Attractive Subfamilies of BLS Curves for Implementing High-Security Pairings

Craig Costello

craig.costello@qut.edu.au
Queensland University of Technology

IndoCrypt 2011
Chennai, India

Joint work with
Kristin Lauter (Microsoft) and Michael Naehrig (Eindhoven)

Balanced security in PBC

- Pairing-based crypto is different to other number-theoretic crypto settings: three groups!

$$\mathbb{G}_1 \times \mathbb{G}_2 \longrightarrow \mathbb{G}_T$$

- $\mathbb{G}_1 = E(\mathbb{F}_q)[r]$ and $\mathbb{G}_2 \subset E(\mathbb{F}_{q^k})[r]$ are elliptic curve groups
- $\mathbb{G}_T = \mu_r \subset \mathbb{F}_{q^k}$ is a subgroup of a finite (extension) field
- $\mathbb{G}_1$ and $\mathbb{G}_2$ must resist **exponential** attacks
- $\mathbb{G}_T$ must resist **subexponential** attacks
- How do we optimally balance this resistance?
- The embedding degree k does exactly this

The embedding degree k

$\mathbb{G}_1$ and $\mathbb{G}_2$ $\mathbb{G}_T$

Security level (in bits)	Subgroup size r (in bits)	Extension field size q^k (in bits)	Embedding degree k	
			$\rho \approx 1$	$\rho \approx 2$
80	160	960 – 1280	6 – 8	2*,3 – 4
112	224	2200 – 3600	10 – 16	5 – 8
128	256	3000 – 5000	12 – 20	6 – 10
192	384	8000 – 10000	20 – 26	10 – 13
256	512	14000 – 18000	28 – 36	14 – 18

- 80-bit security
- $k = 6$, $\rho = 1$ MNT curve: $E/\mathbb{F}_q : y^2 = x^3 - 3x + b$

q = 801819385093403524905014779542892948310645897957
(160 bits)
r = 801819385093403524905015674986573529844218487823
(160 bits)
$\mathbb{F}_{q^6} \approx 960$ bits

The embedding degree k

$\mathbb{G}_1$ and $\mathbb{G}_2$ $\mathbb{G}_T$

Security level (in bits)	Subgroup size r (in bits)	Extension field size q^k (in bits)	Embedding degree k $\rho \approx 1$	$\rho \approx 2$
80	160	960 – 1280	6 – 8	2*,3 – 4
112	224	2200 – 3600	10 – 16	5 – 8
128	256	3000 – 5000	12 – 20	6 – 10
192	384	8000 – 10000	20 – 26	10 – 13
256	512	14000 – 18000	28 – 36	14 – 18

- 128-bit security
- $k = 12$, $\rho = 1$ BN curve: $E/\mathbb{F}_q : y^2 = x^3 + b$

  q = 115792089237314936872688561244471742058375878
  35576120519870040952262966451816 3 (256 bits)

  r = 115792089237314936872688561244471742058035595 9
  888402685844887579994295356170 37 (256 bits)

  $\mathbb{F}_{q^{12}} \approx 3072$ bits

The embedding degree k

$\mathbb{G}_1$ and $\mathbb{G}_2$ $\mathbb{G}_T$

Security level (in bits)	Subgroup size r (in bits)	Extension field size q^k (in bits)	Embedding degree k	
			$\rho \approx 1$	$\rho \approx 2$
80	160	960 – 1280	6 – 8	2*,3 – 4
112	224	2200 – 3600	10 – 16	5 – 8
128	256	3000 – 5000	12 – 20	6 – 10
192	384	8000 – 10000	20 – 26	10 – 13
256	512	14000 – 18000	28 – 36	14 – 18

- 192-bit security
- $k = 18$, $\rho = 1.33$ KSS curve: $E/\mathbb{F}_q : y^2 = x^3 + b$

q = 1439371658719548007677605460638469914138672023932108640095444258664551345484186154160442181069966053963055565407692343301090652336074915081562182907540863517 (519 bits)

r = 37583745740549219845280578393415895486585013666199128505131657943724238216654126921038087699129845495981755041054721 (384 bits)

$\mathbb{F}_{q^{18}} \approx 9192$ bits

Pairing-friendly curves are rare!

