On pairing implementation

Google Private Computing Tech Talk

Youssef El Housni — 07/24/2025

whoami

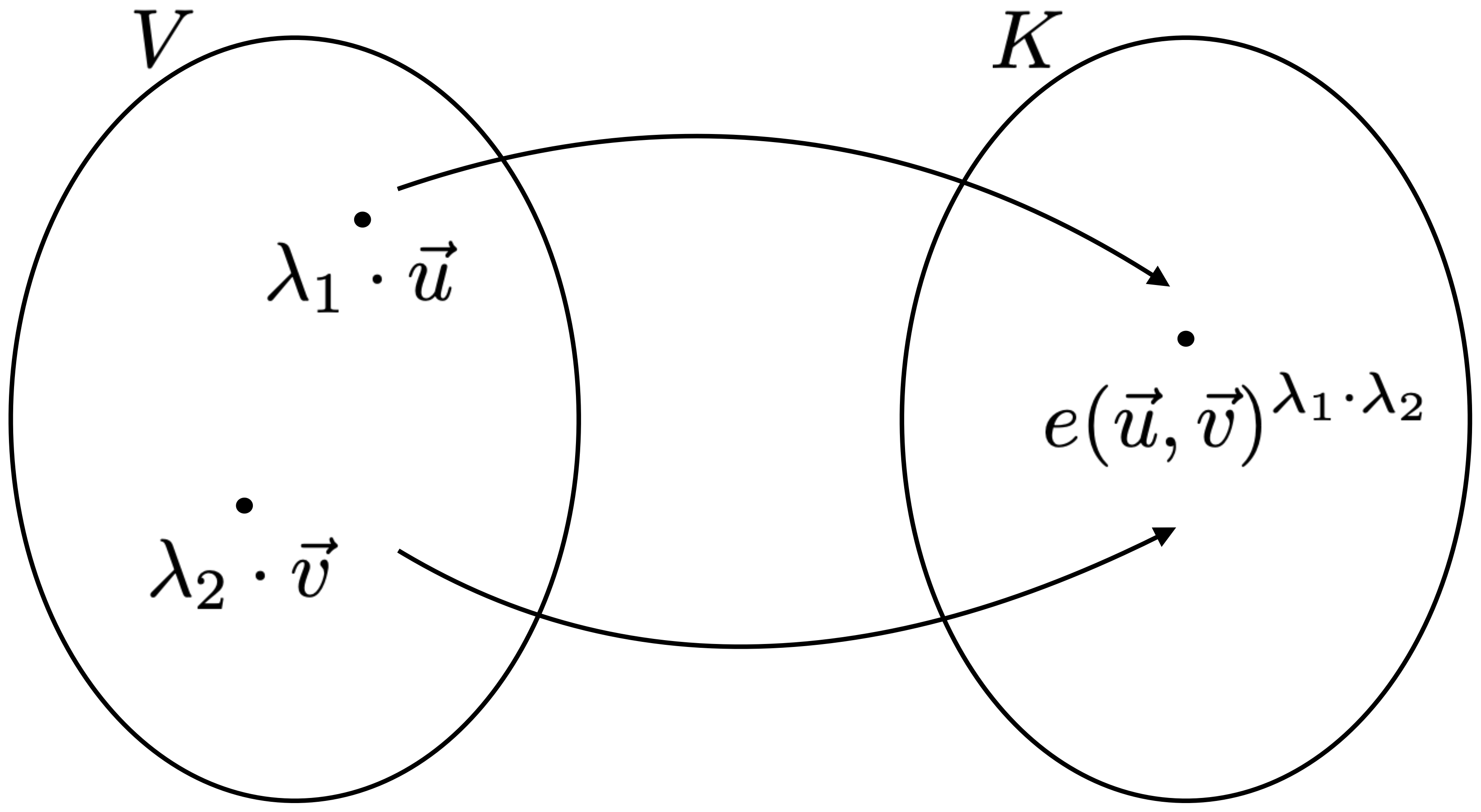
- PhD in cryptography — Ecole Polytechnique (Paris)
- Cryptographer — Consensys (NYC)
- Co-developer of gnark
- Co-developer of Linea

Outline

- Motivation
- History
- Background
- Implementation

Motivation

Bi-linear pairing



- Vector space V and field K $\quad e : V \times V \to K$
- Bi-linearity:

$$e(\vec{u} + \vec{v}, \vec{w}) = e(\vec{u}, \vec{w}) \ \cdot \ e(\vec{v}, \vec{w}) \text{ and } e(\lambda \cdot \vec{u}, \vec{v}) = e(\vec{u}, \vec{v})^{\lambda}$$

$$e(\vec{u}, \vec{v} + \vec{w}) = e(\vec{u}, \vec{v}) \ \cdot \ e(\vec{u}, \vec{w}) \text{ and } e(\vec{u}, \lambda \cdot \vec{v}) = e(\vec{u}, \vec{v})^{\lambda}$$

Motivation

Cryptographic bi-linear pairing

$(\mathbf{G}_1, +), (\mathbf{G}_2, +), (\mathbf{G}_T, \cdot)$ three cyclic groups of large prime order ℓ

Pairing: map $e : \mathbf{G}_1 \times \mathbf{G}_2 \to \mathbf{G}_T$

1. bilinear: $e(P_1 + P_2, Q) = e(P_1, Q) \cdot e(P_2, Q)$, $e(P, Q_1 + Q_2) = e(P, Q_1) \cdot e(P, Q_2)$
2. non-degenerate: $e(G_1, G_2) \neq 1$ for $\langle G_1 \rangle = \mathbf{G}_1$, $\langle G_2 \rangle = \mathbf{G}_2$
3. efficiently computable.

