Subgroup membership testing on elliptic curves via the Tate pairing

Dimitri Koshelev

Parallel Computation Laboratory, École Normale Supérieure de Lyon

12/07/2023, SIAM AG23
TU/e, Eindhoven

ÉCOLE NORMALE SUPÉRIEURE DE LYON

Introduction

Cryptosystems on elliptic curves E over finite fields $\mathbb{F}_q$ are frequently deployed not in the entire $\mathbb{F}_q$-point group $E(\mathbb{F}_q)$, but in its subgroup $\mathbb{G}$ of large prime order r and with cofactor c.

Introduction

Cryptosystems on elliptic curves E over finite fields $\mathbb{F}_q$ are frequently deployed not in the entire $\mathbb{F}_q$-point group $E(\mathbb{F}_q)$, but in its subgroup $\mathbb{G}$ of large prime order r and with cofactor c.

To be protected against the *subgroup attack*, when receiving a point from a communication channel, it is necessary to make sure that it belongs to $\mathbb{G}$, not only to $E(\mathbb{F}_q)$.

Introduction

Cryptosystems on elliptic curves E over finite fields $\mathbb{F}_q$ are frequently deployed not in the entire $\mathbb{F}_q$-point group $E(\mathbb{F}_q)$, but in its subgroup $\mathbb{G}$ of large prime order r and with cofactor c.

To be protected against the *subgroup attack*, when receiving a point from a communication channel, it is necessary to make sure that it belongs to $\mathbb{G}$, not only to $E(\mathbb{F}_q)$.

In fact, to thwart the given attack it is often sufficient to just multiply an obtained point by c if the latter is small (as in the current talk).

Introduction

Cryptosystems on elliptic curves E over finite fields $\mathbb{F}_q$ are frequently deployed not in the entire $\mathbb{F}_q$-point group $E(\mathbb{F}_q)$, but in its subgroup $\mathbb{G}$ of large prime order r and with cofactor c.

To be protected against the *subgroup attack*, when receiving a point from a communication channel, it is necessary to make sure that it belongs to $\mathbb{G}$, not only to $E(\mathbb{F}_q)$.

In fact, to thwart the given attack it is often sufficient to just multiply an obtained point by c if the latter is small (as in the current talk).

Nevertheless, this solution is not a panacea. For example, in the signature scheme, used in CryptoNote cryptocurrencies, it could lead to double-spending if any of the malicious users noticed this bug.

Naive subgroup check

An obvious way to test membership in $\mathbb{G}$ is to multiply a point by r.

Naive subgroup check

An obvious way to test membership in $\mathbb{G}$ is to multiply a point by r.

Even if the curve enjoys an effectively computable endomorphism, which makes it possible to apply the GLV technique or its variations, the mentioned test is still laborious.

Naive subgroup check

An obvious way to test membership in $\mathbb{G}$ is to multiply a point by r.

Even if the curve enjoys an effectively computable endomorphism, which makes it possible to apply the GLV technique or its variations, the mentioned test is still laborious.

More concretely, it performs $\Theta\big(\log_2(r)\big)$ additions in $E(\mathbb{F}_q)$. Hence, its bit complexity equals $\Theta\big(\log_2(r)M\big)$ with a non-little constant behind Θ, where M is the bit complexity of a multiplication in $\mathbb{F}_q$.

Naive subgroup check

An obvious way to test membership in $\mathbb{G}$ is to multiply a point by r.

Even if the curve enjoys an effectively computable endomorphism, which makes it possible to apply the GLV technique or its variations, the mentioned test is still laborious.

More concretely, it performs $\Theta\big(\log_2(r)\big)$ additions in $E(\mathbb{F}_q)$. Hence, its bit complexity equals $\Theta\big(\log_2(r)M\big)$ with a non-little constant behind Θ, where M is the bit complexity of a multiplication in $\mathbb{F}_q$.

We can suppose that $M = \Theta(\ell^2)$ at least for the popular choice $\ell := \log_2(q) \approx 256$. Indeed, it is widely recognized that for such $\mathbb{F}_q$ the “school” multiplication algorithm is more practical.

Naive subgroup check

An obvious way to test membership in $\mathbb{G}$ is to multiply a point by r.

Even if the curve enjoys an effectively computable endomorphism, which makes it possible to apply the GLV technique or its variations, the mentioned test is still laborious.

More concretely, it performs $\Theta\big(\log_2(r)\big)$ additions in $E(\mathbb{F}_q)$. Hence, its bit complexity equals $\Theta\big(\log_2(r)M\big)$ with a non-little constant behind Θ, where M is the bit complexity of a multiplication in $\mathbb{F}_q$.

We can suppose that $M = \Theta(\ell^2)$ at least for the popular choice $\ell := \log_2(q) \approx 256$. Indeed, it is widely recognized that for such $\mathbb{F}_q$ the "school" multiplication algorithm is more practical.

