The Magic of Elliptic Curves and Public-Key Cryptography

Florian Heß, **Andreas Stein**, **Sandra Stein**
Institut für Mathematik
Carl-von-Ossietzky Universität Oldenburg
D-26111 Oldenburg
{florian.hess, andreas.stein1, sandra.stein}@uni-oldenburg.de

Manfred Lochter
Bundesamt für Sicherheit in der Informationstechnik (BSI)
Postfach 200363
D-53133 Bonn
manfred.lochter@bsi.bund.de

March 12, 2012

Abstract

Elliptic curves are beautiful mathematical objects that again and again appear in the most surprising places. Their history certainly originates at least in ancient Greece, whereas the study of arithmetic properties of elliptic curves as objects in algebra, geometry, and number theory traces back to the nineteenth century. Curiously, the earliest use of the term "elliptic curve" in the literature seems to have been by James Thomson in 1727 in "A Poem sacred to the Memory of Sir Isaac Newton":

> "He, first of Men, with awful Wing pursu'd the Comet tro' the long Elliptic Curve."

In 1985, Koblitz and Miller independently proposed to use elliptic curves in cryptography which can only be described as a magnificent and practical application of elliptic curves. This paper intends to mostly present a low-brow introduction of elliptic curves and their use in real-world applications of public-key cryptography.

Andreas

Keywords Cryptography, elliptic curves, discrete logarithm problem, public-key cryptography, pairing-based cryptosystem, Weil pairing, Tate-Lichtenbaum pairing, side channel analysis.

AMS 2010 Mathematics Subject Classification 94A60, 14H52, 11T71, 14G50, 68P25, 11Y40, 11Y16.

1 Introduction

This paper is a short survey on the use of elliptic curves in public-key cryptography. It contains merely topics from well selected areas. Mostly, our aim is to give a low-brow introduction for non-experts and a motivation for further research. We will also provide new results in pairing-based cryptography. The main relevant tools from arithmetic geometry are nicely summarized in a recent article by Frey [Fre10]. In order to obtain a thorough understanding of the constructive and the destructive aspects of elliptic curve cryptography (ECC), tools from the following research areas are required: Number theory, algebra, geometry, cryptology, theoretical computer science, efficient implementations, measurement technology, stochastics, lattices and many others.

In their seminal article in 1976, Diffie and Hellman [DH76] introduced public-key cryptography through a key agreement protocol which uses arithmetic in the multiplicative group $\mathbb{F}_p^*$ of a finite field $\mathbb{F}_p$ of large prime order p. With this protocol two communication partners, usually called A and B or Alice and Bob, agree on a secret key for a symmetric encryption algorithm over an insecure channel. The security of the Diffie-Hellman scheme is related to the presumed computational difficulty of computing discrete logarithms in $\mathbb{F}_p^*$. In 1985, ElGamal [ElG85] then invented a public-key encryption protocol and a signature scheme whose security also relies on the presumed computational intractability of the discrete logarithm problem in $\mathbb{F}_p^*$.

These concepts and their subsequent refinements can be extended to arbitrary finite groups G of order n as long as there exist efficient ways of representing group elements and computing the group law. Furthermore, one should select the group G so that the following discrete[1] logarithm problem (DLP) in G is computationally infeasible: Given two group elements $g, h \in G$, where h lies in the subgroup of G generated by g, determine an integer ℓ such that $h = g^\ell$ and $0 \leq \ell < n$.

In addition to the DLP, and depending on the cryptographic application, the following weaker computational problems are also required to be computationally infeasible: The computational Diffie-Hellman problem (CDH) is to compute $g^{\ell_1 \ell_2}$ from g, h_1, h_2, where ℓ_1 and ℓ_2 are the discrete logarithms of h_1 and h_2 respectively to the base g, and the decisional Diffie-Hellman (DDH) problem is to decide whether $h = g^{\ell_1 \ell_2}$ given g, h_1, h_2, h, with ℓ_1 and ℓ_2 as above.

A very popular and efficient choice of G is the group of points on an elliptic curve over a finite field as independently suggested by Koblitz [Kob87] and Miller [Mil86]. A thorough and comprehensive discussion of elliptic curve cryptography can be found in [CFA+05, HMV04, BSS00, BSSC05, Was08]. For the arithmetic of elliptic curves, we refer to [Sil09, Sil94].

In this article we will only consider elliptic curve variants of the relevant protocols. It is very advantageous that with an elliptic curve cryptosystem the corresponding elliptic curve discrete logarithm problem (ECDLP) appears to be significantly harder than the DLP in conventional discrete logarithm systems if the underlying elliptic curve is properly chosen. We will discuss the parameter choices for ECC in this survey according to their realization in standards which is based on the current attacks to the ECDLP and computer technology. Note that there is no mathematical proof that the ECDLP is intractable.

[1] Sometimes the term DLP is exclusively reserved for the discrete logarithm problem in $\mathbb{F}_p^*$.

1.1 Domain parameters

For the choice of the curve, we mainly follow the recommendations of the ECC Brainpool [ECC05, LM10], a group consisting of universities, industry and government with the goal of promoting elliptic curve cryptography.[2] There are slight deviations of the recommendations depending on the requirements for the applications and the intended level of security. Let E be an elliptic curve defined over the finite field $\mathbb{F}_q$ with q elements, where q is a power of a prime p. We then consider the group $G = E(\mathbb{F}_q)$ of $\mathbb{F}_q$-rational points on E plus a point P on E of order n with coefficients in $\mathbb{F}_q$ and impose the following **domain parameters** on E, q, $\#E(\mathbb{F}_q)$ as well as n in order to prevent mathematical attacks.

- q satisfies
  - either $q = 2^{p'}$, where p' is prime.
  - or $q = p$, where p is prime.
- $q \sim 2^{224}$, 2^{256}, 2^{320}, 2^{384}, or 2^{512}, dependent on the application.
- $\#E(\mathbb{F}_q) = \lambda n$, where n is a prime and $\lambda = 1, 2, 3$, or 4.
- E is not anomalous. (See Section 5.3; for $q = p > 3$, this means $\#E(\mathbb{F}_q) \neq p$.)
- E fulfills the Menezes-Okamoto-Vanstone condition, i.e. $n - 1$ divided by the order of p modulo n is less than 10000. That is, the ECDLP in G must not be reducible to the DLP in a multiplicative group $\mathbb{F}^*_{p^k}$ of a finite field $\mathbb{F}_{p^k}$ of p^k elements for a 'small' integer k. This includes the case that the elliptic curve must not be supersingular. For $q = p > 3$, this means $n \neq p + 1$.
- The class number of the fundamental order of the endomorphism ring of E should be at least 200. This condition counters (hypothetical) lifting attacks. It was first introduced by Spallek in her Diploma thesis, supervised by Frey. When the condition was first introduced it de-facto prohibited the CM method explained below for the construction of cryptographically strong curves. This has changed during the last years, due to improved construction methods. A method for checking the class number condition is described in [LM10].

The choice of the domain parameters seems arbitrary at first sight. We will attempt to justify these values in the course of this paper and refer to Section 5 and 6 for further explanations.

[2] There also exist various technical guidelines and standards published by the German Federal Office for Information Security (BSI) and the National Institute of Standards and Technology (NIST), an agency of the U.S. Department of Commerce, e.g. [BSI09, NIS09, NIS11].

1.2 What is Cryptography?

Historically, cryptography was identified with the design and implementation of secrecy systems. In the last years cryptography has become more than that and covers a broader range of topics related to information security such as confidentiality, integrity, authenticity, and non-repudiation. A cryptosystem is formally defined (see e.g. [Buc04, Sti05]) as a quintuple $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$, where $\mathcal{P}$ is a finite set of admissible plaintext messages, $\mathcal{C}$ is a finite set of admissible ciphertext messages, $\mathcal{K}$ is a finite set of possible keys $K \in \mathcal{K}$, i.e. the key space, $\mathcal{E} = \{E_K | \, K \in \mathcal{K}\}$ is a set of encryption functions $E_K : \mathcal{P} \to \mathcal{C}$, and $\mathcal{D} = \{D_K | \, K \in \mathcal{K}\}$ is a set of decryption functions $D_K : \mathcal{C} \to \mathcal{P}$. For each $K \in \mathcal{K}$, there exist an $E_K \in \mathcal{E}$ and a $D_K \in \mathcal{D}$ such that the following condition is satisfied:

$$D_K(E_K(M)) = M \quad \text{for all } M \in \mathcal{P} \ .$$

A symmetric cryptosystem is constructed in a way so that either D_K and E_K are the same or can be easily derived from each other. Exposure of either D_K or E_K will immediately reveal both and the cryptosystem with the key K will be completely insecure. Necessarily K must be secret and a prior communication of the key between the communicants needs to take place. An asymmetric cryptosystem is constructed so that for each $K \in \mathcal{K}$, it is infeasible to determine D_K given E_K.

1.3 Elliptic curves over finite fields

We only mention some basic properties of elliptic curves over finite fields that are needed to understand the use of elliptic curves in cryptography. For details we refer to [Sil09, Was08, HMV04, CFA+05, BSS00].

Let $\mathbb{F}_q$ be a finite field of q elements and let p be its prime characteristic. An **elliptic curve** E over $\mathbb{F}_q$ is a smooth projective curve over $\mathbb{F}_q$ given by a homogeneous equation $F(x, y, z) \in \mathbb{F}_q[x, y, z]$ of degree 3. Let r be any positive integer. The projective solutions $P = [x : y : z] \in \mathbb{P}^2(\mathbb{F}_{q^r})$ to this homogeneous polynomial F of degree 3 form the set of $\mathbb{F}_{q^r}$-rational points of E. By remembering that there is one point $\mathcal{O} = [0 : 1 : 0]$ on E at infinity and identifying the finite points $P = [x : y : 1]$ with affine points (x, y), one represents an elliptic curve E as an affine Weierstrass equation

$$E : \, y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6 \qquad (a_i \in \mathbb{F}_q) \ . \tag{1.1}$$

The set of $\mathbb{F}_{q^r}$-rational points of E is then

$$E(\mathbb{F}_{q^r}) = \{(x, y) \in \mathbb{F}_{q^r} \times \mathbb{F}_{q^r} : \, y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6\} \cup \{\mathcal{O}\} \ .$$

Note that for $p > 3$ the affine equation (1.1) of an elliptic curve E can be transformed into an isomorphic short Weierstrass form

$$E : \, y^2 = x^3 + ax + b \ , \tag{1.2}$$

where $a, b \in \mathbb{F}_q$ and $\Delta = -16(4a^3 + 27b^2) \neq 0$.

According to a famous theorem of Hasse the cardinality of $E(\mathbb{F}_{q^r})$ is

$$\#E(\mathbb{F}_{q^r}) = (q^r + 1) - t \quad \text{where} \quad |t| \leq 2q^{\frac{r}{2}} \ . \tag{1.3}$$

t is called the trace of E over $\mathbb{F}_{q^r}$. For large values of q^r this means in particular that $\#E(\mathbb{F}_{q^r}) \approx q^r$. If the characteristic p of $\mathbb{F}_q$ divides t then the elliptic curve E is called supersingular, otherwise ordinary.

For cryptographic applications it is fundamental that $E(\mathbb{F}_{q^r})$ forms an abelian group with neutral element $\mathcal{O}$. The group law can be interpreted geometrically by a tangent and chord method and is usually written additively. For any positive integer m we denote by $E(\mathbb{F}_{q^r})[m]$ the subgroup of $E(\mathbb{F}_{q^r})$ consisting of points whose order divides m.

