Twenty-Three Years of Elliptic Curve Cryptography

Alfred Menezes

University of Waterloo

September 3 2008

Further Reading

A. H. Koblitz, N. Koblitz, A. Menezes,

“Elliptic curve cryptography: The serpentine course of a paradigm shift”

Available on my web site very soon.

Public-Key Cryptography Before 1985

Two public-key families emerged that were commercially viable:

1. *RSA*
   (based on integer factorization)

2. *Diffie-Hellman/ElGamal*
   (based on the finite field discrete logarithm problem)

RSA

Invented by Rivest, Shamir and Adleman in *1977*.

Security is based on the hardness of the problem of *factoring an integer* n that is the product of two primes p and q of the same bitlength.

In 1985, n could be factored in *subexponential time* $2^{(\log n)^{1/2}}$ (using the 'quadratic sieve' algorithm).

Consequence: For a 64-bit level of security, one needed $n \approx 2^{512}$.

Fully exponential time: $2^{c(\log n)} = n^c$ [terribly inefficient]

Subexponential time: $2^{(\log n)^c}$ [inefficient, but not terribly so]

Polynomial time: $(\log n)^c$ [efficient]

Finite Fields

Let $q = p^m$ be a prime power.

There is a unique finite field $\mathbb{F}_q$ of size q.

Example: *Prime Fields* $\mathbb{F}_p$
Integers modulo p: $\mathbb{Z}_p = \{0, 1, 2, \ldots, p-1\}$.

Example: *Binary Fields* $\mathbb{F}_{2^m}$
Binary polynomials modulo an irreducible polynomial.

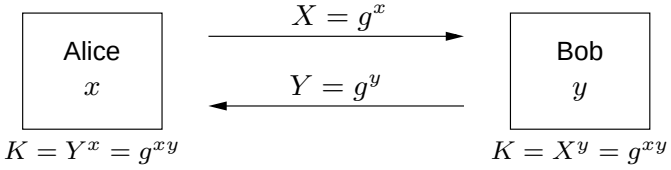
The nonzero elements of $\mathbb{F}_q$ form a cyclic group $\mathbb{F}_q^*$ of size $q-1$.

Let g be an element of order n in $\mathbb{F}_q$ (where n divides $q-1$).
Then $\langle g \rangle = \{g^0, g^1, g^2, \ldots, g^{n-1}\}$ is a group of size n.

Diffie-Hellman

Invented by Diffie and Hellman in *1976*.

Let g be an element of order n in $\mathbb{F}_q$, and let $\mathbb{G} = \langle g \rangle$.



The *finite field DLP*: Given $h \in \mathbb{G}$, compute the integer $z \in [0, n-1]$ such that $h = g^z$.
In 1985, the DLP in $\mathbb{F}_q^*$ could be solved in *subexponential time* $2^{(\log q)^{1/2}}$ (using 'index-calculus' algorithms) and in *fully exponential time* $\sqrt{n}$ (using Pollard's rho method).

Consequence: For a 64-bit level of security, one needed $q \approx 2^{512}$ and $n \approx 2^{128}$.

Discrete Logarithm Problem (DLP)

The DLP can be generalized to arbitrary *cyclic groups*.

Let $\mathbb{G} = \langle g \rangle$ be a group of prime order n.

The *DLP in* $\mathbb{G}$ is: Given $h \in \mathbb{G}$, compute the integer $z \in [0, n-1]$ such that $h = g^z$.

The best generic algorithm for solving the DLP is *Pollard's rho method* which has a running time $\approx \sqrt{n}$.

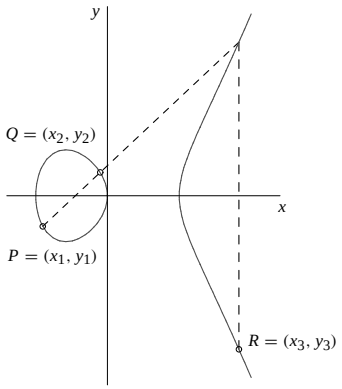
Consequence: For a 64-bit level of security, one needs $n \approx 2^{128}$.