Since $c \approx 1$ by our assumption, i.e., $\ell \approx \log_2(r)$, we eventually get the bit complexity $\Theta(\ell^3)$.

Notation

Consider an elliptic curve $E\colon y^2 = x^3 + a_2x^2 + a_4x + a_6$ (with the point $\mathcal{O} := (0 : 1 : 0)$ at infinity) over a finite field $\mathbb{F}_q$ of char. > 2.

Notation

Consider an elliptic curve $E\colon y^2 = x^3 + a_2x^2 + a_4x + a_6$ (with the point $\mathcal{O} := (0 : 1 : 0)$ at infinity) over a finite field $\mathbb{F}_q$ of char. > 2.

The rational point group $E(\mathbb{F}_q) \simeq \mathbb{Z}/n_0 \times \mathbb{Z}/n_1$, where $n_1 \mid n_0$.

Notation

Consider an elliptic curve $E\colon y^2 = x^3 + a_2x^2 + a_4x + a_6$ (with the point $\mathcal{O} := (0:1:0)$ at infinity) over a finite field $\mathbb{F}_q$ of char. > 2.

The rational point group $E(\mathbb{F}_q) \simeq \mathbb{Z}/n_0 \times \mathbb{Z}/n_1$, where $n_1 \mid n_0$.

As always in discrete logarithm cryptography, there is a subgroup $\mathbb{G} \subset E(\mathbb{F}_q)$ of large prime order r such that $r \mid\mid n_0$, but $r \nmid n_1$.

Notation

Consider an elliptic curve $E\colon y^2 = x^3 + a_2x^2 + a_4x + a_6$ (with the point $\mathcal{O} := (0:1:0)$ at infinity) over a finite field $\mathbb{F}_q$ of char. > 2.

The rational point group $E(\mathbb{F}_q) \simeq \mathbb{Z}/n_0 \times \mathbb{Z}/n_1$, where $n_1 \mid n_0$.

As always in discrete logarithm cryptography, there is a subgroup $\mathbb{G} \subset E(\mathbb{F}_q)$ of large prime order r such that $r \mid\mid n_0$, but $r \nmid n_1$.

In other words, $E(\mathbb{F}_q) = \mathbb{G} \times E(\mathbb{F}_q)[e]$, where $e := n_0/r$. So, the order $N := \#E(\mathbb{F}_q) = n_0 n_1$ and the cofactor $c := N/r = en_1$.

Notation

Consider an elliptic curve $E\colon y^2 = x^3 + a_2x^2 + a_4x + a_6$ (with the point $\mathcal{O} := (0:1:0)$ at infinity) over a finite field $\mathbb{F}_q$ of char. > 2.

The rational point group $E(\mathbb{F}_q) \simeq \mathbb{Z}/n_0 \times \mathbb{Z}/n_1$, where $n_1 \mid n_0$.

As always in discrete logarithm cryptography, there is a subgroup $\mathbb{G} \subset E(\mathbb{F}_q)$ of large prime order r such that $r \mid\mid n_0$, but $r \nmid n_1$.

In other words, $E(\mathbb{F}_q) = \mathbb{G} \times E(\mathbb{F}_q)[e]$, where $e := n_0/r$. So, the order $N := \#E(\mathbb{F}_q) = n_0 n_1$ and the cofactor $c := N/r = en_1$.

For the sake of uniformity, put $e_0 := e$ and $e_1 := n_1$. Besides, let $E(\mathbb{F}_q)[e] = \langle P_0 \rangle \times \langle P_1 \rangle$, where $\mathrm{ord}(P_i) = e_i$.

Reduced Tate pairing

For any $k \mid q-1$, the *reduced Tate pairing* can be represented in the form

$$t_k \colon E(\mathbb{F}_q)[k] \times E(\mathbb{F}_q)/kE(\mathbb{F}_q) \to \mu_k \qquad t_k(P,Q) := f_{k,P}(Q)^{(q-1)/k},$$

where $\mu_k \subset \mathbb{F}_q^*$ is the group of all k-th roots of unity, $P \neq Q \neq \mathcal{O}$, and $f_{k,P} \in \mathbb{F}_q(E)$ is a Miller function satisfying the conditions

$$\mathrm{div}(f_{k,P}) = k(P) - k(\mathcal{O}), \qquad \left(\left(\frac{x}{y}\right)^k \cdot f_{k,P}\right)(\mathcal{O}) = 1.$$

Reduced Tate pairing

For any $k \mid q-1$, the *reduced Tate pairing* can be represented in the form

$$t_k\colon E(\mathbb{F}_q)[k] \times E(\mathbb{F}_q)/kE(\mathbb{F}_q) \to \mu_k \qquad t_k(P,Q) := f_{k,P}(Q)^{(q-1)/k},$$

where $\mu_k \subset \mathbb{F}_q^*$ is the group of all k-th roots of unity, $P \neq Q \neq \mathcal{O}$, and $f_{k,P} \in \mathbb{F}_q(E)$ is a Miller function satisfying the conditions

$$\mathrm{div}(f_{k,P}) = k(P) - k(\mathcal{O}), \qquad \left(\left(\frac{x}{y}\right)^k \cdot f_{k,P}\right)(\mathcal{O}) = 1.$$

The values $f_{k,P}(Q)$ are recursively computed by means of Miller's algorithm with the cost of $\Theta(\log_2(k))$ operations in $\mathbb{F}_q$.

