_q$ setting useful for coding theory

Rational points on a given elliptic curve in a number field

Theorem (Mordell's and Mazur's Theorems)

Let E be an elliptic curve over $\mathbb{Q}$ with group of $\mathbb{Q}$-rational points $E(\mathbb{Q})$.

- *$E(\mathbb{Q})$ is finitely generated (Mordell 1922)*
- *The torsion of $E(\mathbb{Q})$ is isomorphic to $\mathbb{Z}/n\mathbb{Z}$ with $1 \leq n \leq 10$ or $n = 12$, or $\mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/2n\mathbb{Z}$ with $1 \leq n \leq 4$ (Mazur 1977)*

Problems:

- Find **curves of a given (large?) rank** (Birch–Swinnerton-Dyer conjecture, Elkies 2006 — rank 28)
- Find **curves with prescribed torsion**

Other Constructions

Hyperelliptic function fields with class groups of large ℓ-rank (Bauer et al. 2008, Berger et al. 2011, Jacobson et al. 2014, S. Stein 2014)

All degree n function fields of a given Galois group G and discriminant divisor D ($n = p$, $G = D_p$: Weir et al. 2013; $n = 3$, D square-free: Jacobson et al. 2014)

Function field tabulation ($n = 3$, D square-free: Rozenhart et al. 2008, 2009, 2012; $n = p$, $G = D_p$: Weir et al. 2013)

Conclusion

Have now seen a glimpse of some theory, algorithms, applications (crypto) and on-going research in:

- efficient ideal/divisor arithmetic
- different curve models
- discrete logarithm computation
- invariant / class number computation
- constructive methods

Plenty more out there (e.g. isogeny and endomorphism ring computation, cryptography with isogenies, applying isogenies to map DLP to weak curve)

- Lots of open and interesting computational problems!
- Lots of work to be done!