CHIST-ERA

User empowerment for SEcurity and privacy in Internet of Things

Mechanisms and Test-Bed Use-Cases (cryptography)

Deliverable number: D3.4

Version 1.0

Funded by the Future and Emerging Technologies (FET) CHIST-ERA programme of the European Union.

Project Acronym: USEIT
Project Full Title: User empowerment for SEcurity and privacy in Internet of Things
Call: 2015
Grant Number: 20CH21_167531
Project URL: http://useit.eu.prg

Editor:	TUE
Deliverable nature:	Report (R)
Dissemination level:	Public (PU)
Delivery Date:	February 2019
Authors:	Chloe Martindale, TUE

Abstract

We summarize the challenges presented by one of the main building blocks, pairings, of the crypto mechanisms needed for secure privacy-protecting encryption mechanisms meeting the restrictions of the different use cases. In particular, we give a high-level overview of the most recent attacks and two suggestions for countermeasures, along with implementation choices taking into account these countermeasures.

Contents

List of Figures

List of Tables

1 Introduction

This deliverable gives details on a fundamental building block for anonymous signatures and attribute-based encryption, and of a first step towards crypto implementation. That is, in WP 2 (Cryptographic protocols), tasks T2.1, T2.2, and T2.3. The building block in question is the security and fast implementation of pairing-based cryptography.

1.1 Background on pairings

Pairings on elliptic curves have various applications in cryptography, ranging from very basic key exchange protocols, such as one round tripartite Diffie–Hellman [1] [2], to complicated protocols, such as identity-based encryption [3] [4] [5] [6]. Pairings also help to improve currently existing protocols, such as signature schemes, to have shortest possible signatures [7].

1.1.1 Pairings on elliptic curves

Let $\mathbb{G}_1, \mathbb{G}_2$, and $\mathbb{G}_T$ be cyclic groups of prime order r and assume that the discrete logarithm problem is intractable in all three groups. An *abstract pairing* is a bilinear, non-degenerate, efficiently computable map of the form: $\widehat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$. We call $\mathbb{G}_1$ and $\mathbb{G}_2$ the *source groups* and $\mathbb{G}_T$ the *target group*. When $\mathbb{G}_1 \neq \mathbb{G}_2$ the pairing is called *asymmetric*, otherwise it is called *symmetric*.

Let E be an ordinary elliptic curve defined over a prime field $\mathbb{F}_p$ and let r be largest prime such that $r|\#E(\mathbb{F}_p)$. The minimal integer k for which all the r-th roots of unity are contained in $\mathbb{F}_{p^k}$ is called the *embedding degree* of E. For all pairings on elliptic curves that are currently used in cryptography, the source groups $\mathbb{G}_1$ and $\mathbb{G}_2$ are r-order subgroups of $E(\mathbb{F}_{p^k})$ and the target group $\mathbb{G}_T$ is the subgroup $\mu_r \subseteq \mathbb{F}^*_{p^k}$ of r^{th} roots of unity. (Typically $\mathbb{G}_1$ is in fact contained in $E(\mathbb{F}_p)$). That is, a pairing of elliptic curves is a map:

$$\widehat{e} : E(\mathbb{F}_{p^k})[r] \times E(\mathbb{F}_{p^k})[r] \to \mu_r \subset \mathbb{F}^*_{p^k}.$$

The most widely used pairings on ordinary elliptic curves are the *Tate pairing* and its variants and the *Ate pairing* and its variants. All of these different types of pairings can be efficiently computed using variants of Miller's algorithm [8].