Reduced Tate pairing

For any $k \mid q-1$, the *reduced Tate pairing* can be represented in the form

$$t_k \colon E(\mathbb{F}_q)[k] \times E(\mathbb{F}_q)/kE(\mathbb{F}_q) \to \mu_k \qquad t_k(P,Q) := f_{k,P}(Q)^{(q-1)/k},$$

where $\mu_k \subset \mathbb{F}_q^*$ is the group of all k-th roots of unity, $P \neq Q \neq \mathcal{O}$, and $f_{k,P} \in \mathbb{F}_q(E)$ is a Miller function satisfying the conditions

$$\mathrm{div}(f_{k,P}) = k(P) - k(\mathcal{O}), \qquad \left(\left(\frac{x}{y}\right)^k \cdot f_{k,P}\right)(\mathcal{O}) = 1.$$

The values $f_{k,P}(Q)$ are recursively computed by means of Miller's algorithm with the cost of $\Theta(\log_2(k))$ operations in $\mathbb{F}_q$.

Throughout the rest of the talk, we will assume that $e \mid q-1$.

The k-th power residue symbol

The final exponentiation of the pairing t_k is nothing but the *k-th power residue symbol* $\left(\frac{\alpha}{q}\right)_k := \alpha^{(q-1)/k}$ with $\alpha := f_{k,P}(Q)$.

The k-th power residue symbol

The final exponentiation of the pairing t_k is nothing but the *k-th power residue symbol* $\left(\frac{\alpha}{q}\right)_k := \alpha^{(q-1)/k}$ with $\alpha := f_{k,P}(Q)$.

In particular, for $k = 2$ we deal with the ordinary Legendre symbol.

The k-th power residue symbol

The final exponentiation of the pairing t_k is nothing but the *k-th power residue symbol* $\left(\frac{\alpha}{q}\right)_k := \alpha^{(q-1)/k}$ with $\alpha := f_{k,P}(Q)$.

In particular, for $k = 2$ we deal with the ordinary Legendre symbol.

It is worth saying that we always can batch the inversion and symbol computation, since

$$\left(\frac{\alpha_0/\alpha_1}{q}\right)_k = \left(\frac{\alpha_0\alpha_1^{k-1}}{q}\right)_k$$

given $\alpha_i \in \mathbb{F}_q^*$.

The k-th power residue symbol

The final exponentiation of the pairing t_k is nothing but the *k-th power residue symbol* $\left(\frac{\alpha}{q}\right)_k := \alpha^{(q-1)/k}$ with $\alpha := f_{k,P}(Q)$.

In particular, for $k = 2$ we deal with the ordinary Legendre symbol.

It is worth saying that we always can batch the inversion and symbol computation, since

$$\left(\frac{\alpha_0/\alpha_1}{q}\right)_k = \left(\frac{\alpha_0\alpha_1^{k-1}}{q}\right)_k$$

given $\alpha_i \in \mathbb{F}_q^*$.

At least for $k \leqslant 11$, the symbol can be determined by Euclidean-type algorithms whose bit complexity amounts to $O(\ell^2)$.

The k-th power residue symbol

The final exponentiation of the pairing t_k is nothing but the *k-th power residue symbol* $\left(\frac{\alpha}{q}\right)_k := \alpha^{(q-1)/k}$ with $\alpha := f_{k,P}(Q)$.

In particular, for $k = 2$ we deal with the ordinary Legendre symbol.

It is worth saying that we always can batch the inversion and symbol computation, since

$$\left(\frac{\alpha_0/\alpha_1}{q}\right)_k = \left(\frac{\alpha_0\alpha_1^{k-1}}{q}\right)_k$$

given $\alpha_i \in \mathbb{F}_q^*$.

At least for $k \leqslant 11$, the symbol can be determined by Euclidean-type algorithms whose bit complexity amounts to $O(\ell^2)$.

Conversely, if k is not small, then the exponentiation is seemingly the best way to compute $\left(\frac{\alpha}{q}\right)_k$.

