# Cover and Decomposition Index Calculus on Elliptic Curves made practical

## Application to a seemingly secure curve over $\mathbb{F}_{p^6}$

Antoine Joux[1] and Vanessa Vitse[2]

[1] DGA and Université de Versailles Saint-Quentin, Laboratoire PRISM, 45 avenue des États-Unis, F-78035 Versailles cedex, France
antoine.joux@m4x.org

[2] Université de Versailles Saint-Quentin, Laboratoire PRISM, 45 avenue des États-Unis, F-78035 Versailles cedex, France
vanessa.vitse@prism.uvsq.fr

**Abstract.** We present a new variant of cover and decomposition attacks on the elliptic curve discrete logarithm problem, that combines Weil descent and decomposition-based index calculus into a single discrete logarithm algorithm. This variant applies, at least theoretically, to all composite degree extension fields, and is particularly well-suited for curves defined over $\mathbb{F}_{p^6}$. We give a real-size example of discrete logarithm computations on a seemingly secure curve defined over a 130-bit degree 6 extension field.

**Key words:** elliptic curve, discrete logarithm, index calculus, Weil descent, decomposition attack

## 1 Introduction

Elliptic curves are used in cryptography to provide groups where the discrete logarithm problem is thought to be difficult. We recall that given a multiplicative finite group $G$ and two elements $g, h \in G$, the discrete logarithm problem consists in computing (when it exists) an integer $x$ such that $h = g^x$. In the case of elliptic curves, the group law is usually denoted additively and the discrete logarithm problem consists, given two points $P, Q$ to find an integer $x$ such that $Q = xP$.

More precisely, when elliptic curves are used in cryptographic applications, the discrete logarithm problem is usually considered to be as difficult as in a generic group of the same size [32]. As a consequence, for a given security level, the key size is much smaller than for other popular cryptosystems based on factorization or discrete logarithms in finite fields. The first elliptic curves considered [21, 25] in cryptography were defined over either binary or prime fields, but to speed up the arithmetic computations, it has been proposed to use various forms of extension fields. In particular, Optimal Extension Fields have been proposed in [5] to offer high performance in hardware implementations. They are of the form $\mathbb{F}_{p^d}$ where $p$ is a pseudo-Mersenne prime and $d$ is such that there exists an irreducible polynomial of the form $X^d - \omega \in \mathbb{F}_p[X]$. In most examples, the degree $d$ of the extension is rather small.

However, when curves defined over extension fields are considered, some non-generic attacks, such as Weil descent or decomposition attack, can be applied. The first one aims at transferring the DLP from $E(\mathbb{F}_{q^n})$ to the Jacobian of a curve $\mathcal{C}$ defined over $\mathbb{F}_q$ and then use index calculus on this Jacobian [2, 14, 17] to compute the logarithm; it works well when the genus of the curve $\mathcal{C}$ is small, ideally equal to $n$, but this occurs quite infrequently in practice. Many articles have studied the scope of this technique (cf. [9, 12, 13, 16, 18]), but even on vulnerable curves, the Weil descent approach is often just a little more efficient than generic attacks

on the DLP. Decomposition-based index calculus, or decomposition attack, is a more recent algorithm (see [11, 15, 20, 28]), which applies equally well to all (hyper-)elliptic curves defined over an extension field. Its asymptotic complexity is promising, but in practice, due to large hidden constants in the complexity, it becomes better than generic attacks for group sizes too large to be threatened anyway.

In this article, we combine both techniques into a cover and decomposition attack, which applies as soon as the extension degree is composite. The idea is to first transfer the DLP on the Jacobian of a curve defined on an intermediate field, then use the decomposition method on this sub-extension instead of the classical index calculus. This attack is not a mere theoretical possibility: we give concrete examples of curves defined over $\mathbb{F}_{p^6}$ that are seemingly secure against all other attack, but for which our method allows to solve the DLP in a reasonable time. In particular, we have been able to compute logarithms for an elliptic curve defined over a 130-bit degree 6 extension field in less than 30 hours real-time, using approximately 3700 CPU $\cdot$ hours.

The paper is organized as follow: first we briefly recall in Section 2 the principles of Weil descent and of the decomposition method. We then give an explicit description of our attack in Section 3, introducing a useful variant of the decomposition step that can be of independent interest. In particular, we study the case of elliptic curves defined over $\mathbb{F}_{p^6}$, list all the potentially vulnerable curves and give a complexity analysis and a comparison with previously known attacks. Finally, in Section 4, we describe in details the computations on our 130-bit example.

## 2 Survey of previous work

### 2.1 Weil descent and cover attacks

Weil descent has been first introduced in cryptography by Frey [12]; the idea is to view an abelian variety $A$ (e.g. an elliptic curve) of dimension $d$ defined over an extension field $K/k$ as an abelian variety $W_{A/k}$ of dimension $nd$ over $k$, where $n$ is the degree of the extension $K/k$. If $W_{A/k}$ turns out to be the Jacobian of a curve $\mathcal{C}_{|k}$ or can be mapped into such a Jacobian, then the discrete logarithm in $A(K)$ can be transferred to $Jac_{\mathcal{C}}(k)$, where it may become much weaker due to the existence of efficient index calculus algorithms. When the genus of $\mathcal{C}$ is small relatively to the cardinality of the base field, the complexity is in $O(g^2 \log^3 p \cdot g! \cdot p + g^2 \log p \cdot p^2)$ as the cardinality $p$ of $k$ grows to infinity; the first term comes from the relation search and the second from the sparse linear algebra. Following [17], it is possible to rebalance these two terms by using a double large prime variation. In this variant, only a small number $p^\alpha$ of *prime divisors*[3] are considered as genuine, while the rest of the prime divisors are viewed as "large primes". The optimal value of $\alpha$ depends of the cost of the two phases; asymptotically the choice that minimizes the total running time is $1 - 1/g$, yielding a complexity in $\tilde{O}(p^{2-2/g})$ for fixed $g$ as $p$ goes to infinity.

The main difficulty of this Weil descent method is to find the curve $\mathcal{C}$. This problem was first addressed for binary fields by Gaudry, Hess and Smart (GHS [16]) and further generalized by Diem [9] in odd characteristic. To attack an elliptic curve $E$ defined over $\mathbb{F}_{p^n}$ ($p$ prime power), the GHS algorithm builds a curve $\mathcal{C}$ defined over $\mathbb{F}_p$ such that there exists a cover map $\pi : \mathcal{C} \to E$ defined over $\mathbb{F}_{p^n}$. The construction is more easily explained in terms of function fields: the Frobenius automorphism $\sigma_{\mathbb{F}_{p^n}/\mathbb{F}_p}$ can be extended to the composite field $F' = \prod_{i=0}^{n-1} \mathbb{F}_{p^n}(E^{\sigma^i})$, and the function field $F = \mathbb{F}_p(\mathcal{C})$ is defined as the subfield of $F'$ fixed by $\sigma$. The GHS algorithm then uses the so-called conorm-norm map $N_{F'/F} \circ Con_{F'/\mathbb{F}_{p^n}(E)}$ to transfer the discrete logarithm from $E(\mathbb{F}_{p^n})$ to $Jac_{\mathcal{C}}(\mathbb{F}_p)$. An important condition is that the kernel of this map must not intersect the subgroup in which the discrete logarithm takes place, but as remarked in [9, 18], this is not a problem in most cryptographically interesting situations.

[3] The term *prime divisor* is an abuse of language that denotes the linear irreducible polynomials that are used in the index calculus algorithm on $Jac_{\mathcal{C}}(k)$

This technique is particularly efficient when the genus $g$ of $\mathcal{C}$ is close to $n$. However, this only occurs for a small number of curves: in general, $g$ is of the order of $2^n$, which means that index calculus in the Jacobian of $\mathcal{C}$ is much slower than generic attacks on $E(\mathbb{F}_{p^n})$. Still, it has been shown that over some finite fields, most if not all elliptic curves are weak in the sense that Weil descent algorithms are better than generic attacks; these fields have composite extension degree [24]. Indeed, even when the GHS attack does not provide any low genus cover for $E$, it may still be possible to find a sequence of low degree isogenies, a.k.a. an *isogeny walk*, from $E$ to another elliptic $E'$ vulnerable to the GHS attack [13].