The main reasons why the group $E(\mathbb{F}_{q^r})$ of $\mathbb{F}_{q^r}$-rational points of E is a wonderful choice of a finite group G for applications in public-key cryptography are the following. Firstly, points $P \in E(\mathbb{F}_{q^r}) \setminus \{\mathcal{O}\}$ are easily representable in affine coordinates and the group law of $E(\mathbb{F}_{q^r})$ is efficiently computable in the coordinates of points. Secondly, we know that $\#E(\mathbb{F}_{q^r}) \approx q^r$ for large values of q^r. This means that the group size is roughly the size of the finite field $\mathbb{F}_{q^r}$ and parameters can be selected accordingly. In addition, there exist algorithms for efficiently computing $\#E(\mathbb{F}_{q^r})$ for the sizes given in Section 1.1. Thirdly, the discrete logarithm problem in $E(\mathbb{F}_{q^r})$ for the domain parameters in Section 1.1 appears to be computationally infeasible.

1.4 Selected applications of elliptic curves

In the internet age elliptic curves are almost ubiquitous. Everyone who opens a secure session using his web browser has a good chance to use an elliptic curve based protocol. The protocols SSL/TLS which are used for https sessions support elliptic curves. Also packet-based communication based on the IPv4 or IPv6 protocol supports elliptic curve based security mechanisms. Details are for example given with the IPSec standard, especially with IKEv2. In most web browsers one can verify the certificate of websites by clicking left from the web address. Note that in these applications one often only needs one-sided authentication. We want to be sure to communicate with the right web server, but want to stay anonymous. For efficiency reasons web servers prefer to use the ephemeral-static version of the Diffie-Hellman key agreement (see Section 2.2).

The security of the new German identity documents relies on elliptic curve cryptography as well. One of the core elements is the elliptic curve based protocol for password authenticated connection establishment (PACE). We refer to [BSI10, BFK09] for details.

The PACE protocol is an advanced security mechanism for MRTDs (machine readable travel documents) and the respective reader terminals. It is also a framework for authenticated key exchange between the MRTD chip (of user A) and the terminal (user B). The purpose of PACE is to establish a secure channel based on shared passwords with low entropy, where the domain parameters of the MRTD chip are authenticated by a governmental authority. The PACE protocol is secure in a certain real-or-random sense. It is currently under standardization of ISO/IEC. One interesting fact is that the PACE protocol computes an intermediate point on an elliptic curve that has to be kept secret. Its knowledge breaks the scheme. We only mention two additional ECC-based functions of the new German identity card.

- It offers an eID function (see [FP11]) that allows users to identify them-

selves on the internet. For each authorization process the user determines what information he is willing to transmit to the service provider. One example is age verification. Age verification confirms that a user has reached a certain age. Another application is a pseudonym function that allows the user to communicate anonymously in social networks.

- It allows to use qualified electronic signatures according to the German signature law. For this functionality, an appropriate certificate has to be acquired from a certification service provider.

Furthermore, the high-security encryption solutions used by the German government make heavy use of elliptic curve cryptography.

2 Elliptic curve cryptography

The use of elliptic curves in cryptography was independently proposed by Koblitz [Kob87] and Miller [Mil86]. Let us start with some interesting protocols based on the arithmetic of elliptic curves over finite fields. We present the textbook versions. Some problems arising for real-world applications are briefly mentioned. Assume we have two parties called Alice and Bob and they want to send messages to each other over an insecure channel. An eavesdropper Eve is able to intercept the messages sent over the channel.

2.1 Elliptic curve discrete logarithm problem (ECDLP)

Let E be an elliptic curve defined over a finite field $\mathbb{F}_q$ with q elements, where q is a power of a prime. We define the ECDLP which is the discrete logarithm problem in the group $E(\mathbb{F}_q)$ of $\mathbb{F}_q$-rational points on E, now written additively: Given a point $P \in E(\mathbb{F}_q)$ of prime order n and let $Q \in E(\mathbb{F}_q)$ be another point such that $Q = \ell P$ for some integer ℓ; find $\ell \in [0, n-1]$. In general, the quantities q, E, and P should be selected so that the ECDLP is presumably a computationally difficult problem (see Section 1.1 and Section 5).

2.2 Elliptic curve key agreement protocol

Diffie and Hellman introduced in 1976 a key agreement protocol based on the arithmetic in the multiplicative group of a finite field of large prime characteristic. With this protocol Alice and Bob are able to agree on a secret key over an insecure network, whereas Eve is not able to find out the key. In the elliptic curve analogue of this protocol, Alice and Bob first agree on q, an elliptic curve E defined over $\mathbb{F}_q$, and a point $P \in E(\mathbb{F}_q)$. Then Alice and Bob choose a secret integer d_A respectively d_B at random and compute the points $Q_A = d_A P$ respectively $Q_B = d_B P$. Now the communication over the unsecured channel can take place. Alice sends Q_A to Bob and Bob sends Q_B to Alice. In the end, they can both compute the common secret point, namely $K_{AB} = d_A d_B P \in E(\mathbb{F}_q)$ by

$$K_{AB} = d_A Q_B = d_A(d_B P) = d_B(d_A P) = d_B Q_A \ .$$

Eve knows $Q_A = d_A P$ and $Q_B = d_B P$ but neither d_A nor d_B and therefore she cannot compute $K_{AB} = d_A d_B P$ from the known parameters unless she is able

to solve the elliptic curve Diffie-Hellman problem (ECDHP) which is: Given P, d_AP and d_BP, determine d_Ad_BP. Note that if one can solve the ECDLP one can solve the ECDHP.

Nevertheless the Diffie-Hellman scheme is insecure if the messages exchanged between Alice and Bob are not authenticated. In this case Eve can impersonate Bob against Alice and Alice against Bob. She can then agree on a secret point with both of them and act as a man-in-the-middle. Authentication can be achieved by the use of digital signatures (see Section 2.3) and certificates – which have to be distributed in a trusted way.

In practice often only one of the communication partners changes his ephemeral key (the client), the other one uses a static key (the server). This is called static-ephemeral DH. There are also static-static versions of DH.

2.3 Elliptic curve digital signature algorithm (ECDSA)

Similar to handwritten signatures, one uses digital signatures in todays communication in order to achieve three services: Authentication (assurance of identity), data-integrity (assurance that data has not been modified), and non-repudiation (providing evidence to a third-party that a specific party participated in a transaction). The ECDSA achieves these properties with the help of elliptic curves. Assume that Alice and Bob have agreed on q, an elliptic curve E defined over $\mathbb{F}_q$, and a point $P \in E(\mathbb{F}_q)$ of prime order n. For the ECDSA they also have to agree on a collision-free cryptographic hash function H. We simply interpret H as a one-way function that takes an arbitrary binary string as input and returns an integer less than n. We assume that Alice has a public key $Q_A = d_AP$ and a randomly chosen secret key $d_A \in [1, n-1]$.

In order to sign a message m Alice performs the following operations: First she selects a secret integer[3] $k_e \in [1, n-1]$ at random, computes k_eP and transforms the x-coordinate of k_eP into an integer x_1. Then she calculates $r \equiv x_1 \pmod{n}$. If r happens to be 0, then a new ephemeral key k_e has to be selected. Next, Alice computes the value of the hash function $h = H(m)$ and also $s \equiv k_e^{-1}(h + d_Ar) \pmod{n}$. In the unlikely case that $s = 0$ she has to start over with a new value of k_e. If not, the signature generation is completed and the signature is (r, s) which Alice sends along with the message m to Bob.

Now Bob wants to verify the validity of the signature. He looks up Alice's public key Q_A and checks if r and s are in the interval $[1, n-1]$. Then Bob computes the hash value $h = H(m)$ as well as the values $v_1 \equiv hs^{-1} \pmod{n}$ and $v_2 \equiv rs^{-1} \pmod{n}$ where s^{-1} denotes the multiplicative inverse element of s modulo n. Finally he computes $v_1P + v_2Q_A$ which is a point in $E(\mathbb{F}_q)$. If $v_1P + v_2Q_A = \mathcal{O}$ then Bob simply rejects the signature. In fact, this situation should obviously be avoided. Otherwise $v_1P + v_2Q_A$ has coordinates in $\mathbb{F}_q$ and the x-coordinate of this point can be transformed into an integer x_2. Bob has to verify that x_2 is congruent to $r \pmod{n}$. If this is the case the signature is accepted, otherwise it is rejected.

In practice the way of choosing ephemeral keys is crucial. For example there is an unpublished attack on the ECDSA and similar schemes that uses the fact that ephemeral keys will be biased depending on the output of a pseudorandom number generator. This attack led to a change of NIST's original Digital Signa-

[3]This randomly selected, secret integer is usually called the ephemeral key.

ture Standard (DSS). In Germany the use of legally binding digital signatures is regulated by law.

2.4 Elliptic curve ElGamal public-key cryptosystem

Another protocol of elliptic curve cryptography is the elliptic curve version of the ElGamal public-key cryptosystem which works as follows. In a precomputational step, Alice and Bob agree on q, an elliptic curve E defined over $\mathbb{F}_q$, and a point $P \in E(\mathbb{F}_q)$.

Alice randomly chooses a secret multiplier $d_A \in [1, n-1]$, computes the point $Q_A = d_A P$, publishes the point Q_A as her public key, and keeps d_A secret.

Bob wishes to send a message m to Alice. In order to do so, he first looks up Alice's public key Q_A and then converts the plaintext message m into a point $M \in E(\mathbb{F}_q)$. Then Bob selects an integer $k_e \in [1, n-1]$ at random and computes

$$C_1 = k_e P \quad \text{and} \quad C_2 = M + k_e Q_A \ .$$

Finally, the two points (C_1, C_2) are sent to Alice so that she can recover the plaintext by computing

$$C_2 - d_A C_1 = (M + k_e Q_A) - d_A(k_e P) = M + k_e(d_A P) - d_A(k_e P) = M$$

and extracting m from M.

Formally, this cryptosystem is the quintuple $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$, where $\mathcal{P} = E(\mathbb{F}_q)$, $\mathcal{C} = E(\mathbb{F}_q) \times E(\mathbb{F}_q)$, and $\mathcal{K} = \{(q, E, P, n, Q, d) \mid Q = dP\}$. For each $K = (q, E, P, n, Q, d) \in \mathcal{K}$ and $k_e \in [1, n-1]$, $E_K(M) := (k_e P, M + k_e Q)$. For $(C_1, C_2) \in E(\mathbb{F}_q) \times E(\mathbb{F}_q)$, $D_K(C_1, C_2) := C_2 - dC_1$.

We just mention that there are more refined encryption systems such as the Elliptic Curve Integrated Encryption Scheme (ECIES) which are popular choices for state-of-the-art implementations. For details on practical considerations and vulnerabilities of the plain version of the ElGamal cryptosystem described here, we refer for instance to [BJN00].

3 RSA

We now present the ultimate example for an asymmetric cryptosystem, namely the RSA cryptosystem, named after its inventors Rivest, Shamir, and Adleman. For details, we refer to any textbook in cryptography (see e.g. [Sti05] or simply the original research paper [RSA78]).

The communication problem is the following: The sender Alice wishes to transmit a message m securely over a public channel to the receiver Bob. This can be accomplished by the following steps.

1. Bob
   - generates two large primes p and q.
   - computes $N = pq$ and $\varphi(N) = (p-1)(q-1)$, where φ denotes Euler's phi function.
   - chooses a random integer e, $1 < e < \varphi(N)$, so that $\gcd(e, \varphi(N)) = 1$.
   - computes $d \equiv e^{-1} \pmod{\varphi(N)}$ using the Euclidean algorithm.

- publishes (N, e) and keeps d, p, q secret.