Lemma underlying the new subgroup test

For compactness of notation, let's also define the homomorphisms

$$h_i \colon E(\mathbb{F}_q) \to \mu_{e_i} \qquad h_i(Q) := t_e(P_i, Q) = t_{e_i}(P_i, Q).$$

Lemma underlying the new subgroup test

For compactness of notation, let's also define the homomorphisms

$$h_i \colon E(\mathbb{F}_q) \to \mu_{e_i} \qquad h_i(Q) := t_e(P_i, Q) = t_{e_i}(P_i, Q).$$

For our purpose, it is unnecessary to know the values $h_i(P_i)$, hence we can benefit from the above pairing form.

Lemma underlying the new subgroup test

For compactness of notation, let's also define the homomorphisms

$$h_i \colon E(\mathbb{F}_q) \to \mu_{e_i} \qquad h_i(Q) := t_e(P_i, Q) = t_{e_i}(P_i, Q).$$

For our purpose, it is unnecessary to know the values $h_i(P_i)$, hence we can benefit from the above pairing form.

Lemma

There are the equalities $\mathbb{G} = eE(\mathbb{F}_q) = \ker(h_0) \cap \ker(h_1)$.

Lemma underlying the new subgroup test

For compactness of notation, let's also define the homomorphisms

$$h_i \colon E(\mathbb{F}_q) \to \mu_{e_i} \qquad h_i(Q) := t_e(P_i, Q) = t_{e_i}(P_i, Q).$$

For our purpose, it is unnecessary to know the values $h_i(P_i)$, hence we can benefit from the above pairing form.

Lemma

There are the equalities $\mathbb{G} = eE(\mathbb{F}_q) = \ker(h_0) \cap \ker(h_1)$.

Proof.

Given a point $Q \in \mathbb{G}$, we see that $Q = eR$ for $R := (e^{-1} \bmod r)Q$. The opposite inclusion $\mathbb{G} \supset eE(\mathbb{F}_q)$ is even more trivial.

Lemma underlying the new subgroup test

For compactness of notation, let's also define the homomorphisms

$$h_i\colon E(\mathbb{F}_q) \to \mu_{e_i} \qquad h_i(Q) := t_e(P_i, Q) = t_{e_i}(P_i, Q).$$

For our purpose, it is unnecessary to know the values $h_i(P_i)$, hence we can benefit from the above pairing form.

Lemma

There are the equalities $\mathbb{G} = eE(\mathbb{F}_q) = \ker(h_0) \cap \ker(h_1)$.

Proof.

Given a point $Q \in \mathbb{G}$, we see that $Q = eR$ for $R := (e^{-1} \bmod r)Q$. The opposite inclusion $\mathbb{G} \supset eE(\mathbb{F}_q)$ is even more trivial.

Further, the Tate pairing is non-degenerate. Consequently, a point $Q \in E(\mathbb{F}_q)$ in fact belongs to $eE(\mathbb{F}_q)$ if and only if $t_e(P, Q) = 1$ for all $P \in E(\mathbb{F}_q)[e]$ or, equivalently, $h_0(Q) = h_1(Q) = 1$.

Basic examples

The case $e_0 = 2$, $e_1 = 1$. Without loss of generality,
$E\colon y^2 = x(x^2 + a_2x + a_4)$, where $a_2^2 - 4a_4,\ a_4 \notin (\mathbb{F}_q^*)^2$.
The curves E are so-called *double-odd curves*. Clearly,
$P_0 = (0, 0)$ and $f_{2,P_0} = x$.

Basic examples

The case $e_0 = 2$, $e_1 = 1$. Without loss of generality,
$E\colon y^2 = x(x^2 + a_2x + a_4)$, where $a_2^2 - 4a_4,\ a_4 \notin (\mathbb{F}_q^*)^2$.
The curves E are so-called *double-odd curves*. Clearly, $P_0 = (0, 0)$ and $f_{2,P_0} = x$.

The previous lemma states that a point $(x, y) \in E(\mathbb{F}_q)$ lies in $\mathbb{G}$ if and only if $x \in (\mathbb{F}_q^*)^2$. We obtain a folklore subgroup membership test.

Basic examples

The case $e_0 = 2$, $e_1 = 1$. Without loss of generality,
$E\colon y^2 = x(x^2 + a_2 x + a_4)$, where $a_2^2 - 4a_4$, $a_4 \notin (\mathbb{F}_q^*)^2$.
The curves E are so-called *double-odd curves*. Clearly,
$P_0 = (0, 0)$ and $f_{2,P_0} = x$.

The previous lemma states that a point $(x, y) \in E(\mathbb{F}_q)$ lies in $\mathbb{G}$ if and only if $x \in (\mathbb{F}_q^*)^2$. We obtain a folklore subgroup membership test.