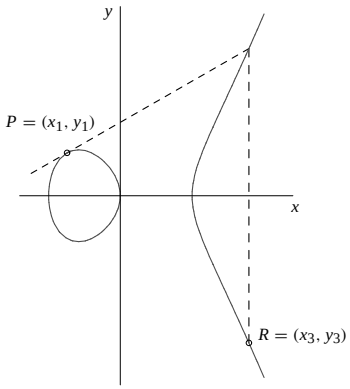
Elliptic Curves

An *elliptic curve* E defined over a finite field $\mathbb{F}_q$ is given by an equation:

$$E/\mathbb{F}_q : Y^2 = X^3 + aX + b.$$

The solutions (x, y), where $x, y \in \mathbb{F}_q$, to the equation form an *group*. This group $E(\mathbb{F}_q)$ has size $\approx q$.





Elliptic Curve Cryptography

In *1985*, Koblitz and Miller proposed using the group of points on an elliptic curve in public-key cryptography.

Their proposal was motivated by two factors:

1. The lack of a subexponential-time algorithm for the DLP on elliptic curves (ECDLP).

   Consequence: For a 64-bit level of security, one needs $q \approx 2^{128}$.

2. The large number of elliptic curves (approximately $2q$) for each finite field $\mathbb{F}_q$.

Hyperelliptic Curves

A *hyperelliptic curve* C of *genus* g defined over a finite field $\mathbb{F}_q$ is given by an equation:

$$C/\mathbb{F}_q : Y^2 = X^{2g+1} + a_{2g}X^{2g} + \cdots + a_1X + a_0.$$

Note that a hyperelliptic curve of genus $g = 1$ is precisely an elliptic curve. [Elliptic curve: $Y^2 = X^3 + aX + b$]

The genus is a 'measure of a curve's complexity'.

A hyperelliptic curve gives a *group* of size $\approx q^g$.

Pollard's rho method for solving the HCDLP has running time $q^{g/2}$.

Hyperelliptic Curve Cryptography

In *1989*, Koblitz proposed using groups from hyperelliptic curves in public-key cryptography.

His proposal was motivated by two factors:

1. The lack of a subexponential-time algorithm for the DLP on hyperelliptic curves (HCDLP).

   Consequence: For a 64-bit level of security, one needs $q^g \approx 2^{128}$.
   For example, $g = 4$ and $q \approx 2^{32}$, or $g = 8$ and $q \approx 2^{16}$.

2. The large number of hyperelliptic curves (approximately q^{2g-1}) for each finite field $\mathbb{F}_q$.

Number Field Sieve

(Pollard, Lenstra, Lenstra, Manasse, ...; *1990*)
Subexponential-time $2^{(\log n)^{1/3}}$ factoring algorithm.

(Gordon; *1990*)
Subexponential-time $2^{(\log p)^{1/3}}$ for the DLP in $\mathbb{F}_p^*$.

Security in bits	Block cipher	Hash function	ECC $\mathbb{F}_q$ $\|q\|_2$	RSA $\|n\|_2$
80	SKIPJACK	(SHA-1)	160	1024
112	Triple-DES	SHA-224	224	2048
128	AES Small	SHA-256	256	3072
192	AES Medium	SHA-384	384	7680
256	AES Large	SHA-512	512	15360

Early Attacks on the ECDLP/HCDLP

(M, Okamoto, Vanstone; *1990*)
Weil pairing attack: For very special elliptic curves $E/\mathbb{F}_q$, the ECDLP can be efficiently reduced to the DLP in the finite field $\mathbb{F}_{q^k}$ for some small k (where subexponential-time attacks are applicable). [k = *embedding degree*]

Example: $E : Y^2 = X^3 + 1$ over $\mathbb{Z}_p$, $p \equiv 11 \pmod{12}$ is supersingular. The embedding degree is $k = 2$.

Supersingular elliptic curves have embedding degree $k \leq 6$.

Early Attacks on the ECDLP/HCDLP

[Recall: Pollard's rho attack on the HCDLP has running time $q^{g/2}$]

(Adleman, DeMarrais, Huang; *1994*)
Index-calculus subexponential $2^{(\log q^g)^{1/2}}$ attack for very high genus hyperelliptic curves ($(2g+1)^{0.98} \geq q$).

(Araki, Satoh, Semaev, Smart; *1997*)
Smart-ASS attack: For elliptic curves $E/\mathbb{F}_p$ of size p, the ECDLP can be solved very efficiently.