Most often used in practice:

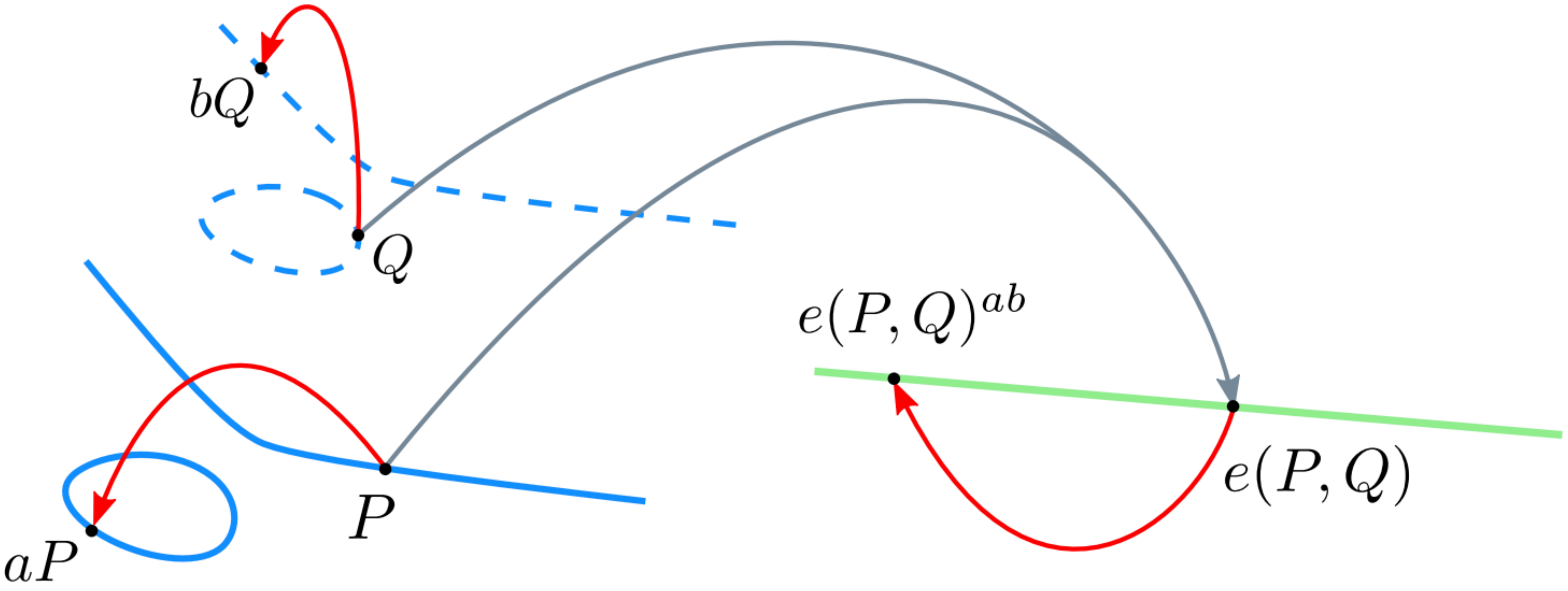
$$e([a]P, [b]Q) = e([b]P, [a]Q) = e(P, Q)^{ab} .$$

Motivation

Cryptographic bi-linear pairing

$$e([a]P, [b]Q) = e([b]P, [a]Q)$$

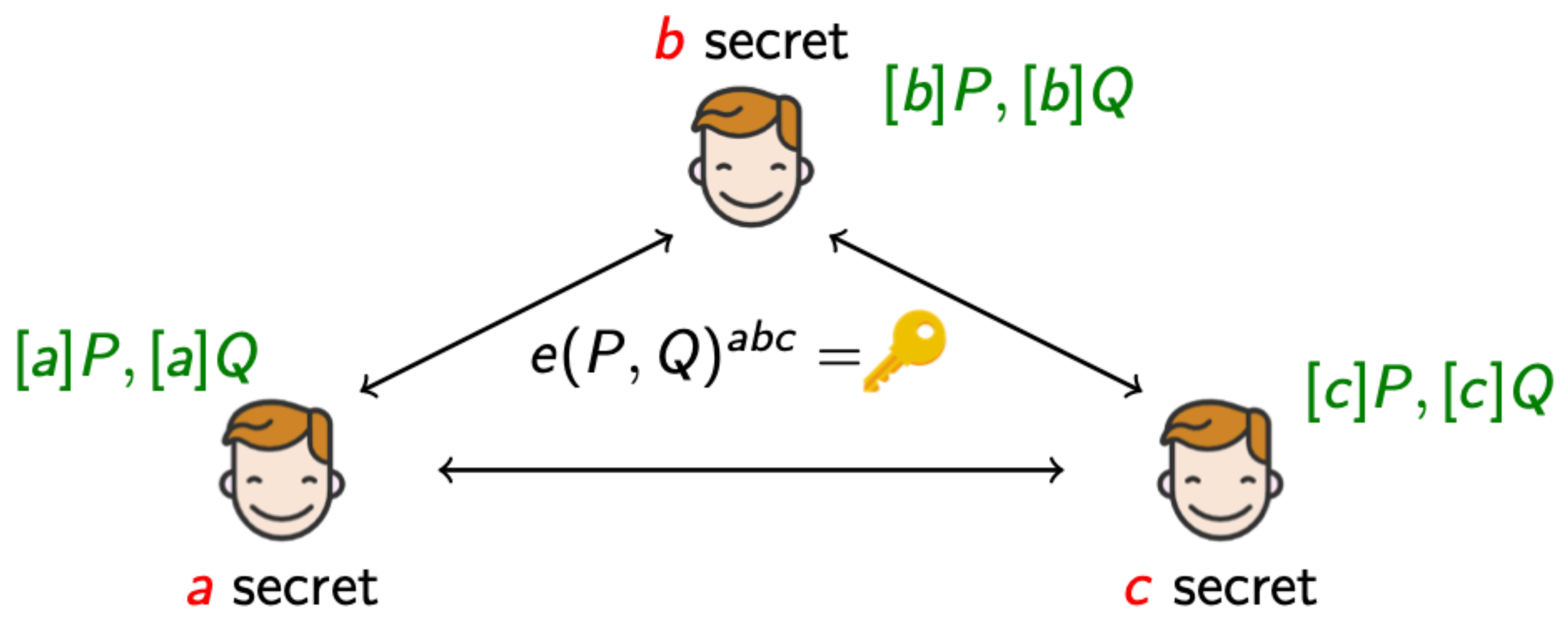
$$\{[a_1]P, \ldots, [a_n]P\}, \{[b_1]Q, \ldots, [b_n]Q\} \implies e(P,Q)^{\vec{a}\cdot\vec{b}}$$