In this article, we are especially interested by elliptic curves defined over finite fields of extension degree $n = 6$. In this case, we can consider the three different extensions $\mathbb{F}_{p^6}/\mathbb{F}_{p^3}$, $\mathbb{F}_{p^6}/\mathbb{F}_{p^2}$ and $\mathbb{F}_{p^6}/\mathbb{F}_p$ for the GHS attack. For the first two extensions, there exist a non-negligible proportion of weak curves, with corresponding covers by a curve of genus respectively 2 and 3 (see [27]). Whereas classical index calculus methods do not provide any complexity improvement for genus 2 coverings, the situation is slightly better for the genus 3 case: using the double large prime variation of [17], the complexity of the discrete logarithm is in $\tilde{O}\left(p^{8/3}\right)$, or even in $\tilde{O}\left(p^2\right)$ when the cover is non-hyperelliptic [10], which is smaller than the $\tilde{O}\left(p^3\right)$ of generic attacks. For the extension $\mathbb{F}_{p^6}/\mathbb{F}_p$, the genus of the cover obtained by the GHS attack cannot be as low as 6: it is at least 8 in characteristic 2, and 9 otherwise, but will be higher for most curves [7].

### 2.2 Decomposition attack

The index calculus method has become ubiquitous in the last decades for the resolution of discrete logarithm problems. However its direct application to elliptic curves faces two major challenges: contrarily to finite fields or hyperelliptic curves, there is no natural choice of factor base and besides there is no equivalent of the notion of factorization of group elements.

The first main breakthrough was achieved in 2004 by Semaev [31] when he suggested to replace factorization by decomposition into a fixed number of points; for that purpose, he introduced the summation polynomials which give an algebraic expression of the fact that a given points decomposes into a sum of factor base elements. But for a lack of an adequate factor base, this approach fails in the general case. Then Gaudry and Diem [11, 15] independently proposed to use Semaev's idea to attack curves defined over small degree extension fields $\mathbb{F}_{p^n}/\mathbb{F}_p$. Their method shares the basic outline of index calculus, but to distinguish it from what has been presented in the previous subsection, we follow [28] and call it the decomposition attack.

On $E(\mathbb{F}_{p^n})$, a convenient choice of factor base is the set of rational points of the curve having their $x$-coordinates in the base field $\mathbb{F}_p$. By combining Semaev's summation polynomials and Weil restriction, the relation search then becomes a resolution of a multivariate polynomial system over $\mathbb{F}_p$. The complexity of this approach can be estimated using double large prime variation, by $O\left(p^{2-2/n}\right)$ for fixed $n$ as $p$ grows to infinity. Moreover, this applies to all curves defined over $\mathbb{F}_{p^n}$. Unfortunately, the hidden constants in this complexity become very large as $n$ grows, and the resolution of the polynomial systems is intractable as soon as $n \geq 4$ (or $n \geq 5$ with the variant given in [20]).

The decomposition attacks can also be applied to higher genus curves. However, the Semaev's polynomials are no longer available in this case and the algebraic expression of the group law is more complicated. In [28], Nagao proposes an elegant way to circumvent this problem, using divisors and Riemann-Roch spaces. For hyperelliptic curves, the decomposition search then amounts to solving a quadratic multivariate polynomial system. This approach is less efficient than Semaev's in the elliptic case, but is the simplest otherwise. For fixed extension degree $n$ and genus $g$, the complexity of a decomposition attack is in $O\left(p^{2-2/ng}\right)$ with a double large prime variation. Again, the resolution of the polynomial system is the main technical difficulty, and is easily feasible for only very few couples $(n, g)$, namely $(2, 2)$, $(2, 3)$ and $(3, 2)$.

## 3 Cover and Decomposition attack

Let $\mathbb{F}_{q^d}/\mathbb{F}_p$ be an extension of finite fields, where $q$ is a power of $p$ (in most applications $p$ denotes a large prime but in general, it can be any prime power), and let $E$ be an elliptic curve defined over $\mathbb{F}_{q^d}$ of cryptographic interest, i.e. containing a subgroup $G$ of large prime order. As $E$ is defined over an extension field, it is subject to the attacks presented above. But if the degree $[\mathbb{F}_{q^d} : \mathbb{F}_p]$ of the extension is larger than 5, then we have seen that $E$ is practically immune to decomposition attacks. In the following, we assume that the potential reduction provided by the GHS attack or its variants is not significant enough to threaten the security of the DLP on the chosen curve $E$.

When $q$ is a strict power of $p$, we have a tower of extensions given by $\mathbb{F}_{q^d}/\mathbb{F}_q$ and $\mathbb{F}_q/\mathbb{F}_p$. In this context, it becomes possible to combine both cover and decomposition methods and obtain an efficient attack of the DLP on $E$. The idea is to use Weil descent on the first extension $\mathbb{F}_{q^d}/\mathbb{F}_q$ to get a cover defined over $\mathbb{F}_q$, with small enough[4] genus $g$. Then we can apply a decomposition attack on the curve thus obtained, making use of the second extension $\mathbb{F}_q/\mathbb{F}_p$.

### 3.1 Description of the attack

We now explicitly detail this cover and decomposition approach. We suppose first that there exists a hyperelliptic curve $H$ of small genus $g > 1$, defined over $\mathbb{F}_q$ together with a covering map $\pi : H \to E$ defined over $\mathbb{F}_{q^d}$. This can be obtained by the GHS attack or its variants, possibly preceded by an isogeny walk. This cover classically allows to transfer the DLP from $G$ to a subgroup $G' \subset Jac_H(\mathbb{F}_q)$ via the conorm-norm map

$$N_{\mathbb{F}_{q^d}/\mathbb{F}_q} \circ \pi^* : E(\mathbb{F}_{q^d}) \simeq Jac_E(\mathbb{F}_{q^d}) \to Jac_H(\mathbb{F}_q)$$

assuming that $\ker(N_{\mathbb{F}_{q^d}/\mathbb{F}_q} \circ \pi^*) \cap G = \{\mathcal{O}_E\}$. For simplicity, we assume that $H$ has an imaginary model given by the equation $y^2 = h(x)$.

The decomposition part of the attack is adapted from Gaudry and Nagao; since it is quite recent, we detail the method. We consider the same factor base as Gaudry and Nagao

$$\mathcal{F} = \{D_Q \in Jac_H(\mathbb{F}_q) \ : \ D_Q \sim (Q) - (\mathcal{O}_H), Q \in H(\mathbb{F}_q), x(Q) \in \mathbb{F}_p\},$$

which contains approximately $p$ elements; as usual, we can use the hyperelliptic involution to reduce the size of the factor base by a factor 2. As in all index calculus based approaches, there are two time consuming steps: in the first one, we have to collect about $p/2$ relations between factor base elements, while in the second one, we compute discrete logarithms by using linear algebra on the matrix of relations.

Let $n$ be the extension degree $[\mathbb{F}_q : \mathbb{F}_p]$. In Nagao's original version, one tries to decompose an arbitrary divisor $D$ (typically obtained by considering a large multiple of some element in $\mathcal{F}$) into a sum of $ng$ divisors in the factor base

$$D \sim \sum_{i=1}^{ng} ((Q_i) - (\mathcal{O}_H)) . \tag{1}$$

Heuristically, there exist approximately $p^{ng}/(ng)!$ distinct sums of $ng$ elements of $\mathcal{F}$, so the probability that a given divisor $D$ is decomposable can be estimated by $1/(ng)!$. To check if $D$ can be decomposed, one considers the Riemann-Roch $\mathbb{F}_q$-vector space

$$\mathcal{L}\left(ng(\mathcal{O}_H) - D\right) = \{f \in \mathbb{F}_q(H)^* : div(f) \geq D - ng(\mathcal{O}_H)\} \cup \{0\}.$$

[4] Meaning that $g$ should be small relatively to the genus that could be obtained by direct Weil descent, using the extension $\mathbb{F}_{q^d}/\mathbb{F}_p$.