The case $e_0 = e_1 = 2$. In this one, $E\colon y^2 = x(x - \alpha_1)(x - \alpha_2)$,
where $\alpha_1, \alpha_2 \in \mathbb{F}_q^*$, but $\alpha_1\alpha_2 \notin (\mathbb{F}_q^*)^2$. Putting $\alpha_0 := 0$ in addition, we get the points $P_i = (\alpha_i, 0)$.

Basic examples

The case $e_0 = 2$, $e_1 = 1$. Without loss of generality,
$E\colon y^2 = x(x^2 + a_2x + a_4)$, where $a_2^2 - 4a_4,\ a_4 \notin (\mathbb{F}_q^*)^2$.
The curves E are so-called *double-odd curves*. Clearly,
$P_0 = (0,0)$ and $f_{2,P_0} = x$.

The previous lemma states that a point $(x, y) \in E(\mathbb{F}_q)$ lies in $\mathbb{G}$ if and only if $x \in (\mathbb{F}_q^*)^2$. We obtain a folklore subgroup membership test.

The case $e_0 = e_1 = 2$. In this one, $E\colon y^2 = x(x - \alpha_1)(x - \alpha_2)$, where $\alpha_1, \alpha_2 \in \mathbb{F}_q^*$, but $\alpha_1\alpha_2 \notin (\mathbb{F}_q^*)^2$. Putting $\alpha_0 := 0$ in addition, we get the points $P_i = (\alpha_i, 0)$.

Consequently, $f_{2,P_i} = x - \alpha_i$. It is readily seen that $x - \alpha_2 \in (\mathbb{F}_q^*)^2$ automatically whenever $x - \alpha_i \in (\mathbb{F}_q^*)^2$ for $i \in \{0, 1\}$.

Some popular elliptic curves of non-prime orders

Let ν be the 2-adicity of $q - 1$, that is, $2^\nu \mid\mid q - 1$.

<table>
<tr><th>Curve</th><th>$\lceil \ell \rceil$</th><th>e_0</th><th>e_1</th><th>ν</th></tr>
<tr><td>Curve25519</td><td>255</td><td>8</td><td rowspan="3">1</td><td>2</td></tr>
<tr><td>Ed448-Goldilocks</td><td>448</td><td>4</td><td>1</td></tr>
<tr><td>Jubjub</td><td rowspan="2">255</td><td>8</td><td rowspan="2">32</td></tr>
<tr><td>Bandersnatch</td><td>2</td><td>2</td></tr>
</table>

Some popular elliptic curves of non-prime orders

Let ν be the 2-adicity of $q-1$, that is, $2^{\nu} \mid\mid q-1$.

<table>
<tr><th>Curve</th><th>$\lceil \ell \rceil$</th><th>e_0</th><th>e_1</th><th>ν</th></tr>
<tr><td>Curve25519</td><td>255</td><td>8</td><td rowspan="3">1</td><td>2</td></tr>
<tr><td>Ed448-Goldilocks</td><td>448</td><td>4</td><td>1</td></tr>
<tr><td>Jubjub</td><td rowspan="2">255</td><td>8</td><td rowspan="2">32</td></tr>
<tr><td>Bandersnatch</td><td>2</td><td>2</td></tr>
</table>

The first two curves were included in the American standard NIST SP 800-186 recently updated. We see that they are unfortunately not appropriate for the new subgroup check.

Some popular elliptic curves of non-prime orders

Let ν be the 2-adicity of $q - 1$, that is, $2^\nu \mid\mid q - 1$.

<table>
<tr><th>Curve</th><th>$\lceil \ell \rceil$</th><th>e_0</th><th>e_1</th><th>ν</th></tr>
<tr><td>Curve25519</td><td>255</td><td>8</td><td rowspan="3">1</td><td>2</td></tr>
<tr><td>Ed448-Goldilocks</td><td>448</td><td>4</td><td>1</td></tr>
<tr><td>Jubjub</td><td rowspan="2">255</td><td>8</td><td rowspan="2">32</td></tr>
<tr><td>Bandersnatch</td><td>2</td><td>2</td></tr>
</table>

The first two curves were included in the American standard NIST SP 800-186 recently updated. We see that they are unfortunately not appropriate for the new subgroup check.

The zk-SNARK-friendly curves Bandersnatch and Jubjub were proposed by the Ethereum and Zcash research teams, respectively. They are currently used in the given cryptocurrencies.

Moving to a finite field extension

Given $i \in \mathbb{N}$, nothing prevents us from applying the base change $E/\mathbb{F}_{q^i}$. Let's introduce the torsion subgroup

$$T(i) := E(\mathbb{F}_{q^i})[e^\infty] = \bigcup_{j=1}^{\infty} E(\mathbb{F}_{q^i})[e^j].$$

Moving to a finite field extension

Given $i \in \mathbb{N}$, nothing prevents us from applying the base change $E/\mathbb{F}_{q^i}$. Let's introduce the torsion subgroup

$$T(i) := E(\mathbb{F}_{q^i})[e^\infty] = \bigcup_{j=1}^{\infty} E(\mathbb{F}_{q^i})[e^j].$$

Whenever i is fixed, the chain $E(\mathbb{F}_{q^i})[e^j]$ is evidently stabilized, starting with a certain $j \in \mathbb{N}$.