2. Alice wants to send a message to Bob. For the purpose of simplicity, we assume the message m to be already encoded as an integer M such that $0 < M < N$. She
   - looks up Bob's public key (N, e).
   - computes $C \equiv M^e \pmod{N}$.
   - sends C to Bob.
3. Bob recovers the message M via

$$C^d \equiv M^{ed} \equiv M^{1+j\varphi(N)} \equiv M \pmod{N} ,$$

   where j is an integer such that $ed = 1 + j\varphi(N)$.

We point out that the following original version of RSA corresponds to the textbook version and is insufficient for secure implementations. As presented, this protocol is the basic frame of the RSA cryptosystem and not the version that is used in implementations. For details on practical considerations, we refer for instance to [BJN00]. For example the pure scheme is homomorphic and can be attacked by lattice-methods if d is to small.

In order to analyze RSA, we mention the idea of trapdoors. For given N, e, we define the trapdoor one-way RSA function[4] as

$$f_{N,e}(M) := M^e \pmod{N} \quad \text{for } M \in \mathbb{Z}_N \ .$$

This function is clearly related to the above protocol. It is

(a) easy to evaluate $M \mapsto M^e \pmod{N}$.

(b) difficult to invert $C \mapsto C^{\frac{1}{e}} \pmod{N}$ for integers C with $1 < C < N$ and $\gcd(C, N) = 1$.

(c) possible to invert $f_{N,e}(M)$ with the "trapdoor" d.

The protocol immediately produces a digital signature scheme for free, since it is known that trapdoor one-way functions yield digital signatures. In our case, this works as follows:

(a) Bob signs the message M by computing $S \equiv M^d \pmod{N}$ and sends S to Alice.

(b) Alice verifies that $S^e \equiv M \pmod{N}$.

In the formal description, given a positive integer N and primes p, q such that $N = pq$, the corresponding RSA asymmetric cryptosystem is the quintuple $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$, where $\mathcal{P} = \mathbb{Z}_N = \mathcal{C}$, and

$$\mathcal{K} = \{(N, p, q, e, d) \mid \ e, d \in \mathbb{Z}^*_{\varphi(N)},\ ed \equiv 1 \pmod{\varphi(N)}\} \ .$$

[4]For any positive integer N we let $\mathbb{Z}_N = \{a \in \mathbb{Z} \mid 0 \leq a < N\}$ be the set of representatives of $\mathbb{Z}/N\mathbb{Z}$. Then $\mathbb{Z}_N$ is a group under addition modulo N with identity 0. Its group of units $\mathbb{Z}^*_N = \{a \in \mathbb{Z} \mid 1 \leq a < N,\ \gcd(a, N) = 1\}$ forms a group under multiplication modulo N with identity 1.

For each $K = (N, p, q, e, d) \in \mathcal{K}$ and $M, C \in \mathbb{Z}_N$, we define

$$E_K(M) := M^e \pmod{N} \quad , \quad D_K(C) := C^d \pmod{N} \ .$$

We obtain the following complexity-theoretic problems.

1. **Breaking RSA**: Inverting $f_{N,e}(M)$; that is, given $N = pq$ and e, C with $\gcd(e, \varphi(N)) = 1$ and $C = f_{N,e}(M)$, compute $C^{\frac{1}{e}} \pmod{N}$. This is precisely the problem that has to be solved by an adversary in order to break the RSA system.

2. **Special Integer Factorization Problem (SIFP)**: Given a positive integer N with $N = pq$, where p and q are primes; determine p and q.

It is easy to see that SIFP $\Rightarrow$ Breaking RSA. Suppose we are given N, e and we are able to efficiently factor N. Then we can determine p and q and thus we are able to compute $\varphi(N) = (p-1)(q-1)$. Obviously, we are now in the position to determine the secret key d by applying the extended Euclidean algorithm. With the knowledge of d and N, we have complete control over the RSA system.

The other direction Breaking RSA $\stackrel{?}{\Rightarrow}$ SIFP is quite unclear yet. However, one can show that SIFP $\Leftrightarrow$ Computing d from (N, e). Even though breaking RSA means to invert $f_{N,e}(x)$, one often identifies the mathematical problem of RSA to be the SIFP. This is a hard and well-studied problem in computational number theory, and also in all of number theory. The fastest known algorithmic solution to this problem is the general number field sieve. Heuristically, under the generalized Riemann hypothesis and various other assumptions, the general number field sieve has a subexponential running time of

$$L(N) = O(\mathrm{e}^{(1.923+o(1))(\ln N)^{1/3}(\ln \ln N)^{2/3}}) \tag{3.1}$$

operations, where $o(1) = \theta(N) \to 0$ for $N \to \infty$. For details, we refer to [Coh93, CP05]. Many experts believe that the SIFP is a computationally hard problem. Even Gauss mentioned:

> The problem of distinguishing prime numbers from composites, and of resolving composite numbers into their prime factors, is one of the most important and useful in all of arithmetic... The dignity of science seems to demand that every aid to the solution of such an elegant and celebrated problem be zealously cultivated.

It is the current belief of the community that despite the enormous efforts of many excellent researchers over the last decades, and despite the most sophisticated computer equipment, the problem of efficiently factoring $N = pq$ is not even close to be solved, and currently not even for 2048-bit numbers N.

4 ECC versus RSA

Elliptic curve cryptography is the most attractive alternative to RSA. Why? Most importantly, ECC ought to be faster than RSA in certain applications because the elliptic curve Diffie-Hellman key agreement protocol as well as the

elliptic curve ElGamal public-key cryptosystem should actually need smaller keys and are ad hoc therefore more efficient than RSA. According to the currently best known attacks on RSA and the ECDLP, the cryptographic security grows of ECC exponentially in the length of the input parameters (see Table 1). The only theoretical disadvantages are that the underlying mathematical problem is still considered "new" and that the patent situation is confusing.

What reasons do exist to pursue further research on the still important problems in ECC? Clearly, the current absence of a subexponential-time algorithm for solving the ECDLP implies significantly smaller parameters in ECC than with competing, established technologies such as RSA, but with equivalent levels of security. Thus the key-per-bit-strength of ECC is somewhat better than that of RSA. An immediate consequence from having smaller parameters is the potential gain in speed and the use of smaller certificates. These advantages are especially important for long-term security and in environments where at least one of the following resources is limited: Storage space, bandwidth, or power. Popular examples in Germany are the national ID cards and passports. Summarizing, ECC is especially well-suited for constrained environments such as smart cards, cellular phones, pagers, digital postal marks, and personal digital assistants (PDAs).

5 ECDLP Attacks

Here, we sketch the ideas of some attacks on the ECDLP which ultimately led to the recommended domain parameters introduced in Chapter 1. We investigate the ECDLP as defined in Section 2.1. For further details and explicit formulations of the attacks we refer to [CFA+05, HMV04, GM05].

5.1 Generic attacks

The usual generic attacks for finite abelian groups apply to solving the ECDLP. The most notable ones in this context are the Pohlig-Hellman attack and Pollard's rho method. Let E be an elliptic curve defined over $\mathbb{F}_q$.

The **Pohlig-Hellman attack** is successful if the group order $\#E(\mathbb{F}_q)$ is known, if $\#E(\mathbb{F}_q)$ can be factored into primes by using an integer factorization method, and if $\#E(\mathbb{F}_q)$ is smooth, i.e. splits into small primes only. In that case, the ECDLP can be solved by first solving the ECDLP in small groups of prime order in parallel and by then using a chinese remainder technique to put the pieces together. In order to prevent this attack, it is therefore recommended to choose the parameters so that $\#E(\mathbb{F}_q) = \lambda n$, where n is a prime and $\lambda = 1, 2, 3$, or 4.

Pollard's rho method can always be applied. It uses a generic pseudo-random walk à la Pollard or Teske in $E(\mathbb{F}_q)$ together with a distinguished point method with very little space. It has an overall expected running time of $O(\sqrt{q})$. In particular, this algorithm can be parallelized effectively with a linear speed-up in the number of processors. If additional knowledge is given, then the expected running time of another attack is measured with respect to Pollard's rho method. Also, the size of the underlying finite field in the domain parameters is determined by current and predicted running times of implementations of the parallelized rho method with up-to-date computer equipment.

5.2 Weil pairing and Tate-Lichtenbaum pairing attack

The basic idea of these attacks stems from the so-called Menezes-Okamoto-Vanstone reduction [MOV93]. Let E be an elliptic curve defined over $\mathbb{F}_q$. One makes use of the fact that there exists a non-degenerate pairing, namely the Weil pairing, and similarly the Tate-Lichtenbaum pairing, from $E(\mathbb{F}_q)[n] \times E(\mathbb{F}_q)[n]$ to the multiplicative group $\mathbb{F}_{q^k}^*$ of $\mathbb{F}_{q^k}$ for some positive integer k, where $n|(q^k-1)$. More details about these pairings and their constructive use in cryptography are given in Section 7. Here, the ECDLP can thus be reduced to the discrete logarithm problem in $\mathbb{F}_{q^k}^*$. Now, in $\mathbb{F}_{q^k}^*$ a variation of the general number field sieve method for factoring exists which solves the corresponding DLP and thus the ECDLP in subexponential-time $L(q^k)$ according to (3.1). In general k is expected to be very large. However, for special curves such as supersingular curves the embedding degree k is comparably small. This attack is therefore successful if $n|(q^k-1)$ for some small value of k so that finding discrete logarithms in $\mathbb{F}_{q^k}^*$ is computationally feasible. This yields the Menezes-Okamoto-Vanstone condition in Section 1.1, i.e. that $n-1$ divided by the order of p modulo n is less than 10000. Note that on the other hand, a small value of k leads to another very productive branch of cryptography (see Section 7).

5.3 Anomalous curves attack

Let E be an elliptic curve defined over $\mathbb{F}_q$. This attack is successful for $\mathbb{F}_q$-anomalous elliptic curves, which are curves with a large p-subgroup. For instance, let $q = p$. Then elliptic curves with $\#E(\mathbb{F}_p) = p$ are $\mathbb{F}_p$-anomalous. By Hasse's theorem we have $\#E(\mathbb{F}_p) = p + 1 - t$, where t denotes the trace of E over $\mathbb{F}_p$. Thus $\#E(\mathbb{F}_p) = p$ is equivalent to $t = 1$. The idea of the attack in this case is in principle to lift the points Q and P to points $\tilde{Q}$ and $\tilde{P}$ on the lifted elliptic curve $\tilde{E}/\mathbb{Q}_p$, where $\mathbb{Q}_p$ denotes the field of p-adic numbers. One then uses the concept of formal p-adic elliptic logarithm on $\tilde{E}/\mathbb{Q}_p$ to solve the ECDLP altogether in polynomial time complexity in the input size. Consequently, one requires the domain parameters in 1.1 selected so that E over $\mathbb{F}_q$ is not $\mathbb{F}_q$-anomalous.

5.4 General index calculus attacks

We first sketch the generic index calculus technique in a finite abelian group G adapted to the additive setting. The following steps should be performed if terms such as "norm", "prime element", "smoothness" are meaningful and hopefully the group G can be generated by elements of small norm.

Let G be a finite abelian group and let $D_1 \in G$ of order n. We assume the group order and n are known. Given D_2 such that $D_2 = \ell D_1$, we wish to find $\ell \in [0, n-1]$. In particular, an index calculus attack to the ECDLP as in 2.1 can be formulated for $G = E(\mathbb{F}_{q^r})$, $D_1 = P$, and $D_2 = Q = \ell P$.