We can assume that the divisor $D$ is reduced and has Mumford representation $(u(x), v(x))$ with $\deg u = g$, so that this $\mathbb{F}_q$-vector space is spanned by $u(x), u(x)x, \ldots, u(x)x^{m_1}, (y - v(x)), x(y - v(x)), \ldots, x^{m_2}(y - v(x))$, where $m_1 = \lfloor (n-1)g/2 \rfloor$ and $m_2 = \lfloor (ng-g-1)/2 \rfloor$. A function $f = \lambda_0 u(x) + \lambda_1 u(x)x + \ldots + \lambda_{m_1} u(x)x^{m_1} + \mu_0(y - v(x)) + \mu_1 x(y - v(x)) + \ldots + \mu_{m_2} x^{m_2}(y - v(x))$ vanishes on the support of $D$ and exactly $ng$ other points (possibly defined on the algebraic closure of $\mathbb{F}_q$) if its top-degree coefficient is not zero. We are looking for a condition on $\lambda_0, \ldots, \lambda_{m_1}, \mu_0, \ldots, \mu_{m_2} \in \mathbb{F}_q$ such that the zeroes $Q_1 \ldots, Q_{ng}$ of $f$ disjoint from $Supp(D)$ have $x$-coordinate in $\mathbb{F}_p$; this event yields a relation as in (1).

Therefore we consider the polynomial $F(x) = f(x, y)f(x, -y)/u(x)$ where $y^2$ has been replaced by $h(x)$. Without loss of generality, we can fix either $\lambda_{m_1} = 1$ or $\mu_{m_2} = 1$ in order to have $F$ monic of degree $ng$. The roots of $F$ are exactly the $x$-coordinates of the zeroes of $f$ distinct from $Supp(D)$, thus we are looking for the values of $\lambda$ and $\mu$ for which $F$ splits in linear factors over $\mathbb{F}_p$. A first necessary condition is that all of its coefficients, which are quadratic polynomials in $\lambda$ and $\mu$, belong to $\mathbb{F}_p$; a Weil restriction on these coefficients then yields a quadratic polynomial system of $(n-1)ng$ equations and variables coming from the components of the variables $\lambda$ and $\mu$. The corresponding ideal is generically of dimension 0, and the solutions of the system can be found using for instance a Gröbner basis computation. Since the number of systems to solve is huge (on average $(ng)! \cdot p/2$), techniques such as the F4-traces algorithm of [19] should be preferred. Once the solutions in $\mathbb{F}_q$ are obtained, it remains to check that the resulting polynomial $F$ splits in $\mathbb{F}_p[x]$, and if it is the case, to compute the corresponding decomposition of $D$.

In this article, instead of using Nagao's original version, we consider a somewhat different approach that offers some similarity with the sieving method used in the number field and function field sieves [1, 23]. More precisely, we no longer have a divisor $D$ to decompose, but instead search for sums of factor base elements equal to 0:

$$\sum_{i=1}^{m} ((Q_i) - (\mathcal{O}_H)) \sim 0 \tag{2}$$

The expected number of relations of the form (2) involving $m$ points of the factor base is approximately $\frac{p^{m-ng}}{m!}$. Since we need to collect at least about $p/2$ relations, we look for sums of $m = ng + 2$ points. As in Nagao's method, we work with the Riemann-Roch $\mathbb{F}_q$-vector space $\mathcal{L}(-m(\mathcal{O}_H))$ which is spanned by $1, x, \ldots, x^{m_1}$, $y, xy, \ldots, x^{m_2}y$, where $m_1 = \lfloor m/2 \rfloor$ and $m_2 = \lfloor (m-1)/2 \rfloor \rfloor - g$. We consider the function $f = \lambda_0 + \lambda_1 x + \ldots + \lambda_{m_1} x^{m_1} + \mu_0 y + \mu_1 xy + \ldots + \mu_{m_2} x^{m_2} y$: it vanishes in exactly $m$ points if its top-degree coefficient is not zero, and the $x$-coordinates of its zeroes are the roots of the polynomial

$$F(x) = f(x, y)f(x, -y) = (\lambda_0 + \lambda_1 x + \ldots + \lambda_{m_1} x^{m_1})^2 - h(x)(\mu_0 + \mu_1 x + \ldots + \mu_{m_2} x^{m_2})^2.$$

As before, we fix $\lambda_{m_1} = 1$ if $m$ is even or $\mu_{m_2} = 1$ otherwise, so that $F$ is monic, and in order to obtain a relation of the form (2), we are looking for values of $\lambda$ and $\mu$ for which $F$ splits in linear factors over $\mathbb{F}_p$. The first condition is that $F$ belongs to $\mathbb{F}_p[x]$; after a Weil restriction on its coefficients, this translates as a quadratic polynomial system of $(n-1)m$ equations and $n(m-g)$ variables. With our choice of $m = ng + 2$, this corresponds to an underdetermined system of $n(n-1)g + 2n - 2$ equations in $n(n-1)g + 2n$ variables.

When the parameters $n$ and $g$ are not too large, we remark that it is possible to compute once for all the corresponding Gröbner basis for a lexicographic order. Each specialization of the last two variables then provides an easy to solve system, namely triangular with low degree. It remains to check whether the corresponding expression of $F$ is indeed split and to deduce the corresponding relations between the points of $\mathcal{F}$. Note that this kind of precomputation cannot be achieved in Nagao's version: it would require to solve the corresponding polynomial system with the representation $(u(x), v(x))$ of $D$ as formal parameters, which is intractable due to a too large number of variables.

In this form, our technique is already faster than Nagao's (timings are given in Section 5), but can still be further improved. Indeed, checking that $F$ is split has a non-negligible cost, since we need to factor a polynomial of degree $m$ into linear terms. To avoid this cost, it is possible to modify the search for relations in order to use a sieving technique; we defer the details to Section 4.2.

Once enough relations of the form (2) have been collected, we can deduce with linear algebra the logarithms of all elements in $\mathcal{F}$ (up to a multiplicative constant, since we have not specified the base). In order to compute the discrete logarithm of an arbitrary divisor $D$, we proceed to a descent phase: we need to decompose this arbitrary divisor as a sum of factor base elements. This decomposition search can be done using Nagao's method as described above. Note that, if $D$ does not decompose as a sum, it suffices to try small multiples $2D$, $3D\ldots$ until we find one correct decomposition. Thanks to this descent step, it is possible to compute many discrete logarithms in the same group for negligible additional cost.

When the cover of $E$ is not hyperelliptic, one can still use Nagao's Riemann-Roch based approach (or the variant we have presented). It is not difficult to compute a basis of the vector spaces $\mathcal{L}(D - ng(\mathcal{O}_H))$ or $\mathcal{L}(-m(\mathcal{O}_H))$ and to consider a function $f(x, y)$ (depending of parameters $\lambda$ and $\mu$) in these spaces. Getting rid of the $y$-variable can be done quite easily by computing the resultant in $y$ of $f$ and the equation of the curve; however, the resulting polynomial $F(x)$ no longer depends quadratically of the parameters $\lambda$ and $\mu$. Consequently, the system obtained by Weil restriction still has the same number of equations and variables but its degree is greater than 2, so that the resolution is more complicated.

### 3.2 Complexity analysis

Constructing the cover $H_{|\mathbb{F}_q}$ of an elliptic curve $E_{|\mathbb{F}_{q^d}}$ with the GHS method and transferring the DLP from $G \subset E(\mathbb{F}_{q^d})$ to $G' \subset Jac_H(\mathbb{F}_q)$ has essentially a unit cost, which is negligible compared to the rest of the attack. The complexity of the decomposition phase is divided between the relation search and the linear algebra steps. For the classical approach of Nagao, in order to collect about $p/2$ relations we need to solve on average $(ng)!\cdot p/2$ quadratic polynomial systems. The resolution cost of this kind of systems using e.g. Gröbner basis is hard to estimate precisely, but is at least polynomial in the degree $2^{(n-1)ng}$ of the corresponding zero-dimensional ideal. The linear algebra step then costs $O(ngp^2)$ operations in $\mathbb{Z}/(\#G)\mathbb{Z}$ using sparse linear algebra techniques. With the variant we have presented, we need to compute first the lexicographic order Gröbner basis of an ideal generated by $n(n-1)g + 2n - 2$ quadratic equations in $n(n-1)g + 2n$ variables. This cost is also at least exponential in $n^2g$, but the Gröbner basis computation has to be done only once. Afterwards, we have to solve on average $(ng+2)! \cdot p/2$ "easy" systems. The complexity of the linear algebra step is the same (the cost of the descent is negligible compared to the sieving phase).