Motivation

Applications in cryptography

Application 1. Tripartite one round key exchange. (Joux 2000)



$$e([b]P, [c]Q)^{a} = e(P, Q)^{bca}$$

Application 2. BLS signature

$H : \{0,1\}^* \to \langle P \rangle$ is a hash function, with $P \in E(\mathbb{F}_p)$ of prime order r.

Secret key: $s_k \in \{2, \ldots, r-1\}$.

Public key: $P_k = [s_k]P$.

Signing a message $M \in \{0,1\}^*$: $\sigma = [s_k]H(M)$.

Verifying the signature: $e(P_k, H(M)) \stackrel{?}{=} e(P, \sigma)$.

$$e(P_k, H(M)) = e([s_k]P, H(M)) = e(P, [s_k]H(M)) = e(P, \sigma)$$

History

- **1984 (Lenstra Jr):** Elliptic Curve Method (ECM) for integer factoring
- **1985 (Koblitz, Miller):** proposed elliptic curves for cryptography

…

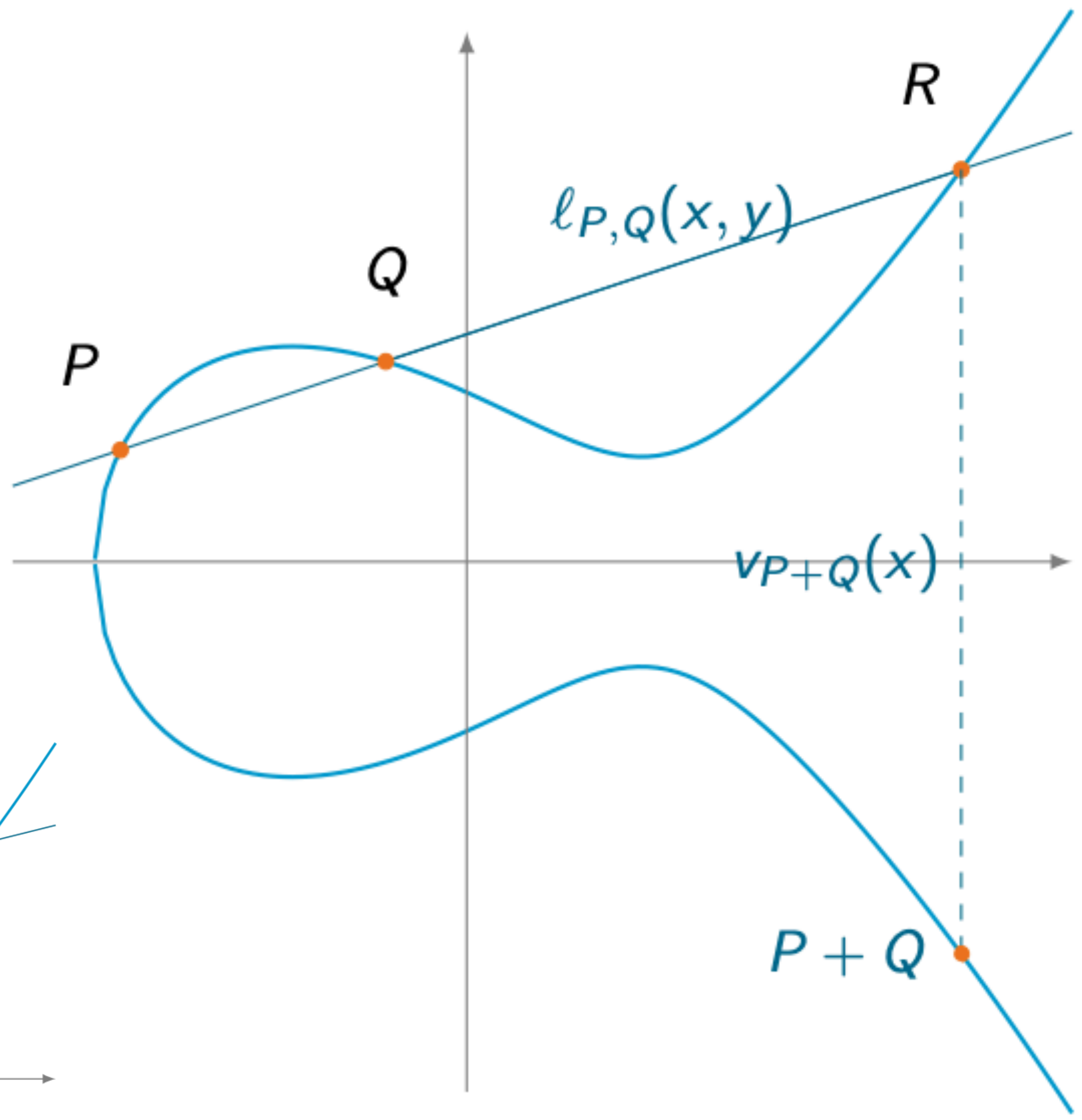
- **1993 (Menezes, Okamoto, Vanstone):** Attack on DLP using pairings
- **1994 (Frey, Rück):** Attack on DLP using pairings (optimisation)

