New Number-Theoretic Cryptographic Primitives

Éric Brier[1], Houda Ferradi[2], Marc Joye[3], and David Naccache[4]

[1] Ingenico
9 Avenue de la Gare, 26300 Alixan, France
eric.brier@ingenico.com
[2] NTT Secure Platform Laboratories
3–9–11 Midori-cho, Musashino-shi, Tokyo 180–8585, Japan
houda.ferradi@ens.fr
[3] OneSpan
Romeinsesteenweg 564 C, 1853 Grimbergen, Belgium
marc.joye@onespan.com
[4] DIENS, École normale supérieure, CNRS, PSL University
45 rue d'Ulm, 75230, Paris CEDEX 05, France
david.naccache@ens.fr

Abstract. This paper introduces new $p^r q$-based one-way functions and companion signature schemes. The new signature schemes are interesting because they *do not* belong to the two common design blueprints, which are the inversion of a trapdoor permutation and the Fiat–Shamir transform.
In the basic signature scheme, the signer generates multiple RSA-like moduli $n_i = {p_i}^2 q_i$ and keeps their factors secret. The signature is a bounded-size prime whose Jacobi symbols with respect to the n_i's match the message digest. The generalized signature schemes replace the Jacobi symbol with higher-power residue symbols. The case of 8^{th}-power residue symbols is fully detailed along with an efficient implementation thereof.
Given of their very unique design the proposed signature schemes seem to be overlooked "missing species" in the corpus of known signature algorithms.

Keywords: r^{th}-power residue symbol; r^{th}-order imprint; $p^r q$ moduli; number theory; one-way functions; digital signatures; cryptographic primitives.

1 Introduction

One-Way Functions. A fundamental building block for constructing secure signature schemes or public-key cryptosystems is *one-way functions* [15, Chapter 2]. Informally, a *one-way function* (OWF) is a function f that is easy to compute in polynomial time (by definition) on every input, but hard to invert given the image of a random input.

Basically, there exist three families of OWFs: (i) one-way permutations which are bijective OWFs, (ii) trapdoor OWFs which are one-way unless some extra information is given, and (iii) collision-free or collision-resistant hash functions. Almost all known OWFs have been based on intractable problems from number theory or some related mathematical fields like coding theory.

Digital Signatures. Diffie and Hellman in their seminal work [11] first pointed out the notion of digital signatures. Since then, there have been many signature proposals built from *trapdoor one-way permutations* based on different algebraic assumptions. The most well-known being the one devised by Rivest, Shamir and Adleman from the so-called RSA assumption [36].

Concurrently to the above, another popular approach to construct signature schemes is by using the Fiat–Shamir tranform [13]. It consists in turning a public-coin proof of knowledge into a signature scheme, which has yielded many efficient signature schemes like the Schnorr signature [42].

Cryptography Modulo $p^r q$. Moduli of the form $p^r q$ have found a few applications in cryptography since the mid 1980s, the most notable of which are probably the ESIGN signature scheme and its variants using $p^2 q$ [33,14,31,19,43], Okamoto–Uchiyama's cryptosystem [32,41], Schmidt-Samoa's cryptosystem [40] or constructions such as [44] and [38].

There are four main approaches of factorization algorithms for the structure $p^r q$: The *Elliptic Curve Method* (ECM) [27] which was improved by Peralta and Okamoto [35], the *Number Field Sieve* (NFS) [24], the *Lattice Factoring Method* (LFM) [4] and factoring using Jacobi symbols. Note that the special structure of $p^r q$ is not threatened by NFS beyond regular RSA moduli are threatened by that same attack. Actually, it turns out that using $p^2 q$ moduli does not seem to render factoring significantly easier. Boneh, Durfee and Howgrave-Graham [4] showed that $n = p^r q$ can be factored in polynomial time when r is large (i.e., $r \simeq \log p$). Consequently, as stated in [30], this LLL-based approach [25] does not apply to the setting considered in this paper where r is rather small. See also [29,28].

Organization. The rest of this paper is organized as follows. In the next section, we introduce some useful notation and review the definitions of the Jacobi symbol and of a signature scheme. Section 3 proposes a new OWF, building on the concept of Jacobi imprint. We then present in Section 4 a first signature scheme relying on this new OWF and prove its security. In Section 5, we generalize our basic design to higher-order residue symbols and introduce the corresponding signature schemes. As an illustration, we implement Octapus in Section 6, a signature scheme based on the octic residue symbol. Finally, we conclude the paper in Section 7.

2 Notation and Basic Definitions

If $\mathcal{D}$ is a finite domain, we let $x \xleftarrow{\$} \mathcal{D}$ denote picking an element of $\mathcal{D}$ uniformly at random and assigning it to x. A boldface variable $\boldsymbol{x}$ is used to denote a vector of elements identified by that variable; i.e., $\boldsymbol{x} = (x_0, \ldots, x_{k-1})$. The symbol $\mathbb{P}$ stands for the set of (rational) primes. Given a vector $\boldsymbol{n} = (n_0, \ldots, n_{k-1})$ of pairwise co-prime integers n_j $(0 \leq j \leq k-1)$ and a vector $\boldsymbol{x} = (x_0, \ldots, x_{k-1})$ of integers, we use $\mathsf{CRT}(\boldsymbol{x}, \boldsymbol{n})$ for the Chinese-remainder function, returning the smallest non-negative integer y such that $y \equiv x_j \pmod{n_j}$ for $0 \leq j \leq k-1$ [12, Chapter 2].