When $p$ is very large relatively to $n$ and $g$, then the linear algebra becomes the dominating step of the algorithm. It is nevertheless possible to rebalance the cost of the two steps. Indeed, collecting extra relations can speed up the computation of the logarithms; this is the idea behind structured Gaussian elimination [22] and double large prime variation. With the former technique, the consequence on the asymptotic complexity is not known. With the latter, the analysis of [17] shows that the asymptotic complexity of this cover and decomposition attack becomes $\tilde{O}(p^{2-2/ng})$ as $p$ grows to infinity for fixed $n$ and $g$ in Nagao's version. For the variant, the asymptotic complexity is higher: $\tilde{O}(p^{2-2/(ng+2)})$ but with a much smaller hidden constant, so that it is faster than Nagao's version for accessible values of $p$. Note that it is straightforward to parallelize the relation search phase; this is also possible, but much less efficiently, for the linear algebra step. This implies that the practical choice of the optimal balance depends not only of the implementation but also of the computing power available.

These complexities crucially depend of the genus of the cover of $E$. If it is too large, generic attacks become more efficient, but it may be possible to transfer the DLP from $E$ to an isogenous curve $E'$ more vulnerable to this cover and decomposition attack. There exist two "isogeny walk" strategies to find the curve $E'$ (if it exists) [6]: one can randomly sample the isogeny class of $E$ via low-degree isogenies until a weak curve is found, or one can try all the weak curves until a curve isogenous to $E$ is found. The best strategy to use and its complexity depend of the size of the isogeny class and of the number of weak curves. For the cases we have considered, this isogeny walk can become the dominating part in the overall complexity (see next section for details).

## 4 Application to elliptic curves defined over $\mathbb{F}_{p^6}$

For an elliptic curve $E$ defined over an extension field $\mathbb{F}_{p^6}$ (where $p$ is a prime power), we can apply our cover and decomposition attack either with the tower $\mathbb{F}_{p^6} - \mathbb{F}_{p^2} - \mathbb{F}_p$ or with the tower $\mathbb{F}_{p^6} - \mathbb{F}_{p^3} - \mathbb{F}_p$. We have seen in Section 2.2 that in practice, we can compute decompositions only for a very limited number of values of $(n, g)$. In particular, our attack is feasible only if $E$ admits a genus 3 (resp. 2) cover; we give examples of such curves below. Of course, this attack needs to be compared with the classic cover attacks or decomposition attacks using the base field $\mathbb{F}_{p^3}, \mathbb{F}_{p^2}$ or $\mathbb{F}_p$, as recalled in Section 2.

### 4.1 Using a genus 3 cover

In the present subsection, we consider the cover and decomposition attack using the tower $\mathbb{F}_{p^6} - \mathbb{F}_{p^2} - \mathbb{F}_p$. Thanks to the results of [9, 27, 34], in odd characteristic, we know that the only elliptic curves defined over $\mathbb{F}_{q^3}$ (in our case, we have $q = p^2$) for which the GHS attack yields a cover by a hyperelliptic curve $H$ of genus 3 defined over $\mathbb{F}_q$, are of the form

$$y^2 = h(x)(x - \alpha)(x - \sigma(\alpha)) \tag{3}$$

where $\sigma$ is the Frobenius automorphism of $\mathbb{F}_{q^3}/\mathbb{F}_q$, $\alpha \in \mathbb{F}_{q^3} \setminus \mathbb{F}_q$ and $h \in \mathbb{F}_q[x]$ of degree 1 or 2. Similar results are also available in characteristic 2 (see [29]), thus our attack is also applicable in characteristic 2; we give details of the construction of the cover in both cases in the Appendix. The number of curves admitting an equation of the form (3) is $\Theta(q^2)$, thus only a small proportion of curves is directly vulnerable to the cover and decomposition attack. However, since this number of weak curves is much larger than the number of isogeny classes (which is about $q^{3/2}$), a rough reasoning would conclude that essentially all curves should be insecure using an isogeny walk strategy. Assuming that the probability for a curve to be weak is independent from its isogeny class, we obtain that the average length of this isogeny walk is about $q = p^2$ steps. It is thus the dominating phase of the algorithm, but is still better than the $\tilde{O}(p^3)$-generic attacks. Nevertheless, all the curves of the form (3) have a cardinality divisible by 4, so obviously not all curves are vulnerable to this isogeny walk (we recall that two curves are isogenous if and only if they have the same cardinality). Still, we conjecture that all curves with cardinality divisible by 4 are vulnerable to this cover and decomposition attack using an isogeny walk.

We can also consider non-hyperelliptic genus 3 covers. In this case, weak curves have equation

$$y^2 = c(x - \alpha)(x - \sigma(\alpha))(x - \beta)(x - \sigma(\beta)) \tag{4}$$

where $c \in \mathbb{F}_{q^3}$ and either $\alpha, \beta \in \mathbb{F}_{q^3} \setminus \mathbb{F}_q$ or $\alpha \in \mathbb{F}_{q^6} \setminus \left(\mathbb{F}_{q^2} \cup \mathbb{F}_{q^3}\right)$ and $\beta = \sigma^3(\alpha)$. Much more curves are directly vulnerable to this cover [27]: actually, about half of the curves having their full 2-torsion defined over $\mathbb{F}_q$ admit an equation of the form (4).

For a genus 3 hyperelliptic cover over $\mathbb{F}_{p^2}$, the quadratic polynomial systems to solve over $\mathbb{F}_p$ are composed of 6 variables and 6 equations with Nagao's approach, or 8 equations and 10 variables with our variant. Such systems can be solved very quickly by any computational algebra system. Unfortunately, with non-hyperelliptic covers, the systems of equations are much more complicated; in particular, we have not been able to successfully compute decompositions using available Gröbner basis implementations.

### 4.2 Details of the sieving technique

As mentioned in Section 3, it is possible to speed up our relation search by using a sieving technique. We describe this technique in the special case of a hyperelliptic genus 3 cover in odd characteristic. We have seen

that with our variant, we obtain a system of 8 equations and 10 variables, together with the corresponding lexicographic order Gröbner basis, which is computed once for all. The standard approach would be to evaluate the last two variables $x_9$ and $x_{10}$, solve the resulting easy (low degree and triangular) system and test if the polynomial $F$ of degree 8 (defined in Section 3.1) is split for these solutions.

Instead, we start by evaluating the last variable $x_{10}$. Due to the simple shape of the Gröbner basis in this particular example, it is then possible to express all the other variables as polynomials in $x_9^2$. Thus, after replacement, $F$ becomes a polynomial in the two variables $x$ and $y = x_9^2$, with a degree in $y$ equal to 2. The key idea is, instead of trying to factor $F$ for many values of $x_9$, to compute for each value of $x \in \mathbb{F}_p$ the values of $y$ such that $F(x, y) = 0$. Since $F$ has degree 2 in $y$, this can be done very efficiently, just by obtaining the square root of the discriminant. In fact, we can speed the process even more by tabulating the square roots modulo $p$. Our sieving process consists, for each root $y$, to increment a counter corresponding to this value of $y$. When one of this counters reaches 8, the corresponding value of $y$ admits 8 corresponding roots. If in addition $y$ is a quadratic residue, then for $x_9 = \pm\sqrt{y}$, the polynomial $F$ evaluated at $x_9$ splits into 8 distinct linear terms. Note that changing the sign of $x_9$ yields the same relation.

Moreover, this technique is well-suited to the double large prime variation. Indeed, if $\mathcal{F}'$ denotes the subset of the factor base $\mathcal{F}$ (given in Section 3.1) composed of a small number $p^\alpha$ of "genuine" points, we just have to substitute into $F$ the values of $x$ corresponding to $x$-coordinates of points in $\mathcal{F}'$. As soon as 6 values of $x$ are associated to one value of $y$, we obtain a relation involving at most 2 large primes (if the remaining degree 2 factor is split, which occurs with probability close to $1/2$). This speeds up the relation search by a factor $p^{1-\alpha}$. In particular, this modifies the optimal balance between the relation search and the linear algebra for our variant: the asymptotically best choice for $\alpha$ is now $1 - 1/7$ and the overall complexity is reduced from $\tilde{O}(p^{2-2/8})$ to $\tilde{O}(p^{2-2/7})$ as $p$ grows to infinity. This reduces the asymptotic gap between Nagao's method and our variant, without degrading the practical performance.