Moving to a finite field extension

Given $i \in \mathbb{N}$, nothing prevents us from applying the base change $E/\mathbb{F}_{q^i}$. Let's introduce the torsion subgroup

$$T(i) := E(\mathbb{F}_{q^i})[e^\infty] = \bigcup_{j=1}^{\infty} E(\mathbb{F}_{q^i})[e^j].$$

Whenever i is fixed, the chain $E(\mathbb{F}_{q^i})[e^j]$ is evidently stabilized, starting with a certain $j \in \mathbb{N}$.

Note that $T(i) = E(\mathbb{F}_{q^i})[2^\infty]$ for e equal to a power of 2.

Moving to a finite field extension

Given $i \in \mathbb{N}$, nothing prevents us from applying the base change $E/\mathbb{F}_{q^i}$. Let's introduce the torsion subgroup

$$T(i) := E(\mathbb{F}_{q^i})[e^\infty] = \bigcup_{j=1}^{\infty} E(\mathbb{F}_{q^i})[e^j].$$

Whenever i is fixed, the chain $E(\mathbb{F}_{q^i})[e^j]$ is evidently stabilized, starting with a certain $j \in \mathbb{N}$.

Note that $T(i) = E(\mathbb{F}_{q^i})[2^\infty]$ for e equal to a power of 2.

Like any finite group on an elliptic curve, $T(i) \simeq \mathbb{Z}/e_0(i) \times \mathbb{Z}/e_1(i)$ for some $e_0(i)$, $e_1(i) \in \mathbb{N}$ such that $e_1(i) \mid e_0(i)$.

Moving to a finite field extension

Given $i \in \mathbb{N}$, nothing prevents us from applying the base change $E/\mathbb{F}_{q^i}$. Let's introduce the torsion subgroup

$$T(i) := E(\mathbb{F}_{q^i})[e^\infty] = \bigcup_{j=1}^{\infty} E(\mathbb{F}_{q^i})[e^j].$$

Whenever i is fixed, the chain $E(\mathbb{F}_{q^i})[e^j]$ is evidently stabilized, starting with a certain $j \in \mathbb{N}$.

Note that $T(i) = E(\mathbb{F}_{q^i})[2^\infty]$ for e equal to a power of 2.

Like any finite group on an elliptic curve, $T(i) \simeq \mathbb{Z}/e_0(i) \times \mathbb{Z}/e_1(i)$ for some $e_0(i)$, $e_1(i) \in \mathbb{N}$ such that $e_1(i) \mid e_0(i)$.

The number $e(i) := e_0(i)$ is nothing but the exponent of $T(i)$.

Dual embedding degree

We need the additional number

$$d := \min\{i \in \mathbb{N} \text{ such that } e(i) \mid q^i - 1\}.$$

Dual embedding degree

We need the additional number

$$d := \min\{i \in \mathbb{N} \text{ such that } e(i) \mid q^i - 1\}.$$

It is logical to call it *dual embedding degree* of the curve $E/\mathbb{F}_q$ (with respect to the subgroup $\mathbb{G}$). Earlier, we considered the case $d = 1$.

Dual embedding degree

We need the additional number

$$d := \min\{i \in \mathbb{N} \text{ such that } e(i) \mid q^i - 1\}.$$

It is logical to call it *dual embedding degree* of the curve $E/\mathbb{F}_q$ (with respect to the subgroup $\mathbb{G}$). Earlier, we considered the case $d = 1$.

Moving to the field $\mathbb{F}_{q^d}$, we get into the previous context. All the results hold true, despite the fact that $\mathbb{G}(d) := e(d) \cdot E(\mathbb{F}_{q^d})$ is not a prime subgroup anymore.

Dual embedding degree

We need the additional number

$$d := \min\{i \in \mathbb{N} \text{ such that } e(i) \mid q^i - 1\}.$$

It is logical to call it *dual embedding degree* of the curve $E/\mathbb{F}_q$ (with respect to the subgroup $\mathbb{G}$). Earlier, we considered the case $d = 1$.

Moving to the field $\mathbb{F}_{q^d}$, we get into the previous context. All the results hold true, despite the fact that $\mathbb{G}(d) := e(d) \cdot E(\mathbb{F}_{q^d})$ is not a prime subgroup anymore.

A much more substantial property consists in relative primality of $\#\mathbb{G}(d)$ and $e(d)$. In other words, $E(\mathbb{F}_{q^d}) = \mathbb{G}(d) \times T(d)$.