Ramifications

Many cryptographers and practitioners were traumatized by these attacks on the ECDLP and HCDLP.

Their fears were somewhat alleviated by prohibiting the weak curves from emerging ECC standards:

- Example: IEEE P1363 and ANSI X9.62 required that $k \geq 20$ (thus excluding all supersingular elliptic curves).

But many experts remained skeptical about the security of both supersingular and ordinary elliptic curves.

In May *1997*, RSA Security posted *ECC Central* on their web site.

Ron Rivest (1997)

"Elliptic curves show promise as an alternative basis on which to implement public-key cryptography. They are a plausible "back-up" to RSA in case someone should discover a fast integer factorization algorithm. And in some applications their apparent ability to utilize smaller public keys might be of interest."

"But the security of cryptosystems based on elliptic curves is not well understood, due in large part to the abstruse nature of elliptic curves. Few cryptographers understand elliptic curves, so there is not the same widespread understanding and consensus concerning the security of elliptic curves that RSA enjoys...."

Ron Rivest (1997)

"...Over time, this may change, but for now trying to get an evaluation of the security of an elliptic-curve cryptosystem is a bit like trying to get an evaluation of some recently discovered Chaldean poetry. **Until elliptic curves have been further studied and evaluated, I would advise against fielding any large-scale applications based on them.**"

"As elliptic curves begin to receive the kind of stringent review already received by RSA, they will (if they are not broken) be added to RSA's toolkits and recommended for small-scale or short-term applications. In the end, time will tell how well they stand up to attack."

Len Adleman (1997)

"It is correct that I am suspicious of elliptic curve cryptosystems. ....."

"I suspect that the lack of a sub-exponential algorithm is merely a matter of neglect...."

"I am fortified in this opinion by the fact that the Jacobians of hyperelliptic curve were also suggested for cryptography.... Nonetheless Ming-Deh Huang, Jonathan DeMarrais and I were able to show that for 'high genus' hyperelliptic curves a subexponential algorithm does exist."

Arjen Lenstra (1997)

"It is true that 160-bit elliptic curve cryptosystems may offer some advantages compared to 1024-bit RSA: smaller keys, less communication, storage, and faster computation. But if I would have to make a choice today between the two, purely based on perceived security, I would opt for 1024-bit RSA. The elliptic curve discrete logarithm problem has been around for a relatively short amount of time. In my opinion only relatively few people have looked at it. Therefore, we cannot yet feel sufficiently confident, where it should be noted that even marginal progress could have very damaging consequences for the security of 160-bit elliptic curve cryptosystems. Thus, right now I think it would not be prudent to switch from 1024-bit RSA to 160-bit elliptic curve cryptosystems."

Standardized Elliptic Curves

1999-2000: The first ECC standards were issued.

Example: *ANSI X9.62*, *1999*

- ECDSA: Elliptic Curve version of the DSA.
- Elliptic curves over prime fields $\mathbb{F}_p$ and binary fields $\mathbb{F}_{2^m}$.
- Requirement on the embedding degree: $k \geq 20$.

Example: NIST's *FIPS 186-2* (ECDSA), *2000*

- 5 randomly selected elliptic curves over prime fields $\mathbb{F}_p$, where the p are Mersenne-like primes (e.g. $p = 2^{192} - 2^{64} - 1$).
- 5 randomly selected elliptic curves over binary fields $\mathbb{F}_{2^m}$, all with m prime.
- 5 Koblitz elliptic curves over binary fields $\mathbb{F}_{2^m}$, all with m prime.

XTR

(A. Lenstra & Verheul; *2000*)

- *ECSTR*: Efficient Compact Subgroup Trace Representation
- Faster than ECC and RSA.

- $n \mid p^2 - p + 1 \mid p^3 + 1 \mid p^6 - 1$, where $p \equiv 2 \pmod 3$.
- XTR group X: order-n subgroup of $\mathbb{F}_{p^6}^*$.
- $g \in X$ is represented as $\mathrm{Tr}_{p^6,p^2}(g) = g + g^{p^2} + g^{p^4} \in \mathbb{F}_{p^2}$.