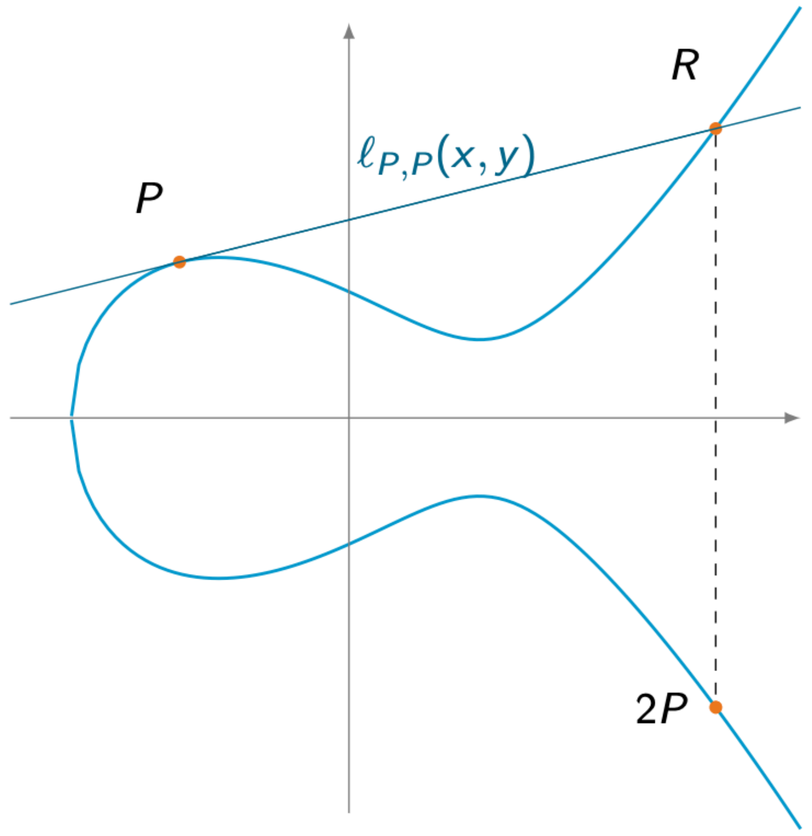
$$[s]P \to e([s]P, Q) = e(P, Q)^s$$

- **2000 (Joux):** One round tripartite Diffie-Hellman protocol
- **2001 (Boneh, Franklin):** identity-based encryption protocol
- **2004 (Boneh, Lynn, Shacham):** short signature protocol

Background
Elliptic curves

Elliptic curve $E/\mathbb{F}_p$: $y^2 = x^3 + ax + b$, $a, b \in \mathbb{F}_p$, $p \geq 5$, group law





- $E(\mathbb{F}_p)$ has an efficient group law $\to$ $\mathbf{G}_1$ (chord and tangent rule)
- $\#E(\mathbb{F}_p) = p + 1 - t$, trace t: $|t| \leq 2\sqrt{p}$
- large prime $q \mid p + 1 - t$ coprime to p
- $E(\mathbb{F}_p)[q] = \{P \in E(\mathbb{F}_p) \colon [q]P = \mathcal{O}\}$ has order q
- $E[q] \simeq \mathbf{Z}/q\mathbf{Z} \times \mathbf{Z}/q\mathbf{Z}$ (for crypto)
- only generic attacks against DLP on well-chosen genus 1 and genus 2 curves
- optimal parameter sizes

Background

Scalar multiplication

With an addition law on E, the points on the curve form a group $E(K)$.

Scalar multiplication (exponentiation)

The **multiplication-by-m** map, or **scalar multiplication** is

$$\begin{aligned}[m]\colon E &\to E \\ P &\mapsto \underbrace{P + \ldots + P}_{m \text{ copies of } P}\end{aligned}$$

for any $m \in \mathbb{Z}$, with $[-m]P = [m](-P)$ and $[0]P = \mathcal{O}$.

- a key-ingredient operation in public-key cryptography
- given $m > 0$, computing $[m]P$ as $P + P + \ldots P$ with $m - 1$ additions is **exponential** in the size of m: $m = e^{\ln m}$
- we can compute $[m]P$ in $O(\log m)$ operations on E.

$\to$ complexity of inverting exponentiation in $O(\sqrt{\#G})$

$\to$ security level 128 bits means $\sqrt{\#G} \geq 2^{128}$

Tate pairing

$\ell \mid p^n - 1,\ E[\ell] \subset E(\mathbb{F}_{p^n})$
Tate Pairing: $e: \ E(\mathbb{F}_{p^n})[\ell] \times E(\mathbb{F}_{p^n})/\ell E(\mathbb{F}_{p^n}) \to \mathbb{F}_{p^n}^*/(\mathbb{F}_{p^n}^*)^\ell$ (equivalence classes)
For cryptography,

- $\mathbf{G}_1 = E(\mathbb{F}_p)[\ell] = \{P \in E(\mathbb{F}_p), [\ell]P = \mathcal{O}\}$
- embedding degree $n > 1$ w.r.t. ℓ: smallest[1] integer n s.t. $\ell \mid p^n - 1$
  $\Leftrightarrow E[\ell] \subset E(\mathbb{F}_{p^n})$
- $\mathbf{G}_2 \subset E(\mathbb{F}_{p^n})[\ell]$
- $\mathbf{G}_1 \cap \mathbf{G}_2 = \mathcal{O}$ by construction for practical applications
- $\mathbf{G}_T = \boldsymbol{\mu}_\ell = \{u \in \mathbb{F}_{p^n}^*,\ u^\ell = 1\} \subset \mathbb{F}_{p^n}^*$

When n is small i.e. $1 \leqslant n \leqslant\sim 50$, the curve is *pairing-friendly*.
This is very rare: For a given curve, $\log n \sim \log \ell$ (Balasubramanian–Koblitz).

Tate pairing
Divisors

Let E be an elliptic curve defined over a field K.