- **Balasubramanian and Koblitz**: $\mathbb{G}_1$ and $\mathbb{G}_2$ defined over $\mathbb{F}_{q^k}$ $\left(E[r] \subset E(\mathbb{F}_{q^k})\right)$ if and only if $r \mid q^k - 1$
- k is smallest i with $r \mid q^i - 1$
- Consequence: $k \approx r$ (huge!) in general
- k needs to be small enough $(k < 50)$ so that we can work in $\mathbb{F}_{q^k}$
- Consequence: pairing-friendly curves are very rare, and sometimes very hard to find

BLS curves

- 2002: **Barreto, Lynn and Scott** (BLS) described several constructions for families of pairing friendly curves

- One of which (for $k = 24$) remains a stand-out candidate for high-security (256-bit) pairings

BLS curves for $k = 24$

- A nice choice for 256-bit secure pairings

$$q(x) = (x-1)^2(x^8 - x^4 + 1)/3 + x$$
$$n(x) = (x-1)^2(x^8 - x^4 + 1)/3$$
$$r(x) = x^8 - x^4 + 1$$
$$t(x) = x + 1$$

- Find any $x \equiv 1 \bmod 3$ with q prime and r (almost) prime, and you have a pairing-friendly BLS curve with $k = 24$
- **Curve always of the form** $y^2 = x^3 + b$

BLS curves for $k = 24$: a baby example

$$q(x) = (x-1)^2(x^8 - x^4 + 1)/3 + x$$
$$n(x) = (x-1)^2(x^8 - x^4 + 1)/3$$
$$r(x) = x^8 - x^4 + 1$$
$$t(x) = x + 1$$

$$x = x_0 = 10$$
$$q = 2699730037 \qquad (32 bits)$$
$$r = 99990001 \qquad (27 bits)$$
$$k = 24 \qquad r \mid p^{24} - 1$$

BLS curves for $k = 24$: a real-world example

$$q(x) = (x-1)^2(x^8 - x^4 + 1)/3 + x$$
$$n(x) = (x-1)^2(x^8 - x^4 + 1)/3$$
$$r(x) = x^8 - x^4 + 1$$
$$t(x) = x + 1$$

$$\begin{aligned}
x = x_0 &= 18338657682652688728 \qquad (64 bits) \\
q &= 1434016616962548944783218664270924317907608905231220493360 \\
&\quad 1327661303199716098754375973960160894842258771468709483957 6 \\
&\quad 6001176835975792058849921228650147683237429431766511865973945 \\
&\quad 755928704738611 \qquad (640 bits) \\
r &= 1279205596716260280573968849354620177704023806848485273906 35 \\
&\quad 9353979893651298023411038699453704764585363166316776814890786 2 \\
&\quad 69457457452526276055453990524928 1 \qquad (512 bits) \\
k &= 24 \qquad r \mid p^{24} - 1 \\
\rho &= 1.25 \qquad (\log p / \log r = 1.25)
\end{aligned}$$

Guaranteed (high-level) properties of $k = 24$ BLS curves

- **Best ρ value for $k = 24$:** $\rho = 1.25$
- **Snug fit for 256-bit security:** $q = 640$ bits gives $r = 512$ and $\mathbb{F}_{p^{24}} = 15360$ bits - perfect for 256-bit security
- **Highest degree twist ($d = 6$) applicable:** points in $\mathbb{G}_2 \subset E(\mathbb{F}_{q^{24}})[r]$ are isomorphic to points on twist $\mathbb{G}_2' = E'(\mathbb{F}_{q^4})[r]$
- **ate pairing is optimal:** pairing loop length lower bound $r/\phi(k)$ is achieved with ate pairing (simple)
- **nice final exponentiation:** addition chain trivial
- **... but some family members are more attractive (implementation-friendly) than others**

Not-always-guaranteed properties of $k = 24$ BLS curves

- What about representing the field $\mathbb{F}_{q^{24}}$? Can we guarantee a highly-efficient construction?
- What about the curve $E/\mathbb{F}_q : y^2 = x^3 + b$? Do we have to test for the correct b? Is it always small?
- What about the twisted curve $E/\mathbb{F}_{q^4} : y^2 = x^3 + b'$? Do we have to test (count points) for the correct b'? Are the twisting/untwisting isomorphisms nice?
- Can we achieve a low hamming-weight (NAF) value of $x = x_0$?
- **If we search with $x \equiv 1 \bmod 3$, we can't always guarantee all of the above for each curve found!**
- **This work: determines subfamilies of BLS curves that (provably) guarantee the above properties**

Splitting up the BLS family

- Instead of searching with $x \equiv 1 \bmod 3$, search with any of $x \equiv 7, 16, 31, 64 \bmod 72$, and all of the previous properties are guaranteed

- For the other 20 congruency classes $x \not\equiv 7, 16, 31, 64 \bmod 72$, we argue that all of the above properties can't be satisfied simultaneously

x_0 (mod 72)	$q(x_0)$ (mod 72)	$n(x_0)$ (mod 72)	efficient tower Prop. 2	E Prop. 3	E' Prop. 4
7	19	12	✓	$y^2 = x^3 + 1$	$y^2 = x^3 \pm 1/v$
16	19	3	✓	$y^2 = x^3 + 4$	$y^2 = x^3 \pm 4v$
31	43	12	✓	$y^2 = x^3 + 1$	$y^2 = x^3 \pm v$
64	19	27	✓	$y^2 = x^3 - 2$	$y^2 = x^3 \pm 2/v$