1.1.2 Attacks on pairings

For a sufficiently generic elliptic curve $E/\mathbb{F}_p$, the complexity of ECDLP in any r-order subgroup of $E(\mathbb{F}_{p^k})$ is $O(\sqrt{r})$, due to Pollard's rho algorithm. The complexity of DLP in the multiplicative group $\mathbb{F}^*_{p^k}$, however, depends both on the factorisation of k and on how the prime p is constructed. In the case of pairing-friendly curves, we may assume that the prime p is large (at least 256-bits) and that it is derived from the evaluation of a polynomial with degree greater that 2. The asymptotic complexity of DLP in $\mathbb{F}^*_{p^k}$ is then

$$L_N[c, \ell] = \exp\left[\big(c + o(1)\big)\big(\ln N\big)^{\ell}\big(\ln \ln N\big)^{1-\ell}\right], \tag{1.1}$$

with $\ell \in [0, 1]$, $c > 0$ and $N = p^k$.

When k is prime, the asymptotic complexity of DLP in $\mathbb{F}^*_{p^k}$ is $L_N[1/3, 1.923]$; the best known attack is the number field sieve (NFS) method. For composite embedding degrees, Kim and Barbulescu's [9] improvements

on the tower number field sieve (TNFS) method have reduced complexity of DLP in $\mathbb{F}^*_{p^k}$ from $L_N[1/3, 1.923]$ to $L_N[1/3, 1.529]$. These new improvements have immediate consequences on the selection of the extension fields $\mathbb{F}_{p^k}$.

1.1.3 Curve families used for pairings

Curves that are suitable for pairings are called *pairing-friendly curves*, and these curves must satisfy specific properties. It is extremely rare that a randomly generated elliptic curve is pairing-friendly, so pairing-friendly curves have to be generated in a specific way. Examples of famous and commonly used pairing-friendly curves include Barreto-Naehrig curves [10] (BN curves), Barreto-Lynn-Scott curves [11] (BLS curves), and Kachisa-Schaefer-Scott curves [12] (KSS curves).

However, these recommendations all pre-date the attacks outlined above. All of these recommendations attempt to minimize the value of $\rho = k\log(p)/\log(r)$ and use $k = 8$ or 12 for 128-bit security, as previously this was most beneficial for efficient implementation. Taking into account the most recent attacks, the most logical suggestions to update the families are to either increase the value of ρ or to increase the embedding degree k. In the attack paper [9] the option of increasing the base field size $\mathbb{F}_p$ is studied, but this is not necessary. Fotiadis and Konstantinou [13] present families of elliptic curves with increased ρ-value to combat the NFS attacks.

2 Pairings on Hessian curves

Chloe Martindale (TUE) wrote a paper in collaboration with Chitchanok Chuengsatiansup (Université de Lyon) 'Pairing-Friendly Twisted Hessian Curves' [14]. This paper presents efficient formulas for pairing computations; more details below.

2.1 Choice of curves and embedding degrees

One way to improve the performance of pairings is to improve the performance of the underlying point arithmetic. Many authors have studied efficient point arithmetic via the representation of elliptic curves in a specific model, for example, Hessian form [15] [16] and Edwards form [17] [18].

Pairings based on Edwards curves, along with examples of pairing-friendly Edwards curves, were proposed by Arene, Lange, Naehrig and Ritzenthaler [19]. They found that the computation of line functions necessary to compute the pairing is much more complicated than if the curves were written in Weierstrass form. In other words, even though Edwards curves allow faster point arithmetic, this gain is somewhat outweighed by the slower computation of line functions. Li, Wu, and Zhang [20] proposed the use of quartic and sextic twists for Edwards curves, improving the efficiency of both the point arithmetic and the computation of the line functions.

Pairings based on Hessian curves with even embedding degrees were proposed by Gu, Gu and Xie [21]. They provided a geometric interpretation of the group law on Hessian curves along with an algorithm for computing Tate pairing on elliptic curves in Hessian form. However, no pairing-friendly curves in Hessian form were given.