Definitions

A **divisor** is a *finite* formal sum of points $P_i \in E(K)$, $D = \sum_{i=1}^{n} a_i(P_i)$, $a_i \neq 0 \in \mathbb{Z}^{\times}$
Degree: $\deg(D) = \sum_{i=1}^{n} a_i \in \mathbb{Z}$: sum of the weights a_i, can be 0
Sum: $\mathrm{sum}(D) = a_1 P_1 + \ldots + a_n P_n$ the sum on E of the weigted points $a_i P_i$.
Support: $\mathrm{supp}(D) = \{P_i\}_{1 \leq i \leq n}$ the set of points of D of weight $a_i \neq 0$.

Evaluating a function at a divisor

Let $D = \sum_{i=1}^{n} a_i(P_i) - (\sum_{j=1}^{m} a_j(P_j))$ where $a_i, a_j > 0$.

$$f(D) = \frac{\prod_{i=1}^{n}(f(P_i))^{a_i}}{\prod_{j=1}^{m}(f(P_j))^{a_j}}$$

Tate pairing

Let E be an elliptic curve over a finite field $\mathbb{F}_q$, let $\ell \mid \#E(\mathbb{F}_q)$, $\gcd(\ell, q) = 1$

pre-$\mathbf{G}_1$, $\mathbf{G}_2$ and $\mathbf{G}_T$

With $\ell E(\mathbb{F}_q) = \{[\ell]Q \colon Q \in E(\mathbb{F}_q)\}$

- $E(\mathbb{F}_q)[\ell] = \{P \in E(\mathbb{F}_q) \colon [\ell]P = \mathcal{O}\}$ has order ℓ
- $E(\mathbb{F}_q)/\ell E(\mathbb{F}_q) = \{P + \ell E(\mathbb{F}_q) \colon P \in E(\mathbb{F}_q)\}$ has order ℓ
- $\mathbb{F}_q^*/(\mathbb{F}_q^*)^\ell$ has order ℓ

The multiplication-by-ℓ map $[\ell]$ on $E(\mathbb{F}_q)$ has image $\ell E(\mathbb{F}_q)$ and kernel $E(\mathbb{F}_q)[\ell]$.
$E(\mathbb{F}_q)/\ell E(\mathbb{F}_q)$ is a notation for an *equivalence relation* $P, Q \in G/H \iff P - Q \in H$
$P, Q \in E(\mathbb{F}_q)/\ell E(\mathbb{F}_q) \iff P - Q \in \ell E(\mathbb{F}_q)$
$E(\mathbb{F}_q)/\ell E(\mathbb{F}_q)$ is the quotient of $E(\mathbb{F}_q)$ by the image of $[\ell]$, and has order ℓ.

Tate pairing

$\ell \mid p^n - 1,\ E[\ell] \subset E(\mathbb{F}_{p^n})$
$P \in \mathbf{G}_1 \subset E(\mathbb{F}_{p^n})[\ell], Q \in \mathbf{G}_2 \subset E(\mathbb{F}_{p^n})[\ell]$.

Definition: Tate pairing

Let D_Q be a divisor such that $\mathrm{sum}(D_Q) = Q$, $\deg(D_Q) = 0$, and $P, \mathcal{O} \notin \mathrm{supp}(D_Q)$.
Let f_P be a function whose divisor is $\mathrm{Div}(f_P) = [\ell]P - [\ell]\mathcal{O}$.
One has $\mathrm{supp}(f_P) \cap \mathrm{supp}(D_Q) = \emptyset$.

$$e_{\mathrm{Tate}}(P, Q) = f_P(D_Q) \in \mathbb{F}_{p^n}^* / (\mathbb{F}_{p^n}^*)^\ell$$

Example: choose some point $R \neq \mathcal{O}, P$ s.t. $R + Q \neq \mathcal{O}, P$ and set
$D_Q = (Q + R) - (R)$, then

$$e_{\mathrm{Tate}}(P, Q) = \frac{f_P(Q + R)}{f_P(R)} .$$

(Modified) Tate pairing

Avoid equivalence classes:
need one representative of the equivalence class instead.
Ensure the pairing is non-degenerate: $\mathbf{G}_1 \cap \mathbf{G}_2 = \mathcal{O}$

$$E[\ell] = \mathbb{Z}/\ell\mathbb{Z} \times \mathbb{Z}\ell\mathbb{Z} = \mathbf{G}_1 \times \mathbf{G}_2$$

Let $P \in \mathbf{G}_1 = E(\mathbb{F}_p)[\ell]$, $Q \in \mathbf{G}_2 \subset E(\mathbb{F}_{p^n})[\ell]$.
Let $f_{\ell,P}$ the function s. t. $\mathrm{Div}(f_{\ell,P}) = \ell(P) - \ell(\mathcal{O})$.
Modified Tate pairing (in cryptography):

$$\begin{array}{ccccccc} & E(\mathbb{F}_p)[\ell] & & E(\mathbb{F}_{p^n})[\ell] & & \\ & \wr\| & & \cup & & \\ e_{\mathrm{Tate}}: & \mathbf{G}_1 & \times & \mathbf{G}_2 & \to & \boldsymbol{\mu}_\ell \subset \mathbb{F}_{p^n}^* \\ & & (P, Q) & & \mapsto & (f_{\ell,P}(Q))^{\frac{p^n-1}{\ell}} \end{array}$$

(Modified) Tate pairing

Final exponentiation to avoid equivalence classes in $\mathbb{F}_{p^n}^*$

$x \mapsto x^{(p^n-1)/\ell}$ clears the cofactors in $(\mathbb{F}_{p^n}^*)^\ell$.