1. One selects a smoothness bound B and chooses a **factor base** $\mathcal{F}_B$ for the relation generation

$$\mathcal{F}_B = \{\mathcal{P}_1, \ldots, \mathcal{P}_s\} = \{ \text{ prime elements in } G \text{ of norm } \leq B\} \ .$$

2. Construct enough different relations and create the relation matrix $A = (a_{ij}) \in \mathbb{Z}_n^{s\times(s+5)}$. This could for instance be done as follows: Perform a pseudo-random walk à la Pollard or Teske in the group to find smooth group elements of the form

$$\alpha_i D_1 + \beta_i D_2 = \sum_{j=1}^{s} a_{ji}\mathcal{P}_j \quad (1 \leq i \leq s+5) \ ,$$

where the α_i's and β_i's are integers. Store A and the coefficients α_i, β_i.

3. Use linear algebra to determine an element $\gamma \in \ker(A)$, i.e. a solution $\gamma = (\gamma_1, \ldots, \gamma_{s+5})$ to $Ax = 0$.

4. If $\sum \beta_i\gamma_i \not\equiv 0 \pmod{n}$, then $\ell \equiv -(\sum \alpha_i\gamma_i)/(\sum \beta_i\gamma_i) \pmod{n}$.

5.5 Xedni attack

We first discuss the basic idea of the relevant version of an index calculus method (see [Mil86, Sil00]). Let E be an elliptic curve defined over $\mathbb{F}_p$, where p is a prime. The direct approach as in the previous section applied to $G = E(\mathbb{F}_p)$, $D_1 = P$, and $D_2 = Q$ is unsuccessful. An explanation is as follows: First one lifts the curve $E/\mathbb{F}_p$ to an elliptic curve $\mathcal{E}/\mathbb{Q}$. Then, one attempts to lift various points from $E/\mathbb{F}_p$ to $\mathcal{E}/\mathbb{Q}$. Finally, one uses relationships among those lifted points to recover the ECDLP. However, this method fails since lifting of the points is difficult and one needs many rational points of small height.

Silverman [Sil00] suggested to proceed conversely. For this reason he called his attack xedni attack. This specific attack has been subsequently analyzed in detail in [JKS$^+$00]. Interestingly, Koblitz pointed out that if Silverman's xedni algorithm were successful, then RSA could be attacked by an extension of the method as well. Silverman's idea was the following: First, one chooses points $P_1, \ldots, P_s$ in $E(\mathbb{F}_p)$ and lifts them to points $Q_1, \ldots, Q_s$ having integer coefficients. This can easily be accomplished. Then one chooses by linear algebra an elliptic curve $\mathcal{E}/\mathbb{Q}$ that goes through the lifted points $Q_1, \ldots, Q_s$. The whole point is that one hopes that the lifted curve $\mathcal{E}/\mathbb{Q}$ has smaller Mordell-Weil rank than expected. Silverman imposed an additional idea of Mestre to make this probability even higher, namely lift the curve $E/\mathbb{F}_p$ to $\mathcal{E}/\mathbb{Q}$ so that the reduced curve $E/\mathbb{F}_u$ for various small primes u satisfies $\#E(\mathbb{F}_u) \approx u + 1 - 2\sqrt{u}$. If there are nontrivial relations among the points $Q_1, \ldots, Q_s$, the ECDLP is solved. However, the analysis in [JKS$^+$00] shows that this attack fails, since mainly the absolute bound on the size of the coefficients of a relation satisfied by the lifted points is too small.

5.6 Semaev's index calculus attack

Even though this attack is not immediately successful, ideas of this attack led to important attacks on elliptic curves defined over finite field extensions (see Section 5.7 and Section 5.8). Again let E be an elliptic curve defined over $\mathbb{F}_p$ and let $\overline{\mathbb{F}}_p$ denote an algebraic closure of $\mathbb{F}_p$. Semaev suggested to use the so-called summation polynomials for the generation of the relation in the general index calculus algorithm. For integers $j \geq 2$ these are recursively defined symmetric polynomials $f_j \in \mathbb{F}_p[X_1, \ldots, X_j]$ with $f_2(X_1, X_2) := X_1 - X_2$ and $\deg_{X_i} f_j =$

2^{j-2}. The following important property holds true: For $(x_1, \ldots, x_j) \in \overline{\mathbb{F}}_p^j$ we have $f(x_1, \ldots, x_j) = 0$ if and only if there exists $(y_1, \ldots, y_j) \in \overline{\mathbb{F}}_p^j$ such that

$$P_1 + P_2 + \ldots + P_{j-1} + P_j = \mathcal{O}$$

and $P_i = (x_i, y_i) \in E(\overline{\mathbb{F}}_p)$.

As usual we now identify $\mathbb{F}_p$ with its set of representatives $\mathbb{Z}_p = \{a \in \mathbb{Z} \mid 0 \leq a < p\}$. Semaev's index calculus attack uses for a cyclic group $E(\mathbb{F}_p)$ and an integer $j \geq 2$ the factor base

$$\mathcal{F}_j = \{(x, y) \in E(\mathbb{F}_p) \,:\, 0 \leq x \leq p^{\frac{1}{j}}\} \ .$$

A relation is constructed by generating a random point $R = k_1 P + k_2 Q \in E(\mathbb{F}_p)$ and by then expressing $R = (x_R, y_R)$ as a sum of points in $\mathcal{F}_j$ by solving the multivariate polynomial congruence

$$f_{j+1}(x_1, \ldots, x_j, x_R) \equiv 0 \pmod{p} \quad \text{and} \quad x_1, \ldots, x_j \leq p^{\frac{1}{j}} \ .$$

If this is solvable, determine the corresponding y-coordinates $\pm y_i \in \mathbb{F}_p$ of points. If all $y_i \in \mathbb{F}_p$, then each $P_i = (x_i, y_i) \in \mathcal{F}_j$ and we have a relation

$$s_1 P_1 + s_2 P_2 + \ldots + s_j P_j = R = k_1 P + k_2 Q \quad (s_i = \pm 1) \ .$$

A detailed analysis shows that the approximate heuristic running time of Semaev's index calculus is

$$O(t_{j,p}\, j!\, p^{\frac{1}{j}} + p^{\frac{2}{j}}) \ ,$$

where $t_{j,p}$ be the expected heuristic running time for solving the multivariate polynomial congruence for small values of x_i. However, there is no indication why solving the multivariate polynomial congruence for small values of x_i should be easy and why this attack would work faster than exhaustive search.

5.7 Gaudry's index calculus attack

Now, let E be an elliptic curve defined over $\mathbb{F}_{q^r}$, where $r > 1$, and for simplicity let $E(\mathbb{F}_{q^r})$ be cyclic. Gaudry [Gau09] suggested to use the ideas of Semaev's index calculus method and especially the summation polynomials (see Section 5.6) in order to derive an attack that works for elliptic curves over $\mathbb{F}_{q^r}$ for certain values of r. Gaudry suggested to use the factor base

$$\mathcal{F} = \{(x, y) \in E(\mathbb{F}_{q^r}) \,:\, x \in \mathbb{F}_q\} \ .$$

For the relation generation step in the general index calculus algorithm one tries to express a random point $R = (x_R, y_R) \in E(\mathbb{F}_{q^r})$ as a sum of points in $\mathcal{F}$. As in Semaev's approach one uses the summation polynomials and finds solutions $(x_1, \ldots, x_r)$ to

$$f_{r+1}(x_1, \ldots, x_r, x_R) = 0 \quad \text{and} \quad x_i \in \mathbb{F}_q \ .$$

Now, let $\{\beta_1, \ldots, \beta_r\}$ be a basis for $\mathbb{F}_{q^r}$ over $\mathbb{F}_q$. One represents x_R and the coefficients of the equations of the elliptic curve in terms of this basis. Inserting this into the original equation for $f_{r+1}(x_1, \ldots, x_r, x_R)$ and equating coefficients

at β_i yields r equations in r variables $x_1, \ldots, x_r$. Eventually, this leads to the following system of polynomial equations

$$g_i(x_1, \ldots, x_r, x_R) = 0 \quad (1 \leq i \leq r) \ .$$

By using optimized algorithms and a so-called double large prime variant one obtains an expected running time of $O(q^{2-\frac{2}{r}})$ for r fixed and small. Comparing this with the expected complexity of Pollard's rho algorithm of $O(q^{\frac{r}{2}})$ we can conclude: Gaudry's algorithm for solving the ECDLP for an elliptic curve E defined over $\mathbb{F}_{q^3}$ or $\mathbb{F}_{q^4}$ is asymptotically faster than Pollard's rho algorithm. These techniques are independent of the arithmetic properties of the elliptic curves.

5.8 Diem's index calculus attack

Diem [Die11] generalized and confirmed Gaudry's results in various ways. We mention only certain aspects of his work. The principal idea goes back to Semaev's index calculus idea by making use of the summation polynomials (see Section 5.6). Let E be an elliptic curve defined over a finite field extension $\mathbb{F}_{q^r}$. In the first step, a factor base has to be selected. Let e be an integer with $e \geq 3$ and put $v := \lceil r/e \rceil$. Randomly select v linear independent elements $\alpha_1, \ldots, \alpha_v \in \mathbb{F}_{q^r}$ and define the subspace $F_v = \langle \alpha_1, \ldots, \alpha_v \rangle$ of dimension v. The factor base is defined as

$$\mathcal{F} = \{(x, y) \in E(\mathbb{F}_{q^r}) \ : \ x \in F_v\} \ .$$

The relation generation is performed as follows: For a random $R \in E(\mathbb{F}_{q^r})$, solve

$$f_{e+1}(x_1, \ldots, x_e, x_R) = 0 \quad \text{and} \quad x_i \in F_v \ .$$

Reformulation with respect to a basis $\{\beta_1, \ldots, \beta_r\}$ for $\mathbb{F}_{q^r}$ over $\mathbb{F}_q$, inserting everything in the polynomial equation and equating coefficients yields r polynomial equations in ev variables.

We mention an important consequence of Diem's analysis. Let a, b be real numbers such that $0 < a < b$. Then Diem's algorithm has an expected subexponential running time $L_{q^r}[\frac{3}{4}, c]$ for $a \log q \leq r \leq b \log q$ and $e \sim \sqrt{\log q}$, where $c = c(a, b)$ is a constant. Notice that Diem's results are remarkable since he proved that a subexponential-time algorithm exists for solving the ECDLP of a certain small class of elliptic curves over finite field extensions and the attack is independent of the arithmetic properties of the curves.

5.9 Weil descent attack

Let E be an elliptic curve defined over $\mathbb{F}_{q^r}$, where $r > 1$. This attack is successful for some elliptic curves with special arithmetic properties. We refer to [Hes05, CFA+05, GS99, GHS02b] for details on the theoretical results. An explicit realization of the attack as well as recent computational examples with running times over $\mathbb{F}_{2^{124}}$ and $\mathbb{F}_{2^{155}}$ can be found in [JMS04, VJS11].

The idea of the attack is to first embed E into the Weil restriction of scalars $\mathcal{W}_E(\mathbb{F}_q)$ over the smaller field $\mathbb{F}_q$. One then tries to find a curve $\mathcal{X} \subseteq \mathcal{W}_E$ with good properties and one constructs an efficiently computable group homomorphism $\Phi : E(\mathbb{F}_{q^r}) \longrightarrow J_{\mathcal{X}}(\mathbb{F}_q)$ such that $\#\ker(\Phi) \leq \lambda$. Then one computes

$D_1 = \Phi(P)$ and $D_2 = \Phi(Q)$. At this point, one has reduced the ECDLP of $E/\mathbb{F}_{q^r}$ for points P and $Q = \ell P$ to the discrete logarithm problem in the Jacobian of a higher genus curve $\mathcal{X}$ over $\mathbb{F}_q$; that is, one solves $D_2 = \ell D_1$ on $J_{\mathcal{X}}(\mathbb{F}_q)$ by known index calculus methods as in 5.4 if the key size is appropriate.