Bos, Costello and Naehrig [22] investigated the possibility of using a model of a curve (such as Edwards or Hessian) allowing for fast point arithmetic and transforming to Weierstrass form for the actual computation of the pairing. They found that for every elliptic curve E in the BN-12, BLS-12, and KSS-18 families of pairing-friendly curves, if E is isomorphic over $\mathbb{F}_p$ to a curve in Hessian or Edwards form, then it is not isomorphic over $\mathbb{F}_{p^k}$ to a curve in Hessian or Edwards form, where k is the embedding degree. This implies that the point arithmetic has to be performed on curves in Weierstrass form — not all curves can be written in special forms such as Hessian or Edwards form. This idea of using different curve models comes at a cost of at least one conversion between other curve models into Weierstrass form.

In [14] we studied the efficiency of curves in Hessian form for pairing computations. Hessian curves with j-invariant 0 have degree-3 twists that can also be written in Hessian form. This means that we could take full advantage of speed-up techniques for point arithmetic and pairing computations that move arithmetic to subfields via the twist, e.g., as studied for Edwards curves in [20], without the expensive curve conversion to Weierstrass form. We use the families proposed by [23], in which we could find three families that can be written in Hessian form.

Regardless of which model of elliptic curve was being studied, most of the previous articles on this topic were considering even embedding degrees. One of the main advantages of even embedding degrees is the applicability of a denominator elimination technique in the pairing computation (avoiding a field inversion) which does not directly apply to odd embedding degrees. Examples of pairing algorithms for curves in Weierstrass form with odd embedding degree include the work by Lin, Zhao, Zhang and Wang in [24], by Mrabet, Guillermin and Ionica in [25], and by Fouotsa, Mrabet and Pecha in [26].

Due to the recent advances in number field sieve (NFS) techniques for attacking the discrete logarithm problem for pairing-friendly elliptic curves over finite fields described above, it is necessary to re-evaluate the security of pairing-friendly curves. In [13], Fotiadis and Konstantinou propose countering these attacks by using families

with a higher ρ-value. In this paper, we investigate the feasibility of an alternative method: increasing the embedding degree.

This has the advantage of keeping the low ρ-value of previously proposed families, but it is disadvantaged by the less efficient pairing computations. This article attempts to analyze the use of Hessian curves in combating this. Previous research on computing pairings with Hessian curves addressed only even embedding degrees, and in order to make use of degree-3 twists the embedding degree should be divisible by 6. Prior to the NFS attacks and their variants, the favoured embedding degree for 128-bit security was 12, so that to increase the embedding degree while making use of cubic twists the next candidate is 15. However, as 15 is odd the formulas of [21] do not apply; for this reason one focus of this article is to provide formulas for embedding degree 15. Similarly, the pre-NFS favourite embedding degree for 192-bit security was 18, which we propose to increase to 21. Observe further that for 192-bit security, the families of [13] all require the embedding degree to be greater than 21.

2.2 Our contributions

In [14], we present formulas for computing pairings on both $\mathbb{G}1 \times \mathbb{G}2$ and $\mathbb{G}2 \times \mathbb{G}1$ for a curve given in Hessian form that admits degree-3 twists. These formulas exploit the degree-3 twists where possible: in moving the point arithmetic in $\mathbb{F}_{p^k}$ to $\mathbb{F}_{p^{k/3}}$ and performing the computations for the line functions in $\mathbb{F}_{p^{k/3}}$ in place of $\mathbb{F}_{p^k}$. For efficient curve arithmetic (before applying the use of twists) we refer to Bernstein, Chuengsatiansup, Kohel, and Lange [27].

We analyze the efficiency of the pairing computation in each case, focussing on the embedding degrees that should correspond to 128- and 192-bit security. Our analysis shows that for embedding degree 12, Hessian curves are outperformed by twisted Edwards curves, but for embedding degrees 15, 21, and 24 our formulas give the most efficient known pairing computation. We do not consider 18 as we do not know of any curve constructions for this case. As explained above, our main focus is on odd embedding degrees, as we propose the use of $k = 15$ and $k = 21$ as a countermeasure against the NFS attacks and their variants.

We also give concrete constructions of pairing-friendly Hessian curves for both embedding degrees and a proof-of-concept implementation of the optimal ate pairing for these cases.