### 4.3 Using a genus 2 cover

We now consider the tower $\mathbb{F}_{p^6} - \mathbb{F}_{p^3} - \mathbb{F}_p$. The existence of genus 2 covers (which are necessarily hyperelliptic) defined over $\mathbb{F}_q$, where $q = p^3$, has been studied in [30, 3]. In odd characteristic, vulnerable curves admit an equation in so-called Scholten form

$$y^2 = ax^3 + bx^2 + \sigma(b)x + \sigma(a) \tag{5}$$

where $a, b \in \mathbb{F}_{q^2}$ and $\sigma$ is the Frobenius automorphism of $\mathbb{F}_{q^2}/\mathbb{F}_q$. An elliptic curve $E$ can be transformed into Scholten form as soon as its full 2-torsion is defined over $\mathbb{F}_{q^2}$ [30] or its cardinality is odd and $j(E) \notin \mathbb{F}_q$ [3]. Consequently, a large proportion of curves are vulnerable to our cover and decomposition attack. Moreover, any curve without full 2-torsion but still with a cardinality divisible by 4, is 2-isogenous to a curve with full 2-torsion [26].

In this setting, the quadratic polynomial systems to solve over $\mathbb{F}_p$ are composed of 12 variables and 12 equations with Nagao's approach, or 16 equations and 18 variables with our variant. Solving such systems is still feasible on current personal computers, but is much slower than in the case of hyperelliptic genus 3 cover defined over $\mathbb{F}_{p^2}$. For example with Nagao's approach, the degree of the corresponding zero-dimensional ideals are respectively $2^{12}$ versus $2^6$. Since the complexity of standard Gröbner basis algorithms for zero-dimensional ideals is bounded from below by the cube of the degree, this means that using a genus 2 cover over $\mathbb{F}_{p^3}$ for our attack becomes at least 250 000 times slower than when it uses a genus 3 cover over $\mathbb{F}_{p^2}$.

### 4.4 Complexity and comparison with other attacks

In order to obtain actual (and not just asymptotic) comparisons, we consider the cryptographically significant example of a curve $E$ defined over $\mathbb{F}_{p^6}$ where $p$ is a prime close to $2^{27}$ and $\#E(\mathbb{F}_{p^6})$ is 4 times a 160-bit prime

number $\ell$. Following [33], we consider that the cost of an arithmetic operation in $\mathbb{F}_q$ (where $q = p, p^2, p^3, p^6$ or $\ell$) is given by $c_q = (\log q)^2$, that the cost of an operation on polynomials (using Karatsuba method) of small degree $g$ over $\mathbb{F}_q$ is $c_{q,g} = g^{1.59} c_q$, and that the cost of an operation in a Jacobian is $c_J = 22 c_{q,g}$ when $g > 3$, $c_J = 10 c_q$ when $g = 1$, $c_J = 40 c_q$ when $g = 2$ and $c_J = 90 c_q$ when $g = 3$.

The basis of comparison for all attacks on the elliptic curve discrete logarithm problem comes from generic algorithms, i.e. algorithms that do not use any information about the actual group structure and only consider the group law. Such attacks, e.g. Shanks's baby-step giant-step or Pollard's Rho, have a complexity in $O(\sqrt{\#G})$ group operations. In our elliptic curve setting, the cardinality of $G$ is up to a small factor equal to the cardinality of $E(\mathbb{F}_{p^6})$, so that generic attacks perform in $\tilde{O}(p^3)$. On the example, Pollard's Rho method necessitates about $2\sqrt{\ell} = 2^{81}$ group operations in the curve, namely a total cost of $2^{99}$.

A second attack we consider is the double large prime variation of the classic index calculus on a hyperelliptic genus 3 cover $H_{|\mathbb{F}_{p^2}}$ (if it exists); we have seen that the asymptotic complexity is in $\tilde{O}(p^{8/3})$ as $p$ grows to infinity. For the complexity of the 162-bit example, we need to find the optimal balance. We consider a reduced factor base composed of $(p^2)^\alpha/2$ genuine primes. The probability that an element of the Jacobian gives a relation with at most two large primes is about $p^{2(\alpha-1)}/2$, and as we need approximately $p^2/2$ such relations, the complexity of the relation search phase is $p^{2(2-\alpha)} c_J = 2^{126-54\alpha}$. The linear algebra costs $3(p^{2\alpha}/2)^2 c_\ell = 2^{108\alpha+14.2}$, so that the optimal value of $\alpha$ is 0.69 and the total cost of the index calculus attack is about $2^{90}$.

Alternatively, one can apply the GHS attack with the extension $\mathbb{F}_{p^6}/\mathbb{F}_p$. Its asymptotic complexity is in $\tilde{O}(p^{2-2/g})$, but the actual behavior depends greatly of the genus $g$ of the cover over $\mathbb{F}_p$. We have seen that in the best case $g = 9$. With our 162-bit example, this gives a complexity of $9! p/2\, c_J \simeq 2^{63}$ for the relation search and $9(p/2)^2 c_\ell \simeq 2^{70}$ for the linear algebra, which can be slightly rebalance by considering a factor base reduced by a one half factor to obtain an overall cost in $2^{69}$. However, this genus 9 case is very exceptional, and in general the genus of the cover given by the GHS attack is much larger. For instance, if we consider the equation of the form (3), which is already a favorable case for the GHS attack, the genus of the cover is $g = 33$, and the complexity of the index calculus becomes much worse than generic attacks.

Instead of cover techniques, we can consider the decomposition attack of [11, 15]. If we consider $\mathbb{F}_{p^2}$ as the base field, the asymptotic complexity is in $\tilde{O}(p^{8/3})$. The actual value involves the cost of the resolution of a multivariate polynomial system in 3 equations and variables of degree 4 over $\mathbb{F}_{p^2}$. Since the corresponding ideal is zero-dimensional, we estimate that this resolution necessitates $D^3$ operations in $\mathbb{F}_{p^2}$, where $D = 4^3$ is the degree of the ideal. In the 162-bit example, we find that the best value of $\alpha$ for the balance between the relation search and the linear algebra is 0.76, which gives a total cost of $2^{97}$. Alternatively, if we take $\mathbb{F}_p$ as the base field, the asymptotic complexity becomes $\tilde{O}(p^{5/3})$. The system to solve at each decomposition trial has 6 equations in 6 variables of degree 32, and the corresponding zero-dimensional ideal has degree $D = 32^6$. The cost of the resolution of such system is prohibitively high, in $2^{100}$, which is clearly not competitive on this example.

Finally, we consider our cover and decomposition attack, whose asymptotic complexity either in $\tilde{O}(p^{5/3})$ with Nagao's decomposition, or $\tilde{O}(p^{12/7})$ with our sieving variant. With the first method, the actual complexity involves the resolution cost of a quadratic multivariate polynomial system. For such systems, we use the different bound $\binom{n+d_{reg}}{n}^3 c_p$ on the complexity of Gröbner basis computations, where $n$ is the number of variables and $d_{reg}$ the degree of regularity, see [20]. If we work with a hyperelliptic genus 3 cover defined over $\mathbb{F}_{p^2}$, then $n = 6$ and $d_{reg} = 7$, so the complexity of the relation search with Nagao's decomposition is $6!\, p/2 \binom{13}{6}^3 c_p \simeq 2^{77}$. This is greater than the complexity of the linear algebra, which is in $6\,(p/2)^2 c_\ell \simeq 2^{68}$, so no balance is needed. If we work instead with a genus 2 cover defined over $\mathbb{F}_{p^3}$, the cost of the linear

algebra is the same but the relation phase is much slower: with $n = 12$ and $d_{reg} = 13$, its complexity becomes about $2^{112}$.

With our variant instead of Nagao's, the Gröbner basis computation is done only once, and the specialized systems we have to solve afterwards are much simpler. In the case of the hyperelliptic genus 3 cover, we can use the sieving technique presented above: at each step, we just have to evaluate the polynomial $F(x, y)$ in $x$ (which costs at most $16c_p$ since $\deg_x F = 8$) and solve the resulting quadratic univariate polynomial in $y$ (this amounts to the computation of a square root, which costs at most $2\log(p)c_p$). The complexity of the relation search is then $8!p/2\,(16 + 2\log p)c_p \simeq 2^{57}$, whereas the complexity of the linear algebra is in $8(p/2)^2 c_\ell \simeq 2^{69}$. This can be rebalanced, and we find that the optimal size of the reduced factor base is $0.173p$, giving a total cost of about $2^{66}$.

| Attack | Asymptotic complexity | 162-bit example cost |
|---|---|---|
| Pollard | $p^3$ | $2^{99}$ |
| Ind. calc. on $H_{\|\mathbb{F}_{p^2}}$, $g(H) = 3$ | $p^{8/3}$ | $2^{90}$ |
| Ind. calc. on $H_{\|\mathbb{F}_p}$, $g(H) = 9$ | $p^{16/9}$ | $2^{69}$ |
| Decomp. on $E_{\|\mathbb{F}_{(p^2)^3}}$ | $p^{8/3}$ | $2^{97}$ |
| Decomp. on $E_{\|\mathbb{F}_{p^6}}$ | $p^{5/3}$ | $2^{135}$ |
| Decomp. on $H_{\|\mathbb{F}_{p^2}}$, $g(H) = 3$ | $p^{5/3}$ | $2^{66}$ |

**Fig. 1.** Comparison of the complexity of various attacks on $E(\mathbb{F}_{p^6})$

## 5 A 130-bit example

In this section, we give a practical example of the genus 3 cover and decomposition attack for an elliptic curve defined over $\mathbb{F}_{p^6}$ where $p = 4\,194\,319 = 2^{22} + 15$ is a 23-bit prime. We define $\mathbb{F}_{p^2}$ as $\mathbb{F}_p[i]$ where $i^2 = -1$ and $\mathbb{F}_{p^6}$ as $\mathbb{F}_{p^2}[\theta]$ where $\theta^3 = 2$.