This method works for a small but significant part of all elliptic curves over $\mathbb{F}_{q^r}$. As a consequence of this result, the results in Section 5.7 as well as Section 5.8, and other results for elliptic curves E defined over finite field $\mathbb{F}_{q^r}$, one recommends to use domain parameters (see (1.1)) such that E is an elliptic curve defined over a finite field $\mathbb{F}_q$ and either q is a prime or $q = 2^{p'}$, where p' is a prime.

6 Parameter selections

6.1 Elliptic curves for security applications

In this section we focus on the selection of elliptic curves for security applications, such as digital signatures, MRTDs, or government use. The requirements in the low-cost area (e.g. RFIDs for copyright protection) or for applications of pairing-based cryptography may differ. There are several aspects that have to be taken into account when selecting elliptic curves for security applications. The explicit selection of the parameters has been listed in Section 1. We refer to [LM10, HMV04, CFA$^+$05] for details.

Implementation issues. Several choices of curve parameters may ease the implementation of elliptic curves and prevent implementation errors.

Resistance against side channel attacks. There are side channel attacks that make use of the existence of curve points with distinguished properties. For instance, it is easier to attack curves with a $\mathbb{F}_p$-rational point whose $x-$ or $y-$coordinate equals zero. In short Weierstrass form (1.2) these conditions translate to: b should be a quadratic nonresidue modulo p and there should be no point of order 2 in the subgroup of E under consideration. For a comprehensive overview of implementational issues we refer to [KLL$^+$11] and Section 8.

Appropriate key length. When used in a hybrid scheme the asymmetric key length should be roughly twice the underlying symmetric key length. However, it is considered appropriate to use 384-bit curves in conjunction with 256-bit symmetric algorithms. For signature applications that are compliant with the German digital signature law, the German Federal Network Agency publishes algorithms and parameter lengths that are considered secure for at least six years. Their catalogue is updated annually and the current version considers a group order of magnitude $\sim 2^{224}$ as secure for qualified electronic signatures until 2015, and $\sim 2^{250}$ until 2018. There is no restriction on the size of the base field.

German identity cards and MRTDs use elliptic curves as well. For identification purposes two different curves are used. The country signing certificate authority uses the elliptic curve brainpoolP384r1, whereas for document signing the curve brainpoolP256r1 is used.

Representation of the curve and choice of the arithmetic. There are many popular representations of elliptic curves that allow fast or secure implementations. To implement the multiplication with scalars also many methods exist, e.g. double-and-add or use of the non-adjacent form (NAF). On the other

hand methods against SCA involve randomization and may slow down the arithmetic. One may benefit from curve representations with uniform addition and doubling formulas, such as curves in Edwards form. Note that for curves in short Weierstrass form the standard formulas for doubling are different form the formulas for adding. It can also be helpful to implement scalar multiplication by using the Montgomery ladder which makes doubling and adding of points indistinguishable. It is still of interest to implement algorithms in a way that makes timing behaviour and power consumption independent of the data processed.

Quantum computers. On (hypothetical) quantum computers Shor's algorithm allows to easily solve the ECDLP. One countermeasure is to keep the curve equation secret. As for curves over prime fields three curve points suffice to find the characteristic of the base field and the equation of the curve. As a consequence one would also have to keep curve points secret. In the case of an unknown curve equation, this can be achieved by point-compression methods that also may help to save on bandwidth.

6.2 Generation of suitable elliptic curves

There are two different approaches to generate suitable curves for security applications.

- The CM method uses class field theory to construct curves with a prescribed number of points whose endomorphism ring has a relatively small class number. There are recent variations which allow to generate curves with relatively large class number. Interestingly the same ideas are used in Atkin's deterministic primality proving algorithm.
- Point counting. By using the Schoof-Elkies-Atkin algorithm (SEA) the number of points of elliptic curves over $\mathbb{F}_q$ in the cryptographic range can easily be computed. Computer Algebra systems such as Magma [BCP97] offer quick implementations. Interestingly, point counting on elliptic curves over the Ring $\mathbb{Z}/N\mathbb{Z}$ and factoring N are equivalent

When constructing an elliptic curve one starts with the choice of a base field. A popular choice are prime fields with pseudo Mersenne characteristic. Such fields offer fast arithmetic which can in turn be used to develop fast implementations. This approach has been taken by NIST. When following this approach one has to carefully consider the patent situation regarding fast arithmetic. One should also consider that the use of special primes may make implementations more vulnerable to side-channel attacks. Another approach is to choose prime fields as well as curve equations pseudo randomly and deterministically.

7 Pairing-based cryptography

The Weil pairing on supersingular elliptic curves was the first pairing to occur in cryptography, and its use has been of a destructive nature as discussed in Section 5.2. This attack lowered the efficiency and security of supersingular elliptic curves to a level comparable to RSA. As a consequence, the general advantages

of elliptic curves over RSA were lost in this special case and supersingular elliptic curves were banned from further cryptographic research for about a decade. Only around 2000 it was then observed that revolutionary new cryptographic primitives could be realized using supersingular elliptic curves and the Weil pairing. This marked the start of pairing-based cryptography.

7.1 Pairing-based cryptographic protocols

Pairing-based cryptographic protocols have taken a central position in cryptographic research in the past twelve years. These protocols are based on classical discrete logarithm based protocols and use a bilinear, non-degenerate map

$$e : G_1 \times G_2 \to G_3$$

of cyclic groups G_1, G_2 and G_3 with prime order n as a key additional feature. It is customary to write the groups laws of G_1, G_2 and G_3 multiplicatively, while in practice G_1 and G_2 will be subgroups of some $E(\mathbb{F}_{q^k})$ with additive group law and G_3 a subgroup of $\mathbb{F}_{q^k}^*$ with multiplicative group law. Bilinearity thus means $e(xy, u) = e(x, u)e(y, u)$ and $e(x, uv) = e(x, u)e(x, v)$ for all $x, y \in G_1$ and $u, v \in G_2$. Non-degeneracy means that there are $x \in G_1$ and $u \in G_2$ with $e(x, u) \neq 1$.

The papers [BF01, Jou00, SOK00] on identity based cryptography and tripartite key exchange have been the starting point for pairing-based cryptography. Much attention was in particular payed to [BF01], which solved an open research problem posed in [Sha85] back in 1984. Since the publication of these three papers a very large number of applications of pairings in cryptography have been exhibited which exceed by far simple encryption or signature protocols. An early survey can be found in [Pat05a].

In the rest of this subsection the protocols from [BF01] and [BLS01] are described in a simplified way to give the reader a rough impression how pairings are used in cryptography and what security notions are put in place.

Identity-based cryptography. Identity-based cryptography and in particular identity-based encryption offer an alternative approach to a traditional public key infrastructure. The idea is that public keys can be arbitrarily prescribed instead of being derived from a secret. For example, public keys could be email addresses, and the corresponding private key would be derived from the public key by a trustworthy third party, called trust center. There is then no need for Bob to obtain the public key of communication partner Alice from Alice herself, but Bob can immediately encrypt messages for Alice using the email address of Alice. On the other hand, Alice can only decrypt when she has queried the trust center for her private key. The trust center is assumed to ensure that private keys are only handed out to the corresponding eligible person. A man-in-the-middle or impersonation attack on Alice is then not possible per assumption on the trust center. Note also that the trust center has access to all data encrypted with Alice's identity. In summary, identity-based cryptography offers some interesting advantages, but also disadvantages over a traditional public key infrastructure. A discussion of relevant aspects in view of practical employment can for example be found in [Pat05b].

We now describe the basic protocol from [BF01]. Assume $G = G_1 = G_2$ and consider a pairing $e : G \times G \to G_3$ and let $g \in G$ be a generator of

G. Each person X has an associated identity string ID_X. Furthermore, let $H_{ID} : \{0,1\}^* \to G$ and $H : G_3 \to \{0,1\}^s$ be two cryptographic hash functions, where $s \approx \log_2(n)$. If $m_1, m_2 \in \{0,1\}^s$ then $m_1 \oplus m_2$ denotes the sum of m_1 and m_2 as elements of $\mathbb{F}_2^s$ (equivalently, $m_1 \oplus m_2$ is the bitwise xor of m_1 and m_2). The trust center is denoted by T. The key generation for identity-based cryptography proceeds as follows: The trust center T chooses $x \in \mathbb{Z}/n\mathbb{Z}$ uniformly at random and computes $g_{pub} = g^x$. The public key of T is g, g_{pub}. The private (secret) key of T is x. In addition, T computes $y_A = H_{ID}(ID_A)$ and $s_A = y_A^x$. The public key of Alice is y_A. The private key of Alice is s_A. In the identity-based encryption protocol from [BF01], Bob computes the cipher text $(u, v) = \mathcal{E}(g, g_{pub}, y_A, m)$ as follows: The plain text m is encoded as $w \in \{0,1\}^s$. Then $u = g^r$ and $v = w \oplus H(e(y_A, g_{pub}^r))$ are computed for $r \in \mathbb{Z}/n\mathbb{Z}$ chosen uniformly at random. The cipher text is (u, v). To decrypt (u, v), Alice computes the plain text $m = \mathcal{D}(s_A, (u, v))$ via $w = v \oplus H(e(s_A, u))$ and decodes w to the plain text m.

The standard security model for identity-based encryption is as follows: The attacker has access to the secret keys of identities different from the target identity and the attacker can also obtain decryptions of arbitrary cipher texts under the target identity by querying an oracle. The attacker is first required to output two different target plain texts and a target identity. The attacker is then given the encryption of one of the target plain texts, chosen uniformly at random, under the target identity. Finally, the attacker is deemed successful, if he can guess the corresponding target plain text with probability significantly different from $1/2$ without querying for the decryption of the given encryption (we leave a precise definition of "significantly" open).

An identity-based encryption protocol is called IND-ID-CCA secure if there is no successful attacker in the above sense. The application of a variant of the Fujisaki-Okamoto transformation to the above basic identity-based encryption protocol yields an improved identity-based encryption protocol that is IND-ID-CCA secure under the following assumptions: The attackers are supposed to work independently of randomly chosen cryptographic hash functions H_{ID} and H (the so called random oracle model) and the bilinear (computational) Diffie-Hellman problem (BDH) is hard. This basic computational problem is: Given uniformly at random chosen g, g^a, g^b, g^c in G compute $e(g, g)^{abc}$. The BDH can easily be reduced to the CDH in G_3 or the CDH in G.

Short deterministic signatures. The signature scheme from [BLS01] uses a cryptographic hash function $H : \{0,1\}^* \to G$. The key generation is the same as in classical discrete logarithm based systems. The signer chooses $x \in \mathbb{Z}/n\mathbb{Z}$ uniformly at random and computes $y = g^x$. The public key is y, the private key x. For a signature computation $\sigma = \mathcal{S}(x, m)$ with the message m the signer computes $\sigma = H(m)^x$. The signature is (m, σ). For a signature verification $b = \mathcal{V}(y, m, \sigma)$ the verifier first computes $v = e(g, \sigma)$ and then $v' = e(y, H(m))$. If $v' = v$ then $b = 1$ and the signature is accepted by the verifier, and if $v' \neq v$ then $b = 0$ and the signature is rejected by the verifier.

The security model for signature schemes considers attackers which can obtain signatures for arbitrary messages of their choice by querying an oracle. The attacker is deemed successful if he can compute a message and a valid signature for the message without having queried for the signature of this message. If there is no such attacker then the signature scheme is called secure with respect

to existential forgery.