- **A large bulk of the paper is dedicated to proving the above claims.**

Highly efficient towering options

- 2005: For $k = 2^i 3^j$, **Koblitz-Menezes** suggest using irreducible binomials to represent $\mathbb{F}_{q^k}$ as a tower of quadratic/cubic extensions from $\mathbb{F}_q$

- 2010: **Benger-Scott** further generalize and give useful theorems for testing if $\mathbb{F}_{q^k}$ is *towering-friendly*

- Nice towers facilitate efficient $\mathbb{F}_{q^k}$ arithmetic, but nicest options not always available... but in our four cases....

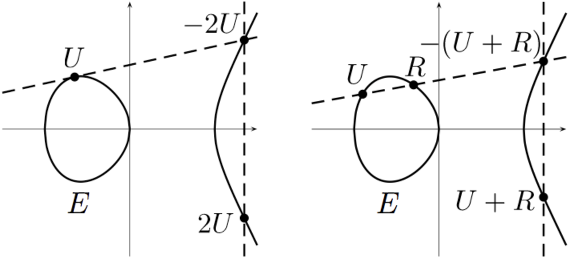
Highly efficient towering options

Proposition 2. *Let $x_0 \in \mathbb{Z}$ be any of $x_0 \equiv 7, 16, 31, 64 \pmod{72}$. If $p = p(x_0)$ given by the polynomial in (1) is prime, then the extension field $\mathbb{F}_{p^{24}}$ can be constructed using any of the following towering options T_1, T_2, T_3:*

$$
\begin{tikzcd}
 & & & \mathbb{F}_{p^8} \arrow[r, "z^3+w"] & \mathbb{F}_{p^{24}}; \quad T_1, \\
\mathbb{F}_p \arrow[r, "u^2+1"] & \mathbb{F}_{p^2} \arrow[r, "v^2+u+1"] & \mathbb{F}_{p^4} \arrow[ur, "w^2+v"] \arrow[rr, "z^6+v"] \arrow[dr, "w^3+v"'] & & \mathbb{F}_{p^{24}}; \quad T_2, \\
 & & & \mathbb{F}_{p^{12}} \arrow[r, "z^2+w"'] & \mathbb{F}_{p^{24}}; \quad T_3.
\end{tikzcd}
$$

- Tricks in cubic and quadratic extension fields facilitate much faster multiplications (squarings) than the naive schoolbook method

Miller's algorithm for ate pairing $f_Q(P)^{(q^k-1)/r}$



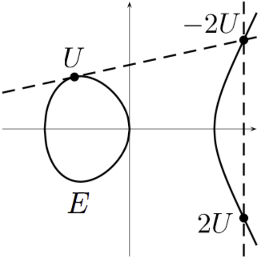
$x_0' = (x_{l-1}, \ldots, x_1, x_0)_2$
initialize: $U = Q$, $f = 1$
for $i = l-2$ to 0 do

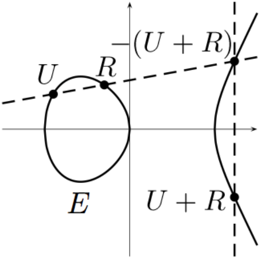
a. i. Compute $f_{\mathrm{DBL}(U)}$ in the doubling of U
   ii. $U \leftarrow [2]U$ //(DBL)
   iii. $f \leftarrow f^2 \cdot f_{\mathrm{DBL}(U)}(P)$

b. if $x_i = 1$ then
   i. Compute $f_{\mathrm{ADD}(U,Q)}$ in the addition of $U + Q$
   ii. $U \leftarrow U + Q$ //(ADD)
   iii. $f \leftarrow f \cdot f_{\mathrm{ADD}(U,Q)}(P)$

c. Exponentiation f to power $(q^k - 1)/r$

Miller's algorithm for ate pairing $f_Q(P)^{(q^k-1)/r}$





$x_0' = (x_{l-1}, \ldots, x_1, x_0)_2$
initialize: $U = Q$, $f = 1$
for $i = l-2$ to 0 do

a. i. Compute $f_{\mathrm{DBL}(U)}$ in the doubling of U
   ii. $U \leftarrow [2]U$ //(DBL)
   iii. $f \leftarrow f^2 \cdot f_{\mathrm{DBL}(U)}(P)$

b. if $x_i = 1$ then
   i. Compute $f_{\mathrm{ADD}(U,Q)}$ in the addition of $U + Q$
   ii. $U \leftarrow U + Q$ //(ADD)
   iii. $f \leftarrow f \cdot f_{\mathrm{ADD}(U,Q)}(P)$

c. Exponentiation f to power $(q^k - 1)/r$

Fast operations and to twist or to untwist?