3 Pairings with post-TNFS 128-bit security

Chloe Martindale (TUE) has written a paper in collaboration with Georgios Fotiadis from the University of the Aegean 'Optimal TNFS-secure pairings on elliptic curves with even embedding degree' [28].

The main goal of this paper to present the best choice of pairing and of elliptic curve that gives 128-bit security according to the state-of-the-art. As families of TNFS-secure elliptic curves are already presented in [13], our main contribution is a comprehensive comparison of pairings and elliptic curve shapes and the consequent selection of a curve from the available families. To our knowledge, this is the first suggestion of a 128-bit secure pairing-friendly elliptic curve that takes into account the latest attacks described in Section 1.1.2.

Our comparison takes into account competing candidates for the most efficient pairings [28, Section 2] and competing curve shapes for the most efficient curve arithmetic [28, Section 3]. We also compute the optimal elliptic curve and pairing choice for a 128-bit security level among known candidates; the conclusions are summarized below.

Additionally, we present a new analysis for efficient curve arithmetic in the case of quadratic twists of Edwards curves and Jacobi Quartic curves for the Ate pairing, and of sextic twists of Jacobi Quartic curves.

For every case that we consider we presented an implementation in MAGMA, available at `www.martindale.info/research`.

3.1 Operation counts

We summarize below the operation counts for each pairing and curve type addressed in [28]. In [28] we applied this review to choose the optimal curve in each known TNFS-secure compact family for 128-bit security level (of which there are 9 competing to be the fastest and to which our methods may be applied), and give the best pairing and curve shape for this curve. We then present the best choice of curve, pairing, and curve shape from these 9 choices, giving the optimal known TNFS-secure pairing-friendly elliptic curve for 128-bit security level, as presented in more detail in [28]. This method can easily be applied also to 192- and 256-bit security level.

Notation

- **s**: time required to square an $\mathbb{F}_p$-element.
- **m**: time required to multiply an $\mathbb{F}_p$-element.
- **mc**: time required to multiply by a (small) constant in $\mathbb{F}_p$.
- DBL: doubling steps of Miller's algorithm.
- ADD: addition steps of Miller's algorithm.
- e: final exponentiation in Miller's algorithm.
- b_x: the bit length of x.
- w_x: the Hamming weight of x.

In Tables 3.1 and 3.2, we compare operation counts for DBL and ADD in each of the cases studied in [28]. For simplicity, where relevant the operation counts are for mixed addition (not general addition). The total cost of the twisted Ate pairing $\widehat{a}_e$ is

$$(b_{T_e} - 1)\text{DBL} + (w_{T_e} - 1)\text{ADD} + e$$

and the total cost of the optimal Ate pairing $\widehat{a}_0$ with parameter s is