The signature scheme from [BLS01] is secure with respect to existential forgery in the random oracle model if the CDH in G is hard. Incidentally, the DDH is easy in this case: The verification step of the signature scheme actually uses the pairing to check whether $(g, g^x, H(m), \sigma)$ is a Diffie-Hellman tupel. In comparison to previous discrete logarithm based signature schemes such as ECDSA this signature scheme is deterministic and requires only about half of the bandwith of the previous signature schemes.

Choice of pairings. Suitable pairings are the Weil and Tate-Lichtenbaum pairings and modifications of these pairings for very carefully chosen elliptic (or hyperelliptic) curves. There are no pairings in other mathematical contexts known which would appear to be both efficiently computable and secure, and it is an interesting open problem to find such pairings. In practice G, G_1, G_2 are thus point groups of elliptic curves (or Picard groups of hyperelliptic curves), and G_3 is a subgroup of the multiplicative group of a finite field.

In order to balance and thus optimize security and efficiency, the groups G, G_1, G_2 and G_3 respectively need to be chosen such that the DLP has roughly the same complexity in each of these groups.

Protocols often require further properties of pairings. Cryptographers usually classify pairings in three types, see [GPS08, CHM10]. Pairings of type 1 come with an isomorphism $G_1 \to G_2$ that can be efficiently computed in either direction. In this situation $G_1 = G_2$ can be assumed. Pairings of type 2 come with a one-way isomorphism $G_2 \to G_1$, and for pairings of type 3 there is no efficiently computable isomorphism from $G_1 \to G_2$ or $G_2 \to G_1$. Some further remarks about the mathematical realization of such types of pairings are be made below.

7.2 Weil pairing and Tate-Lichtenbaum pairing

We now focus on the mathematics behind the pairings. In the following let E denote an elliptic curve over $\mathbb{F}_q$. We assume that its cardinality $\#E(\mathbb{F}_q)$ has a sufficiently large prime divisor $n \neq q$. Let $k \in \mathbb{Z}^{\geq 1}$ be minimal with $n|(q^k - 1)$. We assume in addition, that $k \geq 2$ and that $q^k - 1$ is divisible by n but not divisible by n^2. The number k is called the embedding degree of E.

Under these assumptions we have that $E(\mathbb{F}_{q^k})[n] \cong \mathbb{Z}/n\mathbb{Z} \times \mathbb{Z}/n\mathbb{Z}$ and that the group of n-th roots of unity μ_n is contained in $\mathbb{F}_{q^k}$.

The Tate-Lichtenbaum pairing

$$\langle \cdot, \cdot \rangle_n : E(\mathbb{F}_{q^k})[n] \times E(\mathbb{F}_{q^k})/nE(\mathbb{F}_{q^k}) \to \mathbb{F}_{q^k}^*/(\mathbb{F}_{q^k}^*)^n$$

is defined as follows. For every $P \in E(\mathbb{F}_{q^k})$ and $j \in \mathbb{Z}$ let $f_{j,P} \in \mathbb{F}_{q^k}(E)$ denote a rational function on E with divisor $(f_{j,P}) = j((P) - (\mathcal{O})) - ((jP) - (\mathcal{O}))$, where (P) is the prime divisor on E defined by P. The function $f_{j,P}$ is defined only up to non zero multiples from $\mathbb{F}_{q^k}$ (see [Gal05, Sil09, Sti08] for more details on rational functions, divisors and the theorem of Riemann-Roch on curves, and the example on page 24 how such rational functions can look like). Now let $P \in E(\mathbb{F}_{q^k})[n]$ and $Q \in E(\mathbb{F}_{q^k})$. We choose $R \in E(\mathbb{F}_{q^k})$ with $\{Q+R, R\} \cap \{P, \mathcal{O}\} = \emptyset$ and define

$$\langle P, Q + nE(\mathbb{F}_{q^k}) \rangle_n := f_{n,P}(Q + R) \cdot f_{n,P}(R)^{-1} \cdot (\mathbb{F}_{q^k}^*)^n.$$

With the help of Weil reciprocity one can easily show that $\langle \cdot, \cdot \rangle_n$ is well-defined [Gal05, Hes04].

In applications one usually uses the reduced Tate-Lichtenbaum pairing for simplification reasons. The reduced Tate-Lichtenbaum pairing

$$t_n : E(\mathbb{F}_{q^k})[n] \times E(\mathbb{F}_{q^k})[n] \to \mu_n$$

is defined by

$$t_n(P,Q) = \langle P, Q + nE(\mathbb{F}_{q^k}) \rangle_n^{(q^k-1)/n} = \left(f_{n,P}(Q+R) \cdot f_{n,P}(R)^{-1} \right)^{(q^k-1)/n}.$$

The Tate-Lichtenbaum pairing and the reduced Tate-Lichtenbaum pairing are bilinear and non-degenerate [FR94, Hes04]. For an explicit example of a variant of the reduced Tate pairing see the example on page 24 below.

The Weil pairing

$$e_n : E(\mathbb{F}_{q^k})[n] \times E(\mathbb{F}_{q^k})[n] \to \mu_n$$

can be defined in a fashion similar to the Tate-Lichtenbaum pairing, but using four instead of two function evaluations [Gal05], [Sil09, Exerc. 3.16]. A shorter and computationally more efficient description is the following [Mil04]: For $P, Q \in E(\mathbb{F}_{q^k})[n]$ choose $f_{n,P}$ and $f_{n,Q}$ with $(f_{n,P} \cdot f_{n,Q}^{-1})(\mathcal{O}) = 1$. Then

$$e_n(P,Q) = \begin{cases} 1 & \text{for } P = Q,\ P = \mathcal{O},\ Q = \mathcal{O}, \\ (-1)^n f_{n,P}(Q) \cdot f_{n,Q}(P)^{-1} & \text{else.} \end{cases}$$

The Weil pairing e_n as above is bilinear and non-degenerate since this holds true over the algebraic closure $\bar{\mathbb{F}}_q$ and here $E(\bar{\mathbb{F}}_q)[n] = E(\mathbb{F}_{q^k})[n]$. It is furthermore alternating, i. e. $e_n(P,P) = 1$, which is for example useful for tests whether a given point lies in a given cyclic subgroup. The relation between the Weil and Tate-Lichtenbaum pairings is

$$e_n(P,Q)^{(q^k-1)/n} = t_n(P,Q) \cdot t_n(Q,P)^{-1}.$$

Types 1-3. The application of these pairings in cryptography requires the choice of suitable subgroups of $E(\mathbb{F}_{q^k})$ which is discussed now.

The Frobenius endomorphism π_q of E is given by $(x,y) \mapsto (x^q, y^q)$. It induces an automorphism of the two-dimensional $\mathbb{F}_n$-vector space $E(\mathbb{F}_{q^k})[n]$. Let $P \in E(\mathbb{F}_q)[n]$ be a point of order n. Since $\pi_q(P) = P$ we have that P is an eigenvector of π_q for the eigenvalue 1. The characteristic polynomial of π_q is of the form $x^2 - tx + q$ and has the roots 1 and q modulo n. Hence there is another point $Q \in E(\mathbb{F}_{q^k})$ of order n, such that Q is an eigenvector of π_q for the eigenvalue q, or equivalently such that $\pi_q(Q) = qQ$ holds. In summary we have $E(\mathbb{F}_{q^k})[n] = \langle P \rangle \times \langle Q \rangle$.

For these subgroups $t_n(P,P) = t_n(Q,Q) = 1$ and $t_n(P,Q) \neq 1$, similarly for the Weil pairing. The endomorphism $\mathrm{Tr} = c \sum_{i=0}^{k-1} \pi_q^i$ with $kc \equiv 1 \pmod{n}$ defines a surjective projection $\langle P \rangle \times \langle Q \rangle \to \langle P \rangle$ with kernel $\langle Q \rangle$, the trace zero subgroup.

A distortion map for $T = \lambda P + \mu Q \neq \mathcal{O}$ is an endomorphism ψ of E with $\psi(T) \notin \langle T \rangle$. If λ and μ are not zero, then Tr is a distortion map for T. It can

be shown that there is a distortion map for $T = P$ and $T = Q$ if and only if E is supersingular [Ver04, GR04].

Using these mathematical objects the three types of pairings described at the end of subsection 7.1 can be realized. Type 1 uses supersingular elliptic curves, a distortion map ψ and $Q = \psi(P)$. Thus $G_1 = G_2 = \langle P \rangle$ holds true. Type 2 uses ordinary elliptic curves, $G_1 = \langle P \rangle$ and $G_2 = \langle \lambda P + \mu Q \rangle$ with $\lambda, \mu \neq 0$. Here $G_1 \neq G_2$ and Tr yields a one-way isomorphism $G_2 \to G_1$. Type 3 uses ordinary elliptic curves with $G_1 = \langle P \rangle$ and $G_2 = \langle Q \rangle$. Thus $G_1 \neq G_2$ and there is (as far as one knows) no efficient computable isomorphism between G_1 and G_2. We refer to [GPS08, CHM10] for more details.

Choice of parameters. The embedding degree is the parameter that is most important for security and efficiency. The complexities of the DLP in $E(\mathbb{F}_q)$ and $(\mathbb{F}_{q^k})^*$ are roughly $\exp(1/2 \log q)$ and $\exp((k \log q)^{1/3})$ respectively, the latter similar to (3.1). To balance these complexities for growing q we need to choose $k \approx (\log q)^{2/3}$. The embedding degree thus grows with q. Table 1 gives an overview over the bit-lengths of the keys for symmetric cryptosystems, the value $\log_2(q)$ for elliptic curves, the value $\log_2(N)$ for the RSA cryptosystem or the value $\log_2(q^k)$ for $\mathbb{F}_{q^k}^*$ respectively, and the corresponding embedding degree for rows of comparable security.

Symm	ECC	RSA	k
80	160	1024	6
128	256	3072	12
256	512	15360	30

Table 1: Comparison for bit lengths at roughly equal security

Construction. Supersingular curves yield embedding degrees $k \in \{2, 3, 4, 6\}$ only [MOV93]. In view of Table 1 larger embedding degrees k are highly desirable. This requires the construction of suitable ordinary elliptic curves. The following conditions on q, n, $t = q + 1 - \#E(\mathbb{F}_q)$ and k, called MNT conditions, have to be observed (ϕ_k is the k-th cyclotomic polynomial):

1. $q + 1 - t = cn$.
2. $\phi_k(q) \equiv 0 \pmod{n}$.
3. q is a prime power, n is a prime, $|t| \leq 2\sqrt{q}$.
4. $4q - t^2 = Df^2$ with D small.
5. $\rho = \log(q)/\log(n) \approx 1$.

Condition 2 implies $n|(q^k - 1)$. Condition 3 is required to enable the efficient computation of the elliptic curve by the theory of complex multiplication. Condition 5 means that c from Condition 1 is about as small as possible.

Solutions to the conditions 1-5 for arbitrary k can be found rather easily by a search strategy of Cox and Pinch, if one allows $\rho \approx 2$, see [Gal05]. But the

resulting cryptosystems will be rather inefficient. On the other hand it can be shown that solutions with $\rho \approx 1$ are rare [ULS12] and accordingly difficult to find.

Constructions of ordinary elliptic curves with $\rho = 1$ have been found for the embedding degrees $k \in \{3, 4, 6\}$, $k = 10$ and $k = 12$, see [MNT01, BN06, FST10]. For constructions with $1 < \rho < 2$ see for example [BS08, BW05, CLN11, DCC06, GMV07]. For given k solutions to the conditions 1-5 can often be given in parametrized form $q = q(z)$ and $n = n(z)$ for $z \in \mathbb{Z}$. The methodology for these constructions makes partial use of algebraic number theory and diophantine geometry.