The elliptic curve $E$ is given by the following Weierstrass equation:

$$y^2 = (x - c)(x - \alpha)(x - \sigma(\alpha))$$

where $\sigma : x \mapsto x^{p^2}$, $c = 1\,048\,587$ and $\alpha = 3\,812\,894\,\theta^2 + 3\,527\,164\,\theta + 1\,048\,580\,i$.

This elliptic curve has a genus 3 cover by the hyperelliptic curve $H$ defined over $\mathbb{F}_{p^2}$ by:

$$\begin{aligned} y^2 = x^7 &+ (1\,048\,579\,i + 4\,194\,290)x^6 + (2\,097\,203\,i + 2\,359\,305)x^5 + (2\,686\,984\,i + 393\,267)x^4 + \\ &(3\,538\,925\,i + 1\,359\,881)x^3 + (126\,973\,i + 2\,424\,826)x^2 + (589\,830\,i + 3\,083\,272)x + \\ &4\,021\,007\,i + 1\,363\,461 \end{aligned}$$

which is of the form $y^2 = \left(x + \phi(x) + \phi^\sigma(x) + \phi^{\sigma^2}(x) - 4c\right) N(x)^2$ where $N(x)$ is the minimal polynomial of $\alpha$ over $\mathbb{F}_{p^2}$ and $\phi : x \mapsto \frac{(\alpha - \sigma^2(\alpha))(\sigma(\alpha) - \sigma^2(\alpha))}{x - \sigma^2(\alpha)} + \sigma^2(\alpha)$.

The cover map $\pi$ from $H$ to $E$ is given by:

$$\pi(x, y) = \left( \frac{x + \phi(x) + \phi^\sigma(x) + \phi^{\sigma^2}(x)}{4}, \frac{y(x - \phi^\sigma(x))(x - \phi^{\sigma^2}(x))}{8N(x)(x - \sigma^2(\alpha))} \right).$$

The common cardinality of $E$ over $\mathbb{F}_{p^6}$ and of the Jacobian of $H$ over $\mathbb{F}_{p^2}$ is:

$$N = 4\ell = 4 \cdot 136115867461471233446652598568206220160 1,$$

where $\ell$ is a 131-bit prime. The number of different abscissae $x$ in $\mathbb{F}_p$ that correspond to a pair of points $(x, y)$ and $(x, -y)$ is equal to 2 096 834. However, for technical reasons during the sieving process, we had to remove the two points with abscissa equal to zero. Still, we are doing index calculus with a smoothness basis containing approximately 2.1 millions elements.

For best performances, we use the sieving approach described in Section 3.1. As a first step, we compute a lexicographic order Gröbner basis of the system composed of 10 quadratic equations in 8 variables. This is done in about 1 min on a 2.6 GHz Intel Core 2 Duo processor with Magma V2.16-12 [8]. Instead of the double large prime variation, we execute a structured Gaussian elimination. During the sieving phase, we used 200 cores of quadri-core Intel Xeon 5570 processors at 2.93 GHz[5]. After 3751 sec, we had collected 51 883 659 relations, close to 25 times the number of necessary relations for the linear algebra. Each relation involved 8 distinct elements from the smoothness basis. Thanks to the large number of extra relations, structured Gaussian elimination performed quite well and, after 1357 seconds on a single core, it produces a system of 666 062 equations in 665 061 unknowns, involving between 8 and 62 basis elements. The total number of non-zero entries in all the equations is 33 761 662 and all these entries are equal to $\pm 1$.

The most time consuming step is the iterative linear algebra, which is done with a MPI implementation of the Lanczos algorithm. It took about 27 hours and 16 minutes on 128 cores of the same Intel Xeon processors as above. A large fraction of this time was taken by the MPI communications, since at each round 42.5 Mbytes of data had to be broadcast between the 16 involved bi-processor machine (8 cores/machine). This linear algebra phase produced discrete logarithms for all the smoothness basis elements that remained after structured Gaussian elimination. Substituting these values back in the initial linear system, we recovered, in less than 10 minutes on a single core, the discrete logarithms modulo $\ell$ of all elements in the smoothness basis (given by their coordinates on $H$):

$$\begin{aligned}
\log(1, 1\,778\,117 + 4\,043\,006\,i) &= 47810632712543597011455056269164844 1691\\
\log(2, 2\,470\,708 + 2\,816\,377\,i) &= 60274613536196417229379928410886682 6746\\
\log(3, 2\,962\,826 + 1\,627\,410\,i) &= 70530820889464725509484952408111454 0246\\
&\vdots\\
\log(4\,194\,313, 3\,987\,487 + 990\,581\,i) &= 77168988270700157762936609474336346 2187\\
\log(4\,194\,316, 2\,427\,954 + 2\,537\,863\,i) &= 13535723186646887259684604165458160 94564\\
\log(4\,194\,317, 1\,149\,909 + 103\,530\,i) &= 29756031028093138340311206649817815 5928
\end{aligned}$$

*Individual logarithms of points on the curve.* With the results of the above precomputation, computing logarithm of arbitrary points on the elliptic curve becomes easy. To demonstrate this, we constructed points on $E$ with the following process and computed their logarithms. First, we let:

$$\begin{aligned}
X_0 &= \sum_{j=0}^{5} \left( \lfloor \pi \cdot p^j \rceil \bmod p \right) i^{j \bmod 2} \theta^{j \bmod 3}\\
&= (593\,885 + 3\,175\,989\,i) + (199\,943 + 841\,508\,i)\theta + (411\,724 + 2\,224\,599\,i)\theta^2.
\end{aligned}$$

We then constructed points on $E$ with abscissa $X_0 + \delta$ for small offsets $\delta$. Let $P_1$, $P_2$, $P_3$, $P_4$, $P_5$, $P_6$ and $P_7$ be the points corresponding to offsets 0, 1, 2, 3, 5, 11 and 12. We lift each of these points to the Jacobian

[5] This work was granted access to the HPC resources of CCRT under the allocation 2010-t201006445 made by GENCI (Grand Equipement National de Calcul Intensif)

of $H$ using the Conorm-Norm method. Note that, if a given point cannot be lifted, we instead lift a small multiple of it. After that, we apply the method of Nagao (as in Section 3.1) to small multiples of the lifted element until we find a multiple that decomposes as a sum of elements from the smoothness basis. Looking up the corresponding logarithms (and dividing back by the small multiples that have been included) yields the logarithm of each point. The computation of the Conorm-Norm takes negligible time in magma. On average, we expect to try 720 multiples with Nagao's method before finding a decomposition. To try 2000 multiples, which was enough for each of the seven considered points, we need 350 seconds using magma on an Intel Core i7 at 2.66 GHz. As a consequence, each individual logarithm on $E$ can be performed in a few minutes. We give details in Table 1, in the table, the points involved in the decomposition are described by their abscissa together with a + or − sign that indicates whether the "real" part of the ordinate has a positive or negative representative in $]-(q/2),(q/2)[$. Similarly, we indicate the choice of the points on $E$ with a + or a − depending on the representative of the constant term in the ordinate[6].