Dual embedding degree

We need the additional number

$$d := \min\{i \in \mathbb{N} \text{ such that } e(i) \mid q^i - 1\}.$$

It is logical to call it *dual embedding degree* of the curve $E/\mathbb{F}_q$ (with respect to the subgroup $\mathbb{G}$). Earlier, we considered the case $d = 1$.

Moving to the field $\mathbb{F}_{q^d}$, we get into the previous context. All the results hold true, despite the fact that $\mathbb{G}(d) := e(d) \cdot E(\mathbb{F}_{q^d})$ is not a prime subgroup anymore.

A much more substantial property consists in relative primality of $\#\mathbb{G}(d)$ and $e(d)$. In other words, $E(\mathbb{F}_{q^d}) = \mathbb{G}(d) \times T(d)$.

Lemma

There is the simple equality $\mathbb{G} = E(\mathbb{F}_q) \cap \mathbb{G}(d)$.

Extending the new subgroup test

The subgroup $\mathbb{G}(d)$ is the kernel of the Tate pairing over $\mathbb{F}_{q^d}$. Hence, we are able to check whether $P \in \mathbb{G}(d)$ (and so $P \in \mathbb{G}$) or not, given an arbitrary point $P \in E(\mathbb{F}_q)$.

Extending the new subgroup test

The subgroup $\mathbb{G}(d)$ is the kernel of the Tate pairing over $\mathbb{F}_{q^d}$. Hence, we are able to check whether $P \in \mathbb{G}(d)$ (and so $P \in \mathbb{G}$) or not, given an arbitrary point $P \in E(\mathbb{F}_q)$.

The corresponding bit complexity amounts to $O\big(\log^2(q^d)\big)$, that is, to $O(d^2\ell^2)$. For small d (especially for $d = 2$), we can undoubtedly write $O(\ell^2)$.

Extending the new subgroup test

The subgroup $\mathbb{G}(d)$ is the kernel of the Tate pairing over $\mathbb{F}_{q^d}$. Hence, we are able to check whether $P \in \mathbb{G}(d)$ (and so $P \in \mathbb{G}$) or not, given an arbitrary point $P \in E(\mathbb{F}_q)$.

The corresponding bit complexity amounts to $O\big(\log^2(q^d)\big)$, that is, to $O(d^2\ell^2)$. For small d (especially for $d = 2$), we can undoubtedly write $O(\ell^2)$.

Nonetheless, for pairing-friendly curves the present test does not surpass the state-of-the-art tests in performance (even for $d = 1$).

Extending the new subgroup test

The subgroup $\mathbb{G}(d)$ is the kernel of the Tate pairing over $\mathbb{F}_{q^d}$. Hence, we are able to check whether $P \in \mathbb{G}(d)$ (and so $P \in \mathbb{G}$) or not, given an arbitrary point $P \in E(\mathbb{F}_q)$.

The corresponding bit complexity amounts to $O\big(\log^2(q^d)\big)$, that is, to $O(d^2\ell^2)$. For small d (especially for $d = 2$), we can undoubtedly write $O(\ell^2)$.

Nonetheless, for pairing-friendly curves the present test does not surpass the state-of-the-art tests in performance (even for $d = 1$).

The reason lies in large cofactors, which occur for today's pairing groups $\mathbb{G}_1$, $\mathbb{G}_2$.

Extending the new subgroup test

The subgroup $\mathbb{G}(d)$ is the kernel of the Tate pairing over $\mathbb{F}_{q^d}$. Hence, we are able to check whether $P \in \mathbb{G}(d)$ (and so $P \in \mathbb{G}$) or not, given an arbitrary point $P \in E(\mathbb{F}_q)$.

The corresponding bit complexity amounts to $O\big(\log^2(q^d)\big)$, that is, to $O(d^2\ell^2)$. For small d (especially for $d = 2$), we can undoubtedly write $O(\ell^2)$.

Nonetheless, for pairing-friendly curves the present test does not surpass the state-of-the-art tests in performance (even for $d = 1$).

The reason lies in large cofactors, which occur for today's pairing groups $\mathbb{G}_1$, $\mathbb{G}_2$.

Thus, despite the fact that the Tate pairing underlies the new subgroup check, it is relevant only for non-pairing-friendly curves.

Some noteworthy $\mathbb{F}_q$-curves for which $d > 1$

Let $\nu(i)$ stand for the 2-adicity of $q^i - 1$.