As an example consider the particularly nice elliptic curves from [BN06] with embedding degree 12 and $\rho = 1$, which are very well suited for 128 bit security. Let

$$\begin{aligned} p(z) &= 36z^4 + 36z^3 + 24z^2 + 6z + 1, \\ t(z) &= 6z^2 + 1, \\ n(z) &= p(z) + 1 - t(z). \end{aligned}$$

Then $\phi_{12}(p(z)) \equiv 0 \pmod{n(z)}$ and $4p(z) - t(z)^2 = 3(6z^2 + 4z + 1)^2$. The construction of the corresponding elliptic curves is as follows:

Algorithm.

1. Find $x \in \mathbb{Z}$ such that $p(x)$ and $n(x)$ are primes.
2. Choose an elliptic curve $E : y^2 = x^3 + b$ with $b \in \mathbb{F}_p$ uniformly at random.
3. If $\#E(\mathbb{F}_p) = n(x)$ then output E. Otherwise repeat from step 2.

At least at first sight this is an amazingly simple construction: The computed curves E satisfy automatically the conditions 1-5 for $k = 12$. Furthermore, the construction of E via the complex multiplication method is not necessary and the check in step 3 will be successful after expected 6 choices of E, so that the construction is also very efficient. Is is also possible to prove these statements formally. The explicit construction of such an elliptic curve is most conveniently done using a computer algebra system such as Magma [BCP97].

Efficient computation. It is not possible to discuss all the details of the efficient computation of the Weil- and Tate-Lichtenbaum pairings here. Instead we wish to focus on some more conceptual aspects regarding the Tate-Lichtenbaum pairing.

The efficient computation of the Tate-Lichtenbaum pairing has been investigated in many publications, among the first [GHS02a, BKLS02, BGOS07]. The basic and essential building block, the Ate pairing, for the most efficient pairings today has been introduced in [HSV06]: A particularly efficient recent family of such pairings for embedding degree 24, $\rho = 1.25$ and high security levels has been investigated in [CLN11].

We consider the reduced Tate-Lichtenbaum pairing t_n and the generators P and Q of the eigenspaces G_1 and G_2 as above. Then it can be shown that t_n restricted to $G_1 \times G_2$ or $G_2 \times G_1$ respectively can already be described by $t_n(P, Q) = f_{n,P}(Q)^{(q^k-1)/n}$ and $t_n(Q, P) = f_{n,Q}(P)^{(q^k-1)/n}$ respectively if

$f_{n,P}$ and $f_{n,Q}$ are chosen among all possible scalar multiples in a fixed suitably normalized form. This means that only one function evaluation is necessary. Recall that $f_{n,P}$ and $f_{n,Q}$ have been defined above as very specific rational functions on E. An example is given below.

But it is in fact possible to give a further significant simplification of the pairing is restricted to $G_2 \times G_1$. This yields a new, bilinear and non-degenerate pairing, called Ate pairing in [HSV06], with a significantly simplified defining rational function. Let $T = t-1$, where $\#E(\mathbb{F}_q) = q+1-t$, and suppose $T^k \not\equiv 1 \pmod{n^2}$. The Ate pairing

$$\hat{t}_n : G_2 \times G_1 \to \mu_n$$

is defined by

$$\hat{t}_n(Q,P) = f_{T,Q}(P)^{(q^k-1)/n}.$$

The main point here is that the function $f_{T,Q}$ has degree about $|T|$, which roughly lies between $n^{1/\varphi(k)}$ and $q^{1/2}$. On the other hand, the degree of the functions $f_{n,P}$ and $f_{n,Q}$ of the Tate-Lichtenbaum pairing is about $n \approx q$. This implies a drastic improvement in terms of efficiency for the Ate pairing, in particular for low absolute values of T.

Example. Let $E : y^2 = x^3 + 4$ over $\mathbb{F}_q$ with $q = p = 41761713112311845269$, $n = 715827883$, $k = 31$ and $T = -2$. Then

$$\begin{aligned}\hat{t}_n &: G_2 \times G_1 \to \mu_n,\\ &(Q,P) \mapsto \big(y_P - 3x_Q^2/(2y_Q)x_P - (-x_Q^3+8)/(2y_Q)\big)^{(q^k-1)/n}\end{aligned}$$

defines a non-degenerate bilinear pairing, where $P = (x_P, y_P)$ and $Q = (x_Q, y_Q)$.

Combinations of t_n and $\hat{t}_n$ together with lattice techniques yield pairings with defining rational functions of degree always about equal to $n^{1/\varphi(k)}$, even if T has large absolute value. Pairing functions of such degrees appear to be best possible or at least close to best possible [Hes08, Ver10].

Using the theory of twists the memory requirements for elements from G_2 can be decreased and efficiency be increased. An elliptic curve E' over $\mathbb{F}_q$ is called twist of degree d of E, if there is an isomorphism $\psi : E' \to E$, defined over $\mathbb{F}_{q^d}$ with d minimal. If E is ordinary, $k = ed$ and if E has a twist over $\mathbb{F}_{q^e}$ of degree $d > 1$, then E' and ψ can be chosen such that $E'(\mathbb{F}_{q^e})[n] = \langle \psi^{-1}(Q) \rangle$ holds. We let $Q' = \psi^{-1}(Q)$, $G_2' = \langle Q' \rangle$ and obtain the modified Ate pairing

$$\hat{t}_n' : G_2' \times G_1 \to \mu_n$$

via

$$\hat{t}_n'(Q', P) = \hat{t}_n(\psi(Q'), P).$$

In the case of the elliptic curves discussed above with $k = 12$ from [BN06] we get the following: We have $E : y^2 = x^3 + b$ with $b \in \mathbb{F}_p$ and $p \equiv 1 \pmod 6$. Let $\lambda \in \mathbb{F}_{p^2} \backslash (\mathbb{F}_{p^2})^3$ and $\mu \in \mathbb{F}_{p^2} \backslash (\mathbb{F}_{p^2})^2$. The curve $E' : \mu y^2 = \lambda x^3 + b$ is a twist of E of degree 6 and $\psi : E' \to E$, $\psi(x,y) = (\lambda^{1/3}x, \mu^{1/2}y)$ is the corresponding isomorphism. In the example the degree of $f_{T,Q}$ is about $n^{1/2}$, and using Q' and E' over $\mathbb{F}_{p^2}$ instead of Q and E over $\mathbb{F}_{p^{12}}$ yields an improvement by a factor of 6 in terms of bandwidth.

Finally, it is instructive to interpret the Ate pairing in a broader mathematical context from number theory. It is well known that the Tate-Lichtenbaum pairing gives an algebraic description of the Artin symbol on unramified abelian extensions of exponent n of a global function field containing the n-th roots of unity. As it turns out, the Ate pairing gives an algebraic description of the Artin symbol in the general case where the global function field is not required to contain the n-th roots of unity.

8 Side channel attacks

In recent years side channel analysis (SCA) of physical devices implementing asymmetric and symmetric cryptographic algorithms and protocols and operating on secret data have become a very active area of research, both mathematically and technologically. SCA uses passive and active attacks by exploiting the intended interface of a cryptographic device. This section contains a very important example for actual realization problems which is contrast to the mere theoretical descriptions of the protocols in Section 2, 3, and 5.

Side channel cryptanalysis uses physical observables resulting from internal states and processes of a cryptographic computation as additional source for cryptanalysis. The outcome of the measurement of physical observables are real-valued vectors.

Internal state changes of the cryptographic device including the state change caused by operations with secret or ephemeral keys cause instantaneous leakage, that can be exploited. Examples of information sources are: Varying execution times of operations, varying power consumption during operation, varying electro-magnetic emanation during operation, enforced unexpected behaviour as consequence of induced transient or permanent device faults, enforced error messages of a cryptographic product, or photon emissions.

The exploitation of the above mentioned information sources leads to interesting new mathematical questions and results. Many of these can be formulated and solved as lattice problems in number theory. The use of stochastic modeling in the interpretation of measurements can not be underestimated. Countermeasures include avoiding key dependent power profiles and timing behaviour by uniformizing and randomizing computations. A full overview is given in [KLL$^+$11].

Example. The Nguyen-Shparlinsky attack. We refer to [NS03, BV96] for further details. Suppose that during the generation of ECDSA signatures some bits of the ephemeral key are leaked for many signatures. Then the secret user key can easily be recovered. The Nguyen-Shparlinski attack works when three bits are leaked for each of about a hundred signatures and consists of a reduction of the ECDSA-problem to the so called Hidden Number Problem (HNP) introduced by Boneh and Venkatesan. In this case the recovery of the secret key can be accomplished via lattice methods.

Remark. Note that in real world implementations modified versions of the ECDSA scheme are being used. We mention three main differences:

- Depending on the bitlength of the chosen curve hash values are truncated.
- Smart cards may be fed with hash values by an external source or only perform some rounds of the hash calculation internally.

- Blinding measures may lead to the use of modified ephemeral keys which are longer than the original ephemeral keys. Typically k_e is replaced by $k_e + \lambda \cdot n$, where n is the order of the cyclic subgroup under consideration and λ is a small random number. This increases the attack complexity but does not prevent the lattice attack.

This implies that security proofs for the pure scheme may no longer hold for real world implementations.

Acknowledgements The authors wish to thank several colleagues for carefully proofreading several versions of this paper. We also wish to thank Gabriele Nebe for her patience with this project and for making valuable suggestions.

References

[BCP97] W. Bosma, J. Cannon, and C. Playoust. The Magma algebra system I: The user language. *J. Symbolic Comp.*, 24, 3/4:235–265, 1997.

[BF01] D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. In J. Kilian, editor, *Advances in Cryptology - CRYPTO 2001*, LNCS 2139, pages 213–229. Springer, 2001.

[BFK09] J. Bender, M. Fischlin, and D. Kügler. Security Analysis of the PACE Key-Agreement Protocol. In *Proceedings of the 12th international Conference on information Security (Pisa, Italy, September 07 - 09, 2009)*, volume 5735 of *Lecture Notes in Computer Science*, pages 33–48. Springer, 2009.

[BGOS07] P. Barreto, S. Galbraith, C. O'hEigeartaigh, and M. Scott. Efficient pairing computation on supersingular abelian varieties. *Designs, Codes and Cryptography*, 42(3):239–271, 2007.

[BJN00] D. Boneh, A. Joux, and P. Q. Nguyen. Why textbook ElGamal and RSA encryption are insecure. In *ASIACRYPT '00: Proceedings of the 6th International Conference on the Theory and Application of Cryptology and Information Security*, pages 30–43. Springer, 2000.

[BKLS02] P. Barreto, H. Kim, B. Lynn, and M. Scott. Efficient algorithms for pairing-based cryptosystems. In M. Yung, editor, *Advances in Cryptology - CRYPTO 2002*, LNCS 2442, pages 354–369, Santa Barbara, 2002. Springer.

[BLS01] D. Boneh, B. Lynn, and H. Shacham. Short signatures from the Weil pairing. In C. Boyd, editor, *Advances in Cryptology - ASIACRYPT 2001*, LNCS 2248, pages 514–532, Gold Coast, Australia, 2001. Springer.

[BN06] P. Barreto and M. Naehrig. Pairing-friendly elliptic curves of prime order. In B. Preneel and S. Tavares, editors, *Selected Areas in Cryptography – SAC 2005*, LNCS 3897, pages 319–331, Kingston, ON, Canada, 2006. Springer.

[BS08] G. Bisson and T. Satoh. More discriminants with the Brezing-Weng method. In D. R. Chowdhury, V. Rijmen, and A. Das, editors, *Progress in Cryptology - INDOCRYPT 2009*, LNCS 5365, pages 389–399, Kharagpur, India, 2008. Springer.