- "XTR is not affected by the uncertainty still marring ECC"
- *ECSTR*: Elliptic Curves Still Too Risky

XTR and Supersingular Elliptic Curves

- Let $b \in \mathbb{F}_{p^2}$ be a square but not a cube.
- The elliptic curve $E/\mathbb{F}_{p^2} : Y^2 = X^3 + b$ is supersingular and $\#E(\mathbb{F}_{p^2}) = p^2 - p + 1$.
- Let $P \in E(\mathbb{F}_{p^2})$ be a point of order n.
- $\langle P \rangle$ has embedding degree $k = 3$, and the Weil/Tate pairings give an efficiently computable isomorphism $\phi : \langle P \rangle \longrightarrow X \subset \mathbb{F}_{p^6}^*$.

- In 2000, we asked whether there is an efficiently computable isomorphism $\psi : X \longrightarrow \langle P \rangle$.
- *ECSTR*: Elliptic Curve Supersingular Trace Representation

Verheul's Theorem

In *May 2001*, Verheul proved the following:
Suppose that there is an efficiently-computable isomorphism ψ from the XTR group X to the order-n subgroup $\langle P \rangle$ of $E(\mathbb{F}_{p^2})$. Then the Diffie-Hellman problems in X and $\langle P \rangle$ are efficiently solvable.

He concludes that his results:

> ...provide evidence that the multiplicative group of a finite field provides essentially more, and in any case not less, security than the group of points of a supersingular elliptic curve of comparable size.

The Resurrection

Invention of *pairing-based cryptography*:
Joux & Sakai-Ohgishi-Kasahara (2000)
Boneh-Franklin (*August 2001*)

Permitted functionality not achievable by RSA/ECC.

- Killer application: Identity-based encryption.

Main tool: Weil/Tate pairings on elliptic curves with small embedding degree!!

Supersingular elliptic curves were resurrected from the dead!!

New Assumptions

Bilinear pairing: $e : \mathbb{G} \times \mathbb{G} \longrightarrow \mathbb{G}_T$

Bilinear Diffie-Hellman Problem: Given group element g, g^x, g^y, g^z, compute $e(g,g)^{xyz}$.

Strong Diffie-Hellman Problem: Given the $m+1$ group elements g, g^x, $g^{x^2}, \ldots, g^{x^m}$, find a pair (c,h) (where c is a nonzero integer mod n and h is a group element) such that $h^{x+c} = g$.

Decision Linear Problem: Given group elements g, g^x, g^y, g^{ax}, g^{by} and Z, decide whether $Z = g^{a+b}$.

Acceptance of Pairing-Based Cryptography

It is surprising that despite the prevailing mistrust of ECC in general, and supersingular elliptic curves in particular, pairing-based cryptography (PBC) was immediately accepted by the research community.

There are four reasons for this:

1. The timing was right – the bitter RSA-ECC rivalry had subsided.
2. PBC was not viewed by commercial organizations as disruptive to their interests.
3. PBC was not viewed by academic researchers as disruptive to their interests.
4. PBC protocols were presented with elaborate security proofs.

NSA: The Case for Elliptic Curve Cryptography

NSA: National Security Agency

`www.nsa.gov/ia/industry/crypto_elliptic_curve.cfm`

(*2005*) “The best assured group of new public key techniques is built on the arithmetic of elliptic curves. This paper will outline a case for moving to elliptic curves as a foundation for future Internet security. This case will be based on both the relative security offered by elliptic curves... and the relative performance of these algorithms. While at current security levels elliptic curves do not offer significant benefits over existing public key algorithms, as one scales security upwards over time to meet the evolving threat posed by eavesdroppers and hackers with access to greater computing resources, elliptic curves begin to offer dramatic savings over the old, first generation techniques.”

NSA Suite B, 2005

For the protection of both classified and unclassified National Security Information.

Also for the protection of classified government communications with UK, Canada, and certain other NATO countries.

The only public-key scheme is ECC.
One randomly selected elliptic curve over a 256-bit prime field.
One randomly selected elliptic curve over a 384-bit prime field.

About 1.3 million units of equipment will be replaced over the next 10 years.

Equipment will be used for the next 20-50 years.

NIST Key Lifetimes

- FIPS 201: Personal Identity Verification (PIV) of Federal Employees and Contractor (with smart cards)
- SP 800-78: Cryptographic Algorithms and Key Sizes for PIV.
- Elliptic curves: P-256, P-384.