<table>
<tr><th>Curve</th><th>$\lceil \ell \rceil$</th><th>e_0</th><th>e_1</th><th>ν</th><th>d</th><th>$e_0(d)$</th><th>$e_1(d)$</th><th>$\nu(d)$</th></tr>
<tr><td>Curve25519</td><td>255</td><td>8</td><td rowspan="5">1</td><td>2</td><td colspan="4">?</td></tr>
<tr><td>Ed448-Goldilocks</td><td>448</td><td rowspan="4">4</td><td rowspan="4">1</td><td rowspan="4">2</td><td rowspan="4">4</td><td rowspan="4">4</td><td>225</td></tr>
<tr><td>Million dollar curve</td><td rowspan="2">256</td><td>3</td></tr>
<tr><td rowspan="2">Russian curves</td><td rowspan="2">4</td></tr>
<tr><td>512</td></tr>
</table>

Some noteworthy $\mathbb{F}_q$-curves for which $d > 1$

Let $\nu(i)$ stand for the 2-adicity of $q^i - 1$.

<table>
<tr><th>Curve</th><th>$\lceil \ell \rceil$</th><th>e_0</th><th>e_1</th><th>ν</th><th>d</th><th>$e_0(d)$</th><th>$e_1(d)$</th><th>$\nu(d)$</th></tr>
<tr><td>Curve25519</td><td>255</td><td>8</td><td rowspan="5">1</td><td>2</td><td colspan="4">?</td></tr>
<tr><td>Ed448-Goldilocks</td><td>448</td><td rowspan="4">4</td><td rowspan="4">1</td><td rowspan="4">2</td><td rowspan="4">4</td><td rowspan="4">4</td><td>225</td></tr>
<tr><td>Million dollar curve</td><td rowspan="2">256</td><td>3</td></tr>
<tr><td rowspan="2">Russian curves</td><td rowspan="2">4</td></tr>
<tr><td>512</td></tr>
</table>

For Curve25519 the speaker does not know the quantity d and hence its derivatives $e_0(d)$, $e_1(d)$, $\nu(d)$. It is not even clear whether d is finite or not.

Some noteworthy $\mathbb{F}_q$-curves for which $d > 1$

Let $\nu(i)$ stand for the 2-adicity of $q^i - 1$.

<table>
<tr><th>Curve</th><th>$\lceil \ell \rceil$</th><th>e_0</th><th>e_1</th><th>ν</th><th>d</th><th>$e_0(d)$</th><th>$e_1(d)$</th><th>$\nu(d)$</th></tr>
<tr><td>Curve25519</td><td>255</td><td>8</td><td rowspan="5">1</td><td>2</td><td colspan="4">?</td></tr>
<tr><td>Ed448-Goldilocks</td><td>448</td><td rowspan="4">4</td><td rowspan="4">1</td><td rowspan="4">2</td><td rowspan="4">4</td><td rowspan="4">4</td><td>225</td></tr>
<tr><td>Million dollar curve</td><td rowspan="2">256</td><td>3</td></tr>
<tr><td rowspan="2">Russian curves</td><td rowspan="2">4</td></tr>
<tr><td>512</td></tr>
</table>

For Curve25519 the speaker does not know the quantity d and hence its derivatives $e_0(d)$, $e_1(d)$, $\nu(d)$. It is not even clear whether d is finite or not.

Experiments show that $\nu(i)$ grows very slowly with respect to $e(i)$, which does not allow the condition $e(i) \mid q^i - 1$ to be fulfilled.

The case of Curve25519

Nevertheless, for our purpose, the exact value of d is unnecessary for the following reason.

The case of Curve25519

Nevertheless, for our purpose, the exact value of d is unnecessary for the following reason.

Recall that the k-th power residue symbol can be computed in a (sub-)quadratic bit time only for $k \leqslant 11$.

The case of Curve25519

Nevertheless, for our purpose, the exact value of d is unnecessary for the following reason.

Recall that the k-th power residue symbol can be computed in a (sub-)quadratic bit time only for $k \leqslant 11$.

At the same time, already for $i = 2$ (not to mention greater even i) it turns out that $e(2) = 16$. In turn, for an odd value i we clearly get $\nu(i) = \nu = 2$.

The case of Curve25519

Nevertheless, for our purpose, the exact value of d is unnecessary for the following reason.

Recall that the k-th power residue symbol can be computed in a (sub-)quadratic bit time only for $k \leqslant 11$.

At the same time, already for $i = 2$ (not to mention greater even i) it turns out that $e(2) = 16$. In turn, for an odd value i we clearly get $\nu(i) = \nu = 2$.

Thus, finite field extensions do not provide any advantage in the case of Curve25519.

The case of Curve25519

Nevertheless, for our purpose, the exact value of d is unnecessary for the following reason.

Recall that the k-th power residue symbol can be computed in a (sub-)quadratic bit time only for $k \leqslant 11$.

At the same time, already for $i = 2$ (not to mention greater even i) it turns out that $e(2) = 16$. In turn, for an odd value i we clearly get $\nu(i) = \nu = 2$.

Thus, finite field extensions do not provide any advantage in the case of Curve25519.

Problem

Is there a subgroup membership test for Curve25519 with bit complexity $O(\ell^2)$?

Thank you for your attention!