$$(b_s - 1)\text{DBL} + (w_s - 1)\text{ADD} + e.$$

Table 3.1: Operation counts for DBL

DBL	JQ on $\mathbb{G}_1 \times \mathbb{G}_2$	JQ on $\mathbb{G}_2 \times \mathbb{G}_1$	Ed on $\mathbb{G}_1 \times \mathbb{G}_2$	Ed on $\mathbb{G}_2 \times \mathbb{G}_1$
$2\|k$	$(k^2+k+4)\mathbf{m}$	$\left(2k^2+k\right)\mathbf{m}$	$(k^2+k+4)\mathbf{m}$	$(2k^2+k)\mathbf{m}$
$j \neq 0, 1728$	$+(k^2+8)\mathbf{s}+1\mathbf{mc}$	$+3k^2\mathbf{s}+\frac{k^2}{4}\mathbf{mc}$	$+(k^2+7)\mathbf{s}+2\mathbf{mc}$	$+\frac{11k^2}{4}\mathbf{s}+\frac{k^2}{2}\mathbf{mc}$
$4\|k$	$(\frac{k^2}{2}+\frac{3k}{2}+3)\mathbf{m}$	$(\frac{15k^2}{16}+\frac{k}{2})\mathbf{m}$	$(\frac{k^2}{2}+\frac{3k}{2}+4)\mathbf{m}$	$(2k^2+k)\mathbf{m}$
$j = 1728$	$+(k^2+7)\mathbf{s}+1\mathbf{mc}$	$+\frac{23k^2}{16}\mathbf{s}+\frac{k^2}{16}\mathbf{mc}$	$+(k^2+7)\mathbf{s}+2\mathbf{mc}$	$+\frac{11k^2}{4}\mathbf{s}+\frac{k^2}{2}\mathbf{mc}$
$6\|k$	$(\frac{k^2}{3}+\frac{4k}{3}+4)\mathbf{m}$	$\left(2k^2+k\right)\mathbf{m}$	$(\frac{k^2}{3}+\frac{4k}{3}+4)\mathbf{m}$	$(2k^2+k)\mathbf{m}$
$j = 0$	$+(k^2+8)\mathbf{s}+2\mathbf{mc}$	$+3k^2\mathbf{s}+\frac{k^2}{4}\mathbf{mc}$	$+(k^2+7)\mathbf{s}+3\mathbf{mc}$	$+\frac{11k^2}{4}\mathbf{s}+\frac{k^2}{2}\mathbf{mc}$

Table 3.2: Operation counts for hADD

ADD	JQ on $\mathbb{G}_1 \times \mathbb{G}_2$	JQ on $\mathbb{G}_2 \times \mathbb{G}_1$	Ed on $\mathbb{G}_1 \times \mathbb{G}_2$	Ed on $\mathbb{G}_2 \times \mathbb{G}_1$
$2\|k$	$(k^2+k+16)\mathbf{m}$	$\left(5k^2+k\right)\mathbf{m}$	$(k^2+k+12)\mathbf{m}$	$(4k^2+k)\mathbf{m}$
$j \neq 0, 1728$	$+1\mathbf{s}+4\mathbf{mc}$	$+\frac{k^2}{4}\mathbf{s}+k^2\mathbf{mc}$	$+1\mathbf{mc}$	$+\frac{k^2}{4}\mathbf{mc}$
$4\|k$	$(\frac{k^2}{2}+\frac{3k}{2}+12)\mathbf{m}$	$(\frac{3k^2}{2}+\frac{k}{2})\mathbf{m}$	$(\frac{k^2}{2}+\frac{3k}{2}+12)\mathbf{m}$	$(4k^2+k)\mathbf{m}$
$j = 1728$	$+7\mathbf{s}+1\mathbf{mc}$	$+\frac{7k^2}{16}\mathbf{s}+\frac{k^2}{16}\mathbf{mc}$	$+1\mathbf{mc}$	$+\frac{k^2}{4}\mathbf{mc}$
$6\|k$	$(\frac{k^2}{3}+\frac{4k}{3}+16)\mathbf{m}$	$\left(5k^2+k\right)\mathbf{m}$	$(\frac{k^2}{3}+\frac{4k}{3}+12)\mathbf{m}$	$(4k^2+k)\mathbf{m}$
$j = 0$	$+1\mathbf{s}+5\mathbf{mc}$	$+\frac{k^2}{4}\mathbf{s}+k^2\mathbf{mc}$	$+2\mathbf{mc}$	$+\frac{k^2}{4}\mathbf{mc}$

Besides the comparison in terms of operation count, we also gave the timing of our MAGMA implementation for each of the examples in [28]. These timings are definitely not optimal (but serve as a basic comparison between families) as we have not yet considered optimising finite field arithmetic and the implementation is not yet in C. We leave this for future work.

3.2 Results

We gave a comprehensive comparison of the competing proposals put forward in the literature for curve shapes and pairing choices for elliptic curves with even embedding degree, for each known TNFS-secure complete pairing-friendly family for 128-bit security level. We additionally provided the formulas for the 'gaps' in the literature: utilizing quadratic twists for pairings on $\mathbb{G}_2 \times \mathbb{G}_1$ with Jacobi Quartic and Edwards curves, and utilizing sextic twists for pairings on $\mathbb{G}_1 \times \mathbb{G}_2$ with Jacobi Quartic curves.