[BSI09] BSI. Elliptic curve cryptography. Technical Guideline TR-03111, Version 1.11, April 2009.

[BSI10] BSI. Advanced Security Mechanisms for Machine Readable Travel Documents – Extended Access Control (EAC), Password Authenticated Connection Establishment (PACE), and Restricted Identification (RI). Technical Guideline TR-03110, 2010.

[BSS00] I. Blake, G. Seroussi, and N. Smart. *Elliptic Curves in Cryptography*, volume 265 of *London Mathematical Society*. Cambridge University Press, 2000.

[BSS05] I. Blake, G. Seroussi, and N. Smart, editors. *Advances in Elliptic Curve Cryptography*. Cambridge University Press, Cambridge, 2005.

[BSSC05] I. Blake, G. Seroussi, N. Smart, and J. W. S. Cassels, editors. *Advances in Elliptic Curve Cryptography (London Mathematical Society Lecture Note Series)*, volume 317 of *London Mathematical Society*. Cambridge University Press, 2005.

[Buc04] J. Buchmann. *Introduction to Cryptography*. Springer, 2. edition, 2004.

[BV96] D. Boneh and R. Venkatesan. Hardness of computing the most significant bits of secret keys in Diffie-Hellman and related schemes. In *CRYPTO*, volume 1109 of *Lecture Notes in Computer Science*, pages 129–142. Springer, 1996.

[BW05] F. Brezing and A. Weng. Elliptic curves suitable for pairing based cryptography. *Designs, Codes and Cryptography*, 37:133–141, 2005.

[CFA+05] H. Cohen, G. Frey, R. Avanzi, C. Doche, T. Lange, K. Nguyen, and F. Vercauteren, editors. *Handbook of Elliptic and Hyperelliptic Curve Cryptography*. Number 34 in Discrete Mathematics and Its Applications. Chapman& Hall/CRC, 2005.

[CHM10] S. Chatterjee, D. Hankerson, and A. Menezes. On the efficiency and security of pairing-based protocols in the type 1 and type 4 settings. In M. Hasan and T. Helleseth, editors, *Arithmetic of Finite Fields*, LNCS 6087, pages 114–134, Istanbul, Turkey, 2010. Springer.

[CLN11] C. Costello, K. Lauter, and M. Naehrig. Attractive subfamilies of BLS curves for implementing high-security pairings. In D. Bernstein and S. Chatterjee, editors, *Progress in Cryptology - INDOCRYPT 2011*, LNCS 7107, pages 320–342, Chennai, India, 2011. Springer.

[Coh93] H. Cohen. *A Course in Computational Algebraic Number Theory*. Springer, 1993.

[CP05] R. Crandall and C. Pomerance. *Prime Numbers: A Computational Perspective*. Springer, second edition, 2005.

[DCC06] P. Duan, S. Cui, and C. Chan. Special polynomial families for generating more suitable elliptic curves for pairing-based cryptosystems. EHAC'06 Proceedings of the 5th WSEAS International Conference on Electronics, Hardware, Wireless and Optical Communications, 2006.

[DH76] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Trans. Inf. Theory*, 22:644–654, 1976.

[Die11] C. Diem. On the discrete logarithm problem in elliptic curves. *Compositio Mathematica*, 147(01):75–104, 2011.

[ECC05] ECC Brainpool. ECC brainpool standard curves and curve generation. Internet Draft, `http://www.ecc-brainpool.org/download/Domain-parameters.pdf`, October 2005.

[ElG85] T. ElGamal. A public-key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Trans. Inf. Theory*, IT-31:469–472, 1985.

[FP11] W. Fumy and M. Paeschke, editors. *Handbook of eID Security.* Publicis Publishing, Erlangen, 2011.

[FR94] G. Frey and H.-G. Rück. A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves. *Math. Comp.*, 62:865–874, 1994.

[Fre10] G. Frey. The arithmetic behind cryptography. *Notices Am. Math. Soc.*, 57(3):366–374, 2010.

[FST10] D. Freeman, M. Scott, and E. Teske. A taxonomy of pairing-friendly elliptic curves. *J. Cryptology*, 23(2):224–280, 2010.

[Gal05] S. Galbraith. Pairings (book chapter). In Blake et al. [BSS05].

[Gau09] P. Gaudry. Index calculus for abelian varieties of small dimension and the elliptic curve discrete logarithm problem. *J. Symb. Comput.*, 44:1690–1702, December 2009.

[GHS02a] S. Galbraith, K. Harrison, and S. Soldera. Implementing the Tate pairing. In C. Fieker and D. R. Kohel, editors, *Proceedings of the Fifth Symposium on Algorithmic Number Theory, ANTS-V*, LNCS 2369, pages 324–337, Sydney, Australia, 2002. Springer.

[GHS02b] 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.

[GM05] S. Galbraith and A. Menezes. Algebraic curves and cryptography. *Finite Fields and Applications*, 11(3):544–577, 2005.

[GMV07] S. D. Galbraith, J. F. McKee, and P. C. Valenca. Ordinary Abelian varieties having small embedding degree. *Finite Fields Appl.*, 13(4):800–814, 2007.

[GPS08] S. D. Galbraith, K. G. Paterson, and N. P. Smart. Pairings for cryptographers. *Discrete Appl. Math.*, 156(16):3113–3121, 2008.

[GR04] S. Galbraith and V. Rotger. Easy decision Diffie-Hellman groups. *LMS J. Comput. Math.*, 7:201–218, 2004.

[GS99] S. Galbraith and N. P. Smart. A cryptographic application of Weil descent. In M. Walker, editor, *Cryptography and Coding*, LNCS 1746, pages 191–200, Cirencester, 1999. Springer.

[Hes04] F. Hess. A note on the Tate pairing of curves over finite fields. *Arch. Math.*, 82:28–32, 2004.

[Hes05] F. Hess. Weil descent attacks. In Blake et al. [BSSC05], pages 151–182.

[Hes08] F. Hess. Pairing lattices. In S. Galbraith and K. Paterson, editors, *Progress in Cryptology - INDOCRYPT 2009*, LNCS 5208, pages 18–38, Egham, UK, 2008. Springer.

[HMV04] D. Hankerson, A. Menezes, and S. Vanstone. *Guide to elliptic curve cryptography.* Springer Professional Computing, 2004.

[HSV06] F. Hess, N. Smart, and F. Vercauteren. The Eta pairing revisited. *IEEE Transactions on Information Theory*, 52(10):4595–4602, 2006.

[JKS+00] M. J. Jacobson, Jr., N. Koblitz, J. H. Silverman, A. Stein, and E. Teske. Analysis of the xedni calculus attack. *Designs, Codes and Cryptography*, 20(1):41–64, 2000.

[JMS04] M. J. Jacobson, Jr., A. J. Menezes, and A. Stein. Hyperelliptic curves and cryptography. In *High Primes and Misdemeanours: lectures in honour of the 60th birthday of Hugh Cowie Williams*, volume 41 of *Fields Institute Communications Series*, pages 255–282. Amer. Math. Soc., 2004.

[Jou00] A. Joux. A one round protocol for tripartite Diffie–Hellman. In W. Bosma, editor, *Proceedings of the Fourth Symposium on Algorithmic Number Theory, ANTS-IV*, LNCS 1838, pages 385–394, Leiden, Netherlands, 2000. Springer.

[KLL+11] W. Killmann, T. Lange, M. Lochter, W. Thumser, and G. Wicke. Minimum requirements for evaluating side-channel attack resistance of elliptic curve implementations. Downloadable via `http://www.bsi.bund.de`, 2011.

[Kob87] N. Koblitz. Elliptic curve cryptosystems. *Math. Comput.*, 48:203–209, 1987.

[LM10] M. Lochter and J. Merkle. Elliptic curve cryptography (ecc) brainpool standard curves and curve generation. IETF Internet Draft, RFC 5639, March 2010.

[Mil86] V. Miller. Use of elliptic curves in cryptography. In *Advances in Cryptology - CRYPTO'85*, volume 218 of *Lecture Notes in Computer Science*, pages 417–426. Springer, 1986.

[Mil04] V. Miller. The Weil pairing, and its efficient calculation. *J. Cryptology*, 17:235–261, 2004.

[MNT01] A. Miyaji, M. Nakabayashi, and S. Takano. New explicit conditions of elliptic curve traces for FR-reduction. *IEICE Trans. Fundamentals*, E84:1234–1243, 2001.

[MOV93] A. J. Menezes, T. Okamoto, and S. A. Vanstone. Reducing elliptic curve logarithms to a finite field. *IEEE Trans. on Inform. Theory*, 39:1639–1646, 1993.

[NIS09] NIST. Digital signature standard. FIPS Publication 186-3, 2009.

[NIS11] NIST. Recommendation for key derivation through extraction-then-expansion. NIST Special Publication 800-56C, November 2011.

[NS03] P. Q. Nguyen and I. E. Shparlinski. The insecurity of the elliptic curve digital signature algorithm with partially known nonces. *Des. Codes Cryptography*, 30(2):201–217, 2003.

[Pat05a] K. Paterson. Cryptography from pairings (book chapter). In Blake et al. [BSS05].

[Pat05b] K. Paterson. Identity-based cryptography - panacea or pandemonium? Invited talk at 9th Workshop on Elliptic Curve Cryptography (ECC 2005). Available under `http://www.cacr.math.uwaterloo.ca/conferences/2005/ecc2005/paterson.pdf`, 2005.

[RSA78] R.L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21:120–126, 1978.

[Sha85] A. Shamir. Identity based cryptosystems and signature schemes. In G. R. Blakley and D. Chaum, editors, *Advances in Cryptology - CRYPTO 1984*, LNCS 196, pages 47–53, Santa Barbara, 1985. Springer.

[Sil94] J. H. Silverman. *Advanced Topics in the Arithmetic of Elliptic Curves*, volume 151 of *Graduate Texts in Mathematics*. Springer, 1994.

[Sil00] J. H. Silverman. The xedni calculus and the elliptic curve discrete logarithm problem. *Des. Codes Cryptography*, 20:5–40, April 2000.

[Sil09] J. H. Silverman. *The Arithmetic of Elliptic Curves*, volume 106 of *Graduate Texts in Mathematics*. Springer, second edition, 2009.

[SOK00] R. Sakai, K. Ohgishi, and M. Kasahara. Cryptosystems based on pairing. In *Symposium on Cryptography and Information Security (SCIS2000)*, Okinawa, 2000.

[Sti05] D. R. Stinson. *Cryptography: Theory and Practice*. Chapman & Hall/CRC, third edition, 2005.

[Sti08] H. Stichtenoth. *Algebraic Function Fields and Codes*. Springer, 2. edition, 2008.

[ULS12] J. Urroz, F. Luca, and I. Shparlinski. On the number of isogeny classes of pairing-friendly elliptic curves and statistics of mnt curves. *Math. Comput.*, 81:1093–1110, 2012.

[Ver04] E. Verheul. Evidence that XTR is more secure than supersingular elliptic curve cryptosystems. *J. Cryptology*, 17:277–296, 2004.

[Ver10] F. Vercauteren. Optimal pairings. *IEEE Trans. Inf. Theory*, 56(1):455–461, 2010.

[VJS11] M. D. Velichka, M. J. Jacobson Jr., and A. Stein. Computing discrete logarithms in the jacobian of high-genus hyperelliptic curves over even characteristic finite fields. *IACR Cryptology ePrint Archive*, 2011:98, 2011.

[Was08] L. C. Washington. *Elliptic curves. Number theory and cryptography. 2nd ed.* Boca Raton, FL: Chapman and Hall/CRC. xviii, 513 p., 2008.