- **2004- Chatterjee, Sarkar and Barua:** optimize point operations and line computations simultaneously (*encapsulated* doubling/addition in Miller's algorithm)

- C-Lange-Naehrig PKC2010: optimized formulas in all practical contexts and observation that everything can be done on the twisted curve

$$f_{T,\psi(Q')}(P)^{(q^{24}-1)/r} \qquad vs. \qquad f_{T,Q'}(P')^{(q^{24}-1)/r}$$

- For $k = 24$ BLS, twisting isomorphism ψ^{-1} can be much nicer than untwisting isomorphism ψ (see §4 of the paper)

Recipe: How to use this paper

x_0 (mod 72)	$q(x_0)$ (mod 72)	$n(x_0)$ (mod 72)	efficient tower Prop. 2	E Prop. 3	E' Prop. 4
7	19	12	✓	$y^2 = x^3 + 1$	$y^2 = x^3 \pm 1/v$
16	19	3	✓	$y^2 = x^3 + 4$	$y^2 = x^3 \pm 4v$
31	43	12	✓	$y^2 = x^3 + 1$	$y^2 = x^3 \pm v$
64	19	27	✓	$y^2 = x^3 - 2$	$y^2 = x^3 \pm 2/v$

- Search for BLS curves with any of $x_0 \equiv 7, 16, 31, 64 \bmod 72$ instead of $x_0 \equiv 1 \bmod 3$
  - i **Primality test $p(x_0)$ and $r(x_0)$ only!**
  - ii Compact: all parameters deteremined entirely by x_0
  - iii No point counting or further testing
  - iv Highly efficient tower guaranteed
  - v Nice twist or untwist isomorphism guaranteed
- **OR** use one that we prepared earlier...

security level	$x_0 \equiv 16 \pmod{72}$	weight	p (bits)	words for p	r (bits)	words for r	security (bits)
224	$2^{56} - 2^{53} - 2^{31} - 2^{9}$	4	557	9×64	447	7×64	223
	$-2^{56} + 2^{40} - 2^{26} - 2^{6}$	4	559		448		224
	$2^{56} + 2^{40} - 2^{20}$	3	559		449	15×32	224
	$2^{57} + 2^{25} + 2^{18} + 2^{11}$	4	569		457		228
	$2^{57} + 2^{54} + 2^{51} + 2^{39}$	4	571		458		229

Table: an example chunk from one of our tables

Recipe: How to use this paper (cont.)

$x_0 \pmod{72}$	$q(x_0) \pmod{72}$	$n(x_0) \pmod{72}$	efficient tower Prop. 2	E Prop. 3	E' Prop. 4
7	19	12	✓	$y^2 = x^3 + 1$	$y^2 = x^3 \pm 1/v$
16	19	3	✓	$y^2 = x^3 + 4$	$y^2 = x^3 \pm 4v$
31	43	12	✓	$y^2 = x^3 + 1$	$y^2 = x^3 \pm v$
64	19	27	✓	$y^2 = x^3 - 2$	$y^2 = x^3 \pm 2/v$

- Elliptic curve E and (correct) twisted curve E' are automatically defined
- Use the tower in Proposition 2
- Use encapsulated doubling/addition formulas from C-Lange-Naehrig PKC2010 (see also Aranha *et al.* Eurocrypt 2011)
- Refer to Table 2 to see whether to twist or untwist
- Use final exponentiation routine in Table 3
- **Enjoy highly efficient, implementation-friendly, high-security pairings**

Further benefits...

- **Pereira, Simplício, Naehrig and Barreto**: recently found attractive subfamilies of k=12 BN curves (128-bit security)
- Pereira *et al.*: *"Avoids expensive tests during curve generation"*
- Pereira *et al.*: *"Certain attacks can be prevented by checking that the purported curve contained in a given digital certificate does indeed exhibit the expected properties before using that certificate"*
- Pereira *et al.*: *"e.g. a lightweight certificate server would only need plain integer arithmetic up to primality checking (and no elliptic curve arithmetic support) to attest the well-formedness of the curves"*

Related (upcomming) work...

- BN and BLS curves now have implementation-friendly subfamilies
- What about all the other families (KSS, BLS $k \neq 24$, Brezing-Weng, MNT...) - see Freeman-Scott-Teske "*A taxonomy of pairing-friendly elliptic curves*"
- Perhaps "*a taxonomy of implementation-friendly subfamilies*"... maybe even in time for submission to Pairing2012?

THANKS!