Consider the ℓ-powering $x \mapsto x^\ell$ as an endomorphism of $\mathbb{F}_{p^n}^*$

- $(\cdot)^\ell$ has image $(\mathbb{F}_{p^n}^*)^\ell$ and kernel $\mu_\ell = \{x \in \mathbb{F}_{p^n}^* : x^\ell = 1\}$
- $x_1 \equiv x_2 \in \mathbb{F}_{p^n}^*/(\mathbb{F}_{p^n}^*)^\ell \iff x_1 \cdot x_2^{-1} \in (\mathbb{F}_{p^n}^*)^\ell \iff (x_1/x_2)^{(p^n-1)/\ell} = 1$

Replace Divisor D_Q by point Q in the evaluation

Replace $f_{\ell,P}(Q+R)/f_{\ell,P}(R)$ by $(f_{\ell,P}(Q))^{\frac{p^n-1}{\ell}}$
Lemma 26.3.11 in Galbraith's book

(Modified) Tate pairing

Miller loop

$$f_{i+j,Q} = f_{i,Q} f_{j,Q} \frac{\ell_{[i]Q,[j]Q}}{v_{[i+j]Q}},$$

Input: integer s, points P, Q of order s

Output: $m = f_{s,P}(Q)$, where $\mathrm{Div}(f) = s(P) - s(\mathcal{O})$

$m \leftarrow 1$; $S \leftarrow P$;

for *b from the second most significant bit of s to the least* **do**

  $\ell \leftarrow \ell_{S,S}(Q)$; $S \leftarrow [2]S$; // Double Line

  $v \leftarrow v_{[2]S}(Q)$; // Vertical Line

  $m \leftarrow m^2 \cdot \ell / v$; // Update 1

  if $b = 1$ **then**

    $\ell \leftarrow \ell_{S,P}(Q)$; $S \leftarrow S + P$; // Add Line

    $v \leftarrow v_{S+Q}(Q)$; // Vertical Line

    $m \leftarrow m \cdot \ell / v$; // Update 2

return m;

(Optimal) ate pairing

$$m = f_{s,P}(Q)$$

Tate: $s = \ell$

ate: $s = t-1,\ |t| < 2\sqrt{\ell}$

$$f_{\ell,P}(Q)^m = f_{\ell\cdot m}(Q)$$

$$\underbrace{f_{\ell\cdot m,P}(Q) = f^m_{\ell,Q} \cdot f_{m,\underbrace{[\ell]Q}_{\mathcal{O}}}}_{1}$$

Optimal (Vercauteren eprint 2008/096)

$$L := \begin{pmatrix} r & 0 & 0 & \cdots & 0 \\ -q & 1 & 0 & \cdots & 0 \\ -q^2 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \ddots & \\ -q^{\varphi(k)-1} & 0 & \ldots & 0 & 1 \end{pmatrix}$$

BLS12: $f_{s,P}(Q) = f_{u,P}(Q)$

BN: $f_{s,P}(Q) = f_{6u^2+2,P}(Q) \cdot l_{[6u+2]P,\pi(P)}(Q) \cdot l_{[6u+2]P+\pi(P),-\pi^2(P)}(Q)$

Pairing-friendly curves

Miyaji—Nakabayashi—Takano
MNT

When n is small i.e. $1 \leqslant n \leqslant\sim 50$, the curve is *pairing-friendly*.
This is very rare: For a given curve, $\log n \sim \log \ell$ (Balasubramanian–Koblitz).

$$E/\mathbb{F}_p: \; y^2 = x^3 + ax + b, \; a, b \in \mathbb{F}_p, \; p \geq 5$$

$$a = \frac{3j_0}{1728 - j_0}, \quad b = \frac{2j_0}{1728 - j_0}, \quad j_0 \neq 0, 1728$$

$$H_D(j_0) = 0 \pmod{p} \qquad DV^2 = 4p - t^2$$

k	param	MNT
3	$t(u)$	$-1 \pm 6u$
	$r(u)$	$12u^2 \mp 6u + 1$
	$p(u)$	$12u^2 - 1$
	Dy^2	$12u^2 \pm 12u - 5$
4	$t(u)$	$-u, \; u+1$
	$r(u)$	$u^2 + 2u + 2, \; u^2 + 1$
	$p(u)$	$u^2 + u + 1$
	Dy^2	$3u^2 + 4u + 4$
6	$t(u)$	$1 \pm 2u$
	$r(u)$	$4u^2 \mp 2u + 1$
	$p(u)$	$4u^2 + 1$
	Dy^2	$12u^2 - 4u + 3$

$$\#E(\mathbb{F}_{p(u)}) = p(u) + 1 - t(u), \quad \ell(u) \mid \#E(\mathbb{F}_{p(u)})$$

$$DV(u)^2 = 4p(u) - t(u)^2$$

e.g. k=6: $U^2 - 3DW^2 = -8$

Pairing-friendly curves

Barreto—Naehrig
BN (eprint 2005/133)

$$E_{BN}: \; y^2 = x^3 + b, \; p \equiv 1 \bmod 3, \; D = -3 \text{ (ordinary)}$$

$$\begin{aligned} p &= 36x^4 + 36x^3 + 24x^2 + 6x + 1 \\ t &= 6x^2 + 1 \\ \ell &= p + 1 - t = 36x^4 + 36x^3 + 18x^2 + 6x + 1 \\ t^2 - 4p &= -3(6x^2 + 4x + 1)^2 \; \to \; \text{no CM method needed} \end{aligned}$$

Comes from the Aurifeuillean factorization of Φ_{12} :
$\Phi_{12}(6x^2) = \ell(x)\ell(-x)$

Security level	$\log_2 \ell$	finite field	n	$\log_2 p$	$\deg P, \; p = P(u)$	ρ
102	256	3072	12	256	4	1
123	384	4608	12	384	4	1
132	448	5376	12	448	4	1

Pairing-friendly curves

Barreto—Lynn—Scott
BLS (eprint 2002/088)