PIV authen. key	– 2013	RSA 1024/2048, ECDSA P-256
	2014–	RSA 2048, ECDSA P-256
Card authen. key	– 2013	RSA 1024/2048, ECDSA P-256
	2013 –	RSA 2048, ECDSA P-256
Digital signature key	– 2008	RSA 1024/2048, ECDSA P-256/P-384
	2009 –	RSA 2048, ECDSA P-256/P-384
Key management key	– 2008	RSA 1024/2048, ECDSA P-256/P-384
	2009 –	RSA 2048, ECDSA P-256/P-384

ECC Deployments

Canada Post Digital Postal Marks

Research in Motion's BlackBerry

Microsoft DRM 2.0 (used in Windows media player)

German e-Passports

Pairing-Based Cryptography

IETF RFC 5091 (December *2007*):
Identity-Based Cryptography Standard (IBCS) #1
The only elliptic curves permitted are:
$E : Y^2 = X^3 + 1$ over $\mathbb{Z}_p$, $p \equiv 11 \pmod{12}$

Luther Martin, Voltage (*2008*): "The conservative choice for implementing a pairing-based algorithm is to use a supersingular curve."

Recent Developments: ECDLP

(Frey; *1999*) Weil descent methodology for reducing the ECDLP for an elliptic curve over $\mathbb{F}_{2^{mk}}$ to the HCDLP for a genus-g hyperelliptic curve over $\mathbb{F}_{2^m}$.

(Gaudry, Hess, Smart; *2000*) Realized Frey's methodology in their *GHS Weil descent attack*.

(M, Qu; *2001*) Showed that the GHS attack fails for *all* elliptic curves over $\mathbb{F}_{2^m}$ where $m \in [160, 600]$ is prime.

(Jacobson, M, Stein; *2001*) Showed that the GHS attack is effective for *some* elliptic curves over $\mathbb{F}_{2^m}$ for some *composite* m (e.g., $m = 124$ and 155).

(M, Teske; *2005*) Showed that fields $\mathbb{F}_{2^m}$ with m divisible by $3, 5, 6, 7, 8$ are *weak* (or potentially weak) for ECC.

Recent Developments: ECDLP (2)

(Gaudry; *2004*) Fully exponential $q^{2-2/n}$ attack on ECDLP over extension fields $\mathbb{F}_{q^n}$ for small n (faster than Pollard $q^{n/2}$ for small $n \geq 3$).

(Diem; *2004*) Subexponential $2^{(\log q^n)^{3/4}}$ attack on ECDLP over extension fields $\mathbb{F}_{q^n}$ for $n \approx \log q$.

Recent Developments: HCDLP

[Recall that Pollard's rho attack has running time $q^{g/2}$]

(Gaudry; *2000*) Fully exponential q^2 attack for small genus (faster than Pollard for $g \geq 5$).

(Thériault; *2004*) Fully exponential $q^{2-2/(g+1)}$ attack for small genus (faster than Pollard for $g \geq 4$).

(Gaudry, Thériault, Thomé, Diem; *2007*) Fully exponential $q^{2-2/g}$ attack for small genus (faster than Pollard for $g \geq 3$).

(Smith; *2008*) Fully exponential q attack for about 18% of all genus 3 hyperelliptic curves.

Current Research in Efficient Implementation

Efficient implementation of arithmetic in $\mathbb{F}_{2^m}$ (Aranha, López, ...)

Intel's announcement of carry-less multiplication instruction, PCLMULQDQ

Edwards coordinates (Bernstein and Lange)

Galbraith-Lin-Scott elliptic curves over $\mathbb{F}_{q^2}$.

Pairings (Barreto, Dahab, Devegili, ...)

Genus-2 hyperelliptic curves (Gaudry, Bernstein)

Genus-3 and 4 hyperelliptic curves (Avanzi & Thériault)

Concluding Remarks

The ECDLP for elliptic curves over prime fields $\mathbb{F}_p$ and binary fields $\mathbb{F}_{2^m}$ (with m prime) has withstood the test of time.

But, of course, the future is impossible to predict.....

Deciding whether to use RSA, ECC, genus-2 hyperelliptic curves, pairing-based cryptography, or a post-quantum public-key system, does require *speculation* about future developments and a lot of *faith*.