Our comparisons showed that, from the currently known TNFS-secure families, the best pairing implementation choice for 128-bit security is the optimal Ate pairing applied to the Jacobi Quartic elliptic curve $E/\mathbb{F}_p : y^2 = dx^4 + 1$ (utilizing quartic twists), where

$$\begin{aligned} p \;=\; & 33519519824866674538373388484527266060280336069350658964695523488 \\ & 42908133596080151967071453287452469772970937967942438575522391344 \\ & 43824272763691057038540 9 \end{aligned}$$

and

$$d = 837987995621666863459334712113181651507008401733766474117388087210727033399020037991767863321863117443242734491985609643880597836109560681909227642596352.$$

This choice comes from Family 1 in [28], for which our MAGMA implementation currently runs in 16ms. We leave an optimised implementation of this example to future work.

4 Conclusions

We have presented two papers studying the two options for updating the security of pairings following the recent attacks. The first, joint work with Chitchanok Chuengsatiansup, studies the option of increasing the embedding degree. The second, joint work with Georgios Fotiadis, studies the option of increasing the ρ-value. Both papers give concrete implementation choices; the second also gives a comprehensive comparison between all implementation choices and an optimal choice. In follow-up work we will compare these two methods and look at decreasing bandwidth requirements using genus 2 curves.

Bibliography

[1] A. Joux, "A One Round Protocol for Tripartite Diffie-Hellman," in *ANTS-IV*, 2000, pp. 385–393, http://cgi.di.uoa.gr/~aggelos/crypto/page4/assets/joux-tripartite.pdf.

[2] ——, "A One Round Protocol for Tripartite Diffie-Hellman," *Journal of Cryptology*, vol. 17, no. 4, pp. 263–276, 2004.

[3] D. Boneh and M. K. Franklin, "Identity-Based Encryption from the Weil Pairing," in *CRYPTO 2001*, 2001, pp. 213–229, http://www.iacr.org/archive/crypto2001/21390212.pdf.

[4] J. Horwitz and B. Lynn, "Toward Hierarchical Identity-Based Encryption," in *Eurocrypt 2002*, 2002, pp. 466–481, http://theory.stanford.edu/~horwitz/pubs/hibe.pdf.

[5] C. Gentry and A. Silverberg, "Hierarchical ID-Based Cryptography," in *Asiacrypt 2002*, 2002, pp. 548–566, http://www.cs.ucdavis.edu/~franklin/ecs228/pubs/extra_pubs/hibe.pdf.

[6] A. Sahai and B. Waters, "Fuzzy Identity-Based Encryption," in *Eurocrypt 2005*, 2005, pp. 457–473, http://eprint.iacr.org/2004/086/.

[7] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the Weil pairing," *Journal of Cryptology*, vol. 17, no. 4, pp. 297–319, 2004, http://crypto.stanford.edu/~dabo/pubs/papers/weilsigs.ps.

[8] V. S. Miller, "The Weil Pairing, and Its Efficient Calculation," *Journal of Cryptology*, vol. 17, no. 4, pp. 235–261, 2004.

[9] T. Kim and R. Barbulescu, "Extended Tower Number Field Sieve: A New Complexity for the Medium Prime Case," in *CRYPTO 2016*, 2016, pp. 543–571.

[10] P. S. Barreto and M. Naehrig, "Pairing-Friendly Elliptic Curves of Prime Order," in *SAC 2005*, 2006, pp. 319–331, http://cryptosith.org/papers/pfcpo.pdf.

[11] P. S. L. M. Barreto, B. Lynn, and M. Scott, "On the Selection of Pairing-Friendly Groups," in *SAC 2003*, 2003, pp. 17–25.