Barreto, Lynn, Scott method.
Becomes more and more popular, replacing BN curves

$$E_{BLS}: \; y^2 = x^3 + b, \; p \equiv 1 \bmod 3, \; D = -3 \text{ (ordinary)}$$

$$\begin{aligned}
p &= (u-1)^2/3(u^4 - u^2 + 1) + u \\
t &= u + 1 \\
r &= (u^4 - u^2 + 1) = \Phi_{12}(u) \\
p + 1 - t &= (u-1)^2/3(u^4 - u^2 + 1) \\
t^2 - 4p &= -3y(u)^2 \; \rightarrow \; \text{no CM method needed}
\end{aligned}$$

BLS12-381 with seed `-0xd201000000010000`

Complex Multiplication Method

CM method

Hard problem to compute the curve coefficients (a, b) given a prime p and a trace t.
The other way: given p and (a, b) in $E/\mathbb{F}_p: \ : y^2 = x^3 + ax + b$ and computing the order $\#E(\mathbb{F}_p)$ is done with the SEA algorithm (Schroof–Elkies–Atkin).
The *CM* method computes a j-invariant, given p, t.

1. Compute the discriminant $-D$ as the square-free part in $t^2 - 4p = -Dy^2$
2. If $D \equiv 1, 2 \bmod 4$, $D \leftarrow 4D$
3. Compute a Hilbert Class Polynomial $H_{-D}(X) \bmod p$ with Sutherland's software `classpoly` at `https://math.mit.edu/~drew/`
   Or with PARI-GP
4. Compute a root j_0 of $H_{-D}(X) \bmod p$
5. Set $E\colon y^2 = x^3 + \frac{3j_0}{1728 - j_0}x + \frac{2j_0}{1728 - j_0}$

Some newer strange families

- “Optimized and sec ... table for one layer proof composition”
- “Families of SNARK ... EH—Guillevic 2022*
- “A survey of elliptic ... EH—Guillevic 2022*
- “Families of prime-c ... dded curves on pairing-friendly curv

A 2-cycle of elliptic curves:

$$\begin{tikzcd} \boxed{E_1(\mathbb{F}_{q})} \arrow[d, bend left, "\#E_0(\mathbb{F}_{p}) = q"] \\ \boxed{E_0(\mathbb{F}_{p})} \arrow[u, bend left, "\#E_1(\mathbb{F}_{q}) = p"] \end{tikzcd}$$

A 2-chain of elliptic curves:

$$\begin{tikzcd} \boxed{E_1(\mathbb{F}_{q})} \\ \boxed{E_0(\mathbb{F}_{p})} \arrow[u, "\#E_1(\mathbb{F}_{q}) = h \cdot p"'] \end{tikzcd}$$

Implementation

Miller loop

$\mathbb{F}_p, \mathbb{F}_{p^n}$ arithmetics

EH—Botrel 2022

Karatsuba, Toom-Cook-3

Lazy reduction

Pornin's or Bernstein–Yang binary GCD

Input: integer s, points P, Q of order s

Output: $m = f_{s,P}(Q)$, where $\mathrm{Div}(f) = s(P) - s(\mathcal{O})$

$m \leftarrow 1$; $S \leftarrow P$;

for *b from the second most significant bit of s to the least* **do**

  $\ell \leftarrow \ell_{S,S}(Q)$; $S \leftarrow [2]S$; `// Double Line`

  $v \leftarrow v_{[2]S}(Q)$; `// Vertical Line`

  $m \leftarrow m^2 \cdot \ell / v$; `// Update 1`

  if $b = 1$ **then**

    $\ell \leftarrow \ell_{S,P}(Q)$; $S \leftarrow S + P$; `// Add Line`

    $v \leftarrow v_{S+Q}(Q)$; `// Vertical Line`

    $m \leftarrow m \cdot \ell / v$; `// Update 2`

return m;

Implementation

Miller loop

$$E(\mathbb{F}_{p^n}) \supset \boxed{\mathbb{G}_2 \cong E(\mathbb{F}_{p^{n/d}})[\ell]}$$

$d \in \{2, 4, 6\}$

choose $n = 0 \pmod{d}$

e.g. BLS12: $d = 6, n = 12, n/d = 2$

$\boxed{\mathbb{G}_2 \cong E(\mathbb{F}_{p^2})[\ell]}$

BN06, AKL+11,ABLR14

Input: integer s, points P, Q of order s
Output: $m = f_{s,P}(Q)$, where $\mathrm{Div}(f) = s(P) - s(\mathcal{O})$

```
m ← 1; S ← P;
for b from the second most significant bit of s to the least do
    ℓ ← ℓ_{S,S}(Q); [S ← [2]S];                    // Double Line
    v ← v_{[2]S}(Q) ;                               // Vertical Line
    m ← m^2 · ℓ/v;                                  // Update 1
    if b = 1 then
        ℓ ← ℓ_{S,P}(Q); [S ← S + P];                // Add Line
        v ← v_{S+Q}(Q) ;                            // Vertical Line
        m ← m · ℓ/v ;                               // Update 2
return m;
```

Projective coordinates

Implementation

Miller loop

Barreto — Kim — Lynn — Scott
eprint 2002/008

$$\boxed{n \text{ even}}$$

$$v = x_Q - c \in \mathbb{F}_{p^{n/d}}, d \in \{2,4,6\}$$

$$v^{n/d-1} = 1 \text{ and } v^{n/2-1} = 1$$

$$v^{(n-1)/\ell} = v^{(n/2-1)(n/2+1)/\ell}$$

Input: integer s, points P, Q of order s
Output: $m = f_{s,P}(Q)$, where $\mathrm{Div}(f) = s(P) - s(\mathcal{O})$

```
m ← 1; S ← P;
for b from the second most significant bit of s to the least do
    ℓ ← ℓ_{S,S}(Q); S ← [2]S ;          // Double Line
    v ← v_{[2]S}(Q) ;                    // Vertical Line
    m ← m^2 · ℓ/v ;                      // Update 1
    if b = 1 then
        ℓ ← ℓ_{S,P}(Q); S ← S + P ;      // Add Line
        v ← v_{S+Q}(Q) ;                 // Vertical Line
        m ← m · ℓ/v ;                    // Update 2
return m;
```