| Points | Mult. Conorm-Norm | Mult. Nagao | Points in decomposition |
|---|---|---|---|
| $(X_0)^-$ | 2 | 341 | $370864^-$ $2471314^+$ $2517710^-$ $3195688^-$ $3512289^-$ $3700196^-$ |
| $(X_0+1)^-$ | 2 | 1664 | $1030818^+$ $2692469^+$ $2731382^-$ $3612676^+$ $3920772^-$ $4172888^+$ |
| $(X_0+2)^-$ | 4 | 85 | $399440^-$ $705045^-$ $901013^-$ $1366937^+$ $2079739^+$ $3419126^+$ |
| $(X_0+3)^+$ | 1 | 655 | $37064^+$ $2305706^+$ $2573803^+$ $2665635^-$ $3263560^-$ $4118343^-$ |
| $(X_0+5)^-$ | 2 | 72 | $311191^-$ $1011994^+$ $2166025^-$ $2649962^-$ $2777633^-$ $2900897^+$ |
| $(X_0+11)^-$ | 4 | 140 | $291295^+$ $518109^-$ $863097^-$ $1733917^+$ $3082470^-$ $3588239^+$ |
| $(X_0+12)^+$ | 3 | 1139 | $230555^-$ $385454^+$ $790502^-$ $985560^+$ $1466691^-$ $4062680^+$ |

**Table 1.** Details of individual logarithm computations.

The group structure of $E$ is $\mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/(2\ell)\mathbb{Z}$ and all the logarithms are computed modulo $\ell$. Thus, in order to obtain points of order $\ell$, we multiply each of the points $P_j$ by 2. To obtain the discrete logarithms in base $P_1$, we simply divide all the obtained logarithms by the logarithm of $P_1$. Finally, we obtain:

$$
\begin{aligned}
2 \cdot P_2 &= 771503218032571282830154288894596893 83 \cdot 2 \cdot P_1 \\
2 \cdot P_3 &= 277607596028887848748187645469867507392 \cdot 2 \cdot P_1 \\
2 \cdot P_4 &= 950556100385309676489669420946201334105 \cdot 2 \cdot P_1 \\
2 \cdot P_5 &= 317720686887855216292082605854593050146 \cdot 2 \cdot P_1 \\
2 \cdot P_6 &= 131228309389018991767706064340727221 4266 \cdot 2 \cdot P_1 \\
2 \cdot P_7 &= 119035784514809263757574253742461295 5882 \cdot 2 \cdot P_1 .
\end{aligned}
$$

All in all, we have been able to completely solve the discrete logarithm problem on the 130-bit elliptic curve $E$ in less than 30 hours real-time with no more than 200 cores, corresponding approximately to 3700 CPU·hours. This is several orders of magnitude faster than any generic algorithm. For comparison, an attack of the Certicom ECC2K-130 challenge using a state-of-the-art implementation of Pollard's Rho is currently underway [4]; it was expected to solve the DLP on a Koblitz elliptic curve defined over $\mathbb{F}_{2^{131}}$ in about one year with 3000 3GHz Core 2 CPUs, but has not succeeded yet.

[6] We did not really choose the points, but simply took the first point produced by Magma with the specified abscissa.

## 6 Conclusion and perspectives

In this paper, we have proposed a new index calculus algorithm to compute discrete logarithms on elliptic curves defined over extension fields of composite degree. In particular, degree 6 extensions are very well-suited to this method, as we have practically demonstrated on a 130-bit example.

This combination of cover and decomposition techniques raises many questions. For example, it would be interesting to know if elliptic curves of prime cardinality defined over a degree 6 extension field can be efficiently attacked. A related problem is how to target more curves easily: this requires either an improvement of the isogeny walk, or an efficient use of non-hyperelliptic covers. Finally, whether our method applies to different extension degrees is an important issue; clearly, degree 4 extensions are also susceptible, but the advantage over generic methods is then less significant.

## References


1. L. M. Adleman. The function field sieve. In *Algorithmic number theory (Ithaca, NY, 1994)*, volume 877 of *Lecture Notes in Comput. Sci.*, pages 108–121. Springer, Berlin, 1994.
2. L. M. Adleman, J. DeMarrais, and M.-D. Huang. A subexponential algorithm for discrete logarithms over the rational subgroup of the Jacobians of large genus hyperelliptic curves over finite fields. In *Algorithmic number theory (Ithaca, NY, 1994)*, volume 877 of *Lecture Notes in Comput. Sci.*, pages 28–40. Springer, Berlin, 1994.
3. S. Arita, K. Matsuo, K.-I. Nagao, and M. Shimura. A weil descent attack against elliptic curve cryptosystems over quartic extension fields. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, E89-A:1246–1254, May 2006.
4. D. V. Bailey, L. Batina, D. J. Bernstein, P. Birkner, J. W. Bos, H.-C. Chen, C.-M. Cheng, G. van Damme, G. de Meulenaer, L. J. D. Perez, J. Fan, T. Gneysu, F. Gurkaynak, T. Kleinjung, T. Lange, N. Mentens, R. Niederhagen, C. Paar, F. Regazzoni, P. Schwabe, L. Uhsadel, A. V. Herrewege, and B.-Y. Yang. Breaking ECC2K-130. Cryptology ePrint Archive, Report 2009/541, 2009. `http://eprint.iacr.org/`.
5. D. V. Bailey and C. Paar. Efficient arithmetic in finite field extensions with application in elliptic curve cryptography. *J. Cryptology*, 14(3):153–176, 2001.
6. I. F. Blake, G. Seroussi, and N. P. Smart. *Elliptic curves in cryptography*, volume 265 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, Cambridge, 2000. Reprint of the 1999 original.
7. I. F. Blake, G. Seroussi, and N. P. Smart, editors. *Advances in elliptic curve cryptography*, volume 317 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, Cambridge, 2005.
8. W. Bosma, J. J. Cannon, and C. Playoust. The Magma algebra system I: The user language. *J. Symb. Comput.*, 24(3/4):235–265, 1997.
9. C. Diem. The GHS attack in odd characteristic. *J. Ramanujan Math. Soc.*, 18(1):1–32, 2003.
10. C. Diem. An index calculus algorithm for plane curves of small degree. In *Algorithmic number theory*, volume 4076 of *Lecture Notes in Comput. Sci.*, pages 543–557. Springer, Berlin, 2006.
11. C. Diem. On the discrete logarithm problem in elliptic curves. Preprint, available at: `http://www.math.uni-leipzig.de/~diem/preprints/dlp-ell-curves.pdf`, 2009.
12. G. Frey. How to disguise an elliptic curve (weil descent). Talk at the 2nd Elliptic Curve Cryptography Workshop (ECC), 1998.
13. S. D. Galbraith, F. Hess, and N. P. Smart. Extending the GHS Weil descent attack. In *Advances in cryptology—EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Comput. Sci.*, pages 29–44. Springer, Berlin, 2002.
14. P. Gaudry. An algorithm for solving the discrete log problem on hyperelliptic curves. In *Advances in cryptology—EUROCRYPT 2000*, volume 1807 of *Lecture Notes in Comput. Sci.*, pages 19–34. Springer, Berlin, 2000.
15. P. Gaudry. Index calculus for abelian varieties of small dimension and the elliptic curve discrete logarithm problem. *J. Symbolic Computation*, 2008. doi:10.1016/j.jsc.2008.08.005.
16. P. Gaudry, F. Hess, and N. P. Smart. Constructive and destructive facets of Weil descent on elliptic curves. *J. Cryptology*, 15(1):19–46, 2002.
17. P. Gaudry, E. Thomé, N. Thériault, and C. Diem. A double large prime variation for small genus hyperelliptic index calculus. *Mathematics of Computation*, 76:475–492, 2007.
18. F. Hess. Generalising the GHS attack on the elliptic curve discrete logarithm problem. *LMS J. Comput. Math.*, 7:167–192 (electronic), 2004.