[12] E. J. Kachisa, E. F. Schaefer, and M. Scott, "Constructing Brezing-Weng Pairing-Friendly Elliptic Curves Using Elements in the Cyclotomic Field," in *Pairing 2008*, 2008, pp. 126–135.

[13] G. Fotiadis and E. Konstantinou, "TNFS resistant families of pairing-friendly elliptic curves," *Journal of Theoretical Computer Science*, 2018, (to appear). http://eprint.iacr.org/2018/1017.

[14] C. Chuensatiansup and C. Martindale, "Pairing-friendly twisted hessian curves," in *Progress in Cryptology - INDOCRYPT 2018*, 2018, pp. 228–247.

[15] N. P. Smart, "The Hessian form of an elliptic curve," in *CHES 2001*, 2001, pp. 118–125.

[16] M. Joye and J.-J. Quisquater, "Hessian elliptic curves and side-channel attacks," in *CHES 2001*, 2001, pp. 402–410, http://joye.site88.net/.

[17] H. M. Edwards, "A normal form for elliptic curves," *Bulletin of the American Mathematical Society*, vol. 44, pp. 393–422, 2007, http://www.ams.org/bull/2007-44-03/S0273-0979-07-01153-6/home.html.

[18] D. J. Bernstein and T. Lange, "Faster addition and doubling on elliptic curves," in *Asiacrypt 2007*, 2007, pp. 29–50, http://cr.yp.to/newelliptic/newelliptic-20070906.pdf.

[19] C. Arene, T. Lange, M. Naehrig, and C. Ritzenthaler, "Faster Computation of the Tate Pairing," *IACR Cryptology ePrint Archive*, vol. 2009, p. 155, 2009, http://eprint.iacr.org/2009/155.

[20] L. Li, H. Wu, and F. Zhang, "Pairing Computation on Edwards Curves with High-Degree Twists," in *Inscrypt 2013*, 2014, https://doi.org/10.1007/978-3-319-12087-4_12.

[21] H. Gu, D. Gu, and W. Xie, "Efficient Pairing Computation on Elliptic Curves in Hessian Form," in *ICISC 2010*, 2010, pp. 169–176.

[22] J. W. Bos, C. Costello, and M. Naehrig, "Exponentiating in Pairing Groups," in *SAC 2013*, 2013, https://eprint.iacr.org/2013/458.pdf.

[23] D. Freeman, M. Scott, and E. Teske, "A Taxonomy of Pairing-Friendly Elliptic Curves," *Journal of Cryptology*, vol. 23, no. 2, pp. 224–280, 2010, http://eprint.iacr.org/2006/372/.

[24] X. Lin, C. Zhao, F. Zhang, and Y. Wang, "Computing the Ate Pairing on Elliptic Curves with Embedding Degree $k = 9$," *IEICE Transactions*, vol. 91-A, no. 9, pp. 2387–2393, 2008.

[25] N. E. Mrabet, N. Guillermin, and S. Ionica, "A study of pairing computation for elliptic curves with embedding degree 15," *IACR Cryptology ePrint Archive*, vol. 2009, p. 370, 2009, http://eprint.iacr.org/2009/370.

[26] E. Fouotsa, N. E. Mrabet, and A. Pecha, "Optimal Ate Pairing on Elliptic Curves with Embedding Degree 9, 15 and 27," *IACR Cryptology ePrint Archive*, vol. 2016, p. 1187, 2016, http://eprint.iacr.org/2016/1187.

[27] D. J. Bernstein, C. Chuengsatiansup, D. Kohel, and T. Lange, "Twisted Hessian Curves," in *LATINCRYPT 2015*, 2015, pp. 269–294, http://cr.yp.to/papers.html#hessian.

[28] G. Fotiadis and C. Martindale, "Optimal tnfs-secure pairings on elliptic curves with even embedding degree," *Cryptology ePrint Archive*, 2018, https://eprint.iacr.org/2018/969.