Implementation

Miller loop

ABLR14, Scott19

$$l_{S,P}(Q) : y_Q + a x_Q + b \in \mathbb{F}_{p^n}$$

$$l_{S,P}(Q) : (y_Q + a x_Q + b) \cdot f_? \in \mathbb{F}_{p^n}$$

sparse multiplication in $\mathbb{F}_{p^n}$

Input: integer s, points P, Q of order s

Output: $m = f_{s,P}(Q)$, where $\mathrm{Div}(f) = s(P) - s(\mathcal{O})$

$m \leftarrow 1$; $S \leftarrow P$;

for *b from the second most significant bit of s to the least* **do**

  $\ell \leftarrow \ell_{S,S}(Q)$; $S \leftarrow [2]S$; `// Double Line`

  $v \leftarrow v_{[2]S}(Q)$; `// Vertical Line`

  $m \leftarrow m^2 \cdot \ell / v$; `// Update 1`

  if $b = 1$ **then**

    $\ell \leftarrow \ell_{S,P}(Q)$; $S \leftarrow S + P$; `// Add Line`

    $v \leftarrow v_{S+Q}(Q)$; `// Vertical Line`

    $m \leftarrow m \cdot \ell / v$; `// Update 2`

return m;

Implementation

Final exponentiation

e.g. BLS12, BN

$$(p^{12}-1)/\ell = (p^6-1)(p^2+1)(p^4-p^2+1)/\ell$$

$$3(p(u)^4-p(u)^2+1)/\ell(u) = (u-1)^2(u+p)(u^2+p^2-1)+3$$

$$2u(6u+3u+1)(p(u)^4-p(u)^2+1)/\ell(u) = \dots$$

$$\frac{(p^n-1)}{\ell} = \frac{p^n-1}{\Phi_n(p)} \cdot \frac{\Phi_n(p)}{\ell}$$

easy part: a polynomial in p with small coefficients (Frobenius maps)
e.g. (BLS12): `1F2 + 1Conj + 1Inv + 1Mul` in $\mathbb{F}_{p^{12}}$

hard part: More expensive. Vectorial or lattice-based
Optimizations [HHT, AFK+13, GF16]
dominating cost: $\boxed{\texttt{CycloSqr}}$ [GS10, Kar13] + `Mul` in $\mathbb{F}_{p^k}$

$$\text{ML: } f \mapsto f^{\frac{p^n-1}{\Phi_n(p)}} = f'$$

$$f'^{\Phi_n(p)} = f^{p^n-1} = 1$$

Implementation

gnark

https://github.com/Consensys/gnark-crypto
https://github.com/Consensys/gnark

Frontend
(write a "circuit")

Backend
(proof generation & verification)

- Groth16, PLONK w/ KZG (or FRI)
- std: hashes, signatures, pairings, commitments…
- Native and non-native field arithmetic

Pairing and elliptic curve cryptography

gnark-crypto

- BN254, BLS12-381, BLS12-377/BW6-761, BLS24…
- Fast cryptographic primitives (MSM, pairings,...)
- KZG, FRI, Plookup…
- Sumcheck (GKR)

Field arithmetic (~big integer library)

gnark-crypto

- 768-bit, 384-bit, 256-bit, goldilocks… on multi-targets
- SotA mul, Pornin's inverse, FFT…

Implementation

Gnark

Audits

- Kudelski Security - October 2022 - gnark-crypto (contracted by Algorand Foundation)
- Sigma Prime - May 2023 - gnark-crypto KZG (contracted by Ethereum Foundation)
- Consensys Diligence - June 2023 - gnark PLONK Solidity verifier
- LeastAuthority - August 2023 - gnark Groth16 Solidity verifier template (contracted by Worldcoin)
- OpenZeppelin - November 2023 - gnark PLONK Solidity verifier template
- ZKSecurity.xyz - May 2024 - gnark standard library
- OpenZeppelin - June 2024 - gnark PLONK prover and verifier
- LeastAuthority - September 2024 - gnark general and GKR
- LeastAuthority - November 2024 - Linea zkEVM

Implementation

Alternative ways

- Elliptic nets (slow) https://eprint.iacr.org/2006/392
- Cubic bi-extensions (fast?) https://eprint.iacr.org/2024/517

Funny pairings

Pairings on non-pairing-friendly pairings

$$E \cong \mathbb{Z}_{e_1} \bigoplus \mathbb{Z}_{e_2 \cdot r} \text{ and } e_2 \mid p-1$$

$$f_{e_1,P_1}(P)^{(p-1)/e_1} =_? 1 \quad \text{and} \quad f_{e_2,P_2}(P)^{(p-1)/e_2} =_? 1$$

$e_i = 2: \; f_{2,P_i} = X - X_{P_i}$

$f \mapsto f^{(p-1)/2}$ is a Legendre symbol

$e_i = 3: \; f_{3,P_i} = Y - Y_{P_i}$

$f \mapsto f^{(p-1)/3}$ is a cubic residue symbol

$e_i \leq 16$

Euclidean-type power-residue symbol?

Koshelev 2022
Koshelev—EH—Fotiadis 2025

Funny pairings

“Proving” pairings aka pairings in SNARK circuit

- “Pairings in R1CS” *EH 2023*: torus-based arithmetic, affine coordinates,…
- “On proving pairings” *Eagen—Novakovic 2024*: ~no final exponentiation…
- “*Garaga, gnark*”: Fp^n arithmetic costs ~as much as Fp arithmetic

Thank you

https://linea.build/

https://gnark.io/

- youssef.elhousni@consensys.net
- TG: @ElMarroqui
- X: @YoussefElHousn3
- GH: @yelhousni
- yelhousni.github.io