19. A. Joux and V. Vitse. A variant of the F4 algorithm. To appear in CT-RSA 2011.
20. A. Joux and V. Vitse. Elliptic curve discrete logarithm problem over small degree extension fields. Application to the static Diffie–Hellman problem on $E(\mathbb{F}_{q^5})$. Cryptology ePrint Archive, Report 2010/157, 2010.
21. N. Koblitz. Elliptic curve cryptosystems. *Math. Comp.*, 48(177):203–209, 1987.
22. B. A. LaMacchia and A. M. Odlyzko. Computation of discrete logarithms in prime fields. *Des. Codes Cryptogr.*, 1(1):47–62, 1991.
23. A. K. Lenstra and H. W. Lenstra, Jr., editors. *The development of the number field sieve*, volume 1554 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1993.
24. A. Menezes, E. Teske, and A. Weng. Weak fields for ECC. In *Topics in cryptology—CT-RSA 2004*, volume 2964 of *Lecture Notes in Comput. Sci.*, pages 366–386. Springer, Berlin, 2004.
25. V. S. Miller. Use of elliptic curves in cryptography. In *Advances in cryptology—CRYPTO '85 (Santa Barbara, Calif., 1985)*, volume 218 of *Lecture Notes in Comput. Sci.*, pages 417–426. Springer, Berlin, 1986.
26. J. M. Miret, R. Moreno, D. Sadornil, J. Tena, and M. Valls. An algorithm to compute volcanoes of 2-isogenies of elliptic curves over finite fields. *Applied Mathematics and Computation*, 176(2):739–750, 2006.
27. F. Momose and J. Chao. Scholten forms and elliptic/hyperelliptic curves with weak weil restrictions. Cryptology ePrint Archive, Report 2005/277, 2005. `http://eprint.iacr.org/`.
28. K.-i. Nagao. Decomposed attack for the jacobian of a hyperelliptic curve over an extension field. In *Algorithmic Number Theory – ANTS-IX*. Springer–Verlag, 2010.
29. E. Nart and C. Ritzenthaler. Genus 3 curves with many involutions and application to maximal curves in characteristic 2, 2009. To appear in the proceedings of AGCT-12.
30. J. Scholten. Weil restriction of an elliptic curve over a quadratic extension. Preprint, available at `http://homes.esat.kuleuven.be/~jscholte/weilres.pdf`, 2003.
31. I. Semaev. Summation polynomials and the discrete logarithm problem on elliptic curves. Cryptology ePrint Archive, Report 2004/031, 2004.
32. V. Shoup. Lower bounds for discrete logarithms and related problems. In *Advances in cryptology—EUROCRYPT '97 (Konstanz)*, volume 1233 of *Lecture Notes in Comput. Sci.*, pages 256–266. Springer, Berlin, 1997.
33. N. P. Smart. How secure are elliptic curves over composite extension fields? In *Advances in cryptology—EUROCRYPT 2001*, volume 2045 of *Lecture Notes in Comput. Sci.*, pages 30–39. Springer, Berlin, 2001.
34. N. Thériault. Weil descent attack for Kummer extensions. *J. Ramanujan Math. Soc.*, 18(3):281–312, 2003.

# A Genus 3 cover

## A.1 Odd characteristic

We consider elliptic curves defined over $\mathbb{F}_{q^3}$ of the form

$$y^2 = h(x)(x-\alpha)(x-\sigma(\alpha)) \tag{6}$$

where $\sigma$ is the Frobenius automorphism of $\mathbb{F}_{q^3}/\mathbb{F}_q$, $\alpha \in \mathbb{F}_{q^3} \setminus \mathbb{F}_q$ and $h \in \mathbb{F}_q[x]$ of degree 1 or 2. Such elliptic curves were studied by [9, 34]; they are the only elliptic curves for which the GHS attack yields a cover by a hyperelliptic curve $H$ of genus 3 defined over $\mathbb{F}_q$.

We give now an explicit description of the cover $\pi : H \to E$; following [27], we express this cover as a quotient by a bi-elliptic involution, instead of using the GHS approach. For simplicity, we will assume that $h(x) = x$ (this can always be achieved by an appropriate change of coordinates if $h$ has a root in $\mathbb{F}_q$).

Let $\phi : x \mapsto \frac{D}{x-\sigma^2(\alpha)} + \sigma^2(\alpha)$ be the unique involution of $\mathbb{P}^1(\overline{\mathbb{F}_q})$ sending $\sigma^2(\alpha)$ to $\infty$ and $\alpha$ to $\sigma(\alpha)$, so that $D = \big(\alpha - \sigma^2(\alpha)\big)\big(\sigma(\alpha) - \sigma^2(\alpha)\big)$. If $\phi$ lifts to an involution of a hyperelliptic curve $H_{|\mathbb{F}_q}$, then necessarily $\phi^\sigma$ and $\phi^{\sigma^2}$ will be also involutions of $H$. Observing that $\{Id, \phi, \phi^\sigma, \phi^{\sigma^2}\}$ forms a group, this leads us to consider the curve of equation $y^2 = x + \phi(x) + \phi^\sigma(x) + \phi^{\sigma^2}(x)$; a more usual form for this equation is

$$H: \quad y^2 = F(x)N(x) \tag{7}$$

where $N(x) = (x-\alpha)\,(x-\sigma(\alpha))\,\left(x-\sigma^2(\alpha)\right)$ is the minimal polynomial of $\alpha$ over $\mathbb{F}_q$ and $F(x) = N(x)\left(x+\phi(x)+\phi^{\sigma}(x)+\phi^{\sigma^2}(x)\right) \in \mathbb{F}_q[x]$. It is clear that $\phi$ gives an involution of $H$, still denoted by $\phi : (x,y) \mapsto \left(\frac{D}{x-\sigma^2(\alpha)} + \sigma^2(\alpha), y\frac{D^2}{(x-\sigma^2(\alpha))^4}\right)$.

The quotient of this genus 3 hyperelliptic curve $H$ by $\phi$ is the elliptic curve

$$E' : \quad y^2 = (x-\alpha-\sigma(\alpha))\left(x^2-4\alpha\sigma(\alpha)\right)$$

and the quotient map $\pi' : H \to E'$ satisfies $\pi'(x,y) = \left(x+\phi(x), y/(x-\sigma^2(\alpha))^2\right)$. The curve $E'$ is 2-isogenous to the original curve $E : y^2 = x(x-\alpha)(x-\sigma(\alpha))$ via the map:

$$(x,y) \mapsto \left(\frac{x^2-4\alpha\sigma(\alpha)}{4(x-\alpha-\sigma(\alpha))}, y\frac{(x-2\alpha)(x-2\sigma(\alpha))}{8(x-\alpha-\sigma(\alpha))^2}\right).$$

Finally, the cover map $\pi : H \to E$ has the expression

$$\pi(x,y) = \left(\frac{F(x)}{4N(x)}, \frac{y(x-\phi^{\sigma}(x))(x-\phi^{\sigma^2}(x))}{8N(x)(x-\sigma^2(\alpha))}\right). \tag{8}$$

In the general case, when $E$ has equation (6), the cover (8) remains the same and the corresponding hyperelliptic curve $H$ of genus 3 defined over $\mathbb{F}_q$ has the following equation:

$$H : \quad y^2 = 4N(x)^2\, h\left(\frac{F(x)}{4N(x)}\right).$$

### A.2 Characteristic 2

Let $E$ be an ordinary curve defined over a binary field $\mathbb{F}_{q^3}$; it admits an equation of the form

$$E : \quad y^2 + xy = x^3 + ax^2 + b \tag{9}$$

where $b = 1/j(E)$. As already apparent in [16], the GHS attack produces a genus 3 hyperelliptic cover of $E$ when $\mathrm{Tr}_{\mathbb{F}_{q^3}/\mathbb{F}_q}(b) = 0$, so that $\Theta(q^2)$ curves are directly vulnerable. To describe this cover, we slightly adapt the description of [27, 29], already used in the previous subsection.

Let $\sigma : x \mapsto x^q$ be the Frobenius automorphism and let $v = \sqrt[4]{b}$; by assumption its trace over $\mathbb{F}_q$ is zero. As in the case of odd characteristic, we consider the involution $\phi : \; x \mapsto \frac{\sigma(v)\sigma^2(v)}{x+v} + v$ of $\mathbb{P}^1(\overline{\mathbb{F}_q})$ sending $v$ to infinity and $\sigma(v)$ to $\sigma^2(v)$. We denote by $N$ the minimal polynomial of $v$ over $\mathbb{F}_q$ and by $F$ the product $N(x)\left(x+\phi(x)+\phi^{\sigma}(x)+\phi^{\sigma^2}(x)\right) \in \mathbb{F}_q[x]$. Then, $\phi$ lifts to a bi-elliptic involution of the hyperelliptic curve $H_{|\mathbb{F}_q}$ defined by

$$H : \quad y^2 + N(x)y = F(x)N(x) + aN(x)^2. \tag{10}$$

The curve $E$ is up to a change of variable the quotient of $H$ by $\phi$ and the cover map from $H$ to $E$ is given by:

$$\pi : (x,y) \mapsto \left(x+\phi(x)+v, \frac{y(x+\phi(x)+v)}{N(x)} + v^2\right). \tag{11}$$