2.1 The Jacobi Symbol

Given a positive integer n, an integer a with $\gcd(a, n) = 1$ is called a *quadratic residue modulo* n if and only if $x^2 \equiv a \pmod{n}$ is solvable. If a is not a quadratic residue then it is called a *quadratic non-residue modulo* n.

Let a be an integer and let $p \in \mathbb{P}$, $p \neq 2$. The *Legendre symbol* $\left(\frac{a}{p}\right)$ is defined as:

$$\left(\frac{a}{p}\right) = \begin{cases} 1 & \text{if } a \text{ is a quadratic residue modulo } p\,, \\ -1 & \text{if } a \text{ is a quadratic non-residue modulo } p\,, \\ 0 & \text{if } \gcd(a,p) \neq 1\,. \end{cases}$$

The Legendre symbol satisfies Euler's criterion, namely $\left(\frac{a}{p}\right) \equiv a^{\frac{p-1}{2}} \pmod{p}$.

The *Jacobi symbol* is a natural generalization of the Legendre symbol.

Definition 1. *Let n be an odd positive integer with prime factorization $n = \prod_j p_j^{e_j}$. Then, for an integer a, the* Jacobi symbol $\left(\frac{a}{n}\right)$ *is given by*

$$\left(\frac{a}{n}\right) = \prod_j \left(\frac{a}{p_j}\right)^{e_j}$$

with the convention $\left(\frac{a}{1}\right) = 1$ for all integers a.

Interestingly, the prime factorization of n is not required for evaluating $\left(\frac{a}{n}\right)$. It can be efficiently computed with $O((\log_2 a)(\log_2 n))$ bit operations [1, § 5.9]. We point out that the Legendre and Jacobi symbols are indistinguishable when n is an odd prime. Also, we note that the Legendre symbol allows to determine whether an integer is a quadratic residue or not, whereas the Jacobi symbol does not allow checking this property.

2.2 Digital Signatures

A *signature scheme* [21] is a tuple, $\Sigma = (\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify})$, of probabilistic polynomial-time algorithms satisfying:

$\mathsf{KeyGen}(1^\kappa)$ On input security parameter 1^κ, key generation algorithm KeyGen produces a pair $(\mathsf{pk}, \mathsf{sk})$ of matching public and private keys.

$\mathsf{Sign}(\mathsf{sk}, m)$ Given a private key sk and a message m in a set $\mathcal{M}$ of messages, signing algorithm Sign produces a signature σ.

$\mathsf{Verify}(\mathsf{pk}, m, \sigma)$ Given a public key pk, a message $m \in \mathcal{M}$, and a signature σ, the verifying algorithm Verify checks whether σ is a valid signature on m with respect to pk.

The classical security notion for signature schemes is *existential unforgeability against chosen-message attacks* (in short, EUF-CMA) [17]. Basically, it requires that an adversary having access to a signing oracle returning the signature on messages of its choice is unable to produce a valid signature on a message not previously submitted to the signing oracle. In the *random oracle model* [2], the adversary has in addition access to a hash oracle viewed as a random oracle. More formally:

Definition 2. *A signature scheme Σ is* EUF-CMA *secure if, for every probabilistic polynomial-time adversary $\mathcal{A}$, the success probability,* $\mathsf{Adv}^{\mathsf{EUF}}_{\mathcal{A},\Sigma}(\kappa) := \Pr\left[\mathsf{EUF}^{\mathcal{A}}_{\Sigma}(\kappa) = 1\right]$*, is negligible against the security game defined in Figure 1.*

$\underline{\mathsf{EUF}^{\mathcal{A}}_{\Sigma}(\kappa)\text{:}}$
$\mathrm{Hist} \leftarrow \emptyset$
$(\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \Sigma.\mathsf{KeyGen}(1^\kappa)$
$(m^*, \sigma^*) \leftarrow \mathcal{A}^{\mathsf{Sign}(\mathsf{sk},\cdot)}(\mathsf{pk})$
if $m^* \notin \mathrm{Hist}$
  return $\Sigma.\mathsf{Verify}(\mathsf{pk}, m^*, \sigma^*)$
return 0

$\underline{\mathsf{Sign}(\mathsf{sk}, m)\text{:}}$
$\sigma \xleftarrow{\$} \Sigma.\mathsf{Sign}(\mathsf{sk}, m)$
$\mathrm{Hist} \leftarrow \mathrm{Hist} \cup \{m\}$
return σ

$\underline{\mathsf{Verify}(\mathsf{pk}, m, \sigma)\text{:}}$
return $\Sigma.\mathsf{Verify}(\mathsf{pk}, m, \sigma)$

Fig. 1. EUF-CMA experiment for digital signature schemes.

3 A Candidate One-Way Function

If p is an odd prime then half of the integers in the sequence $1, 2, \ldots, p-1$ are quadratic residues modulo p, and half are not. The problem of counting the number of occurrences of k distinct integers $(a_0, a_1, \ldots, a_{k-1})$ modulo p obeying a given pattern $(\epsilon_0, \epsilon_1, \ldots, \epsilon_{k-1})$ with $\epsilon_j = \left(\frac{a_j}{p}\right) \in \{-1, 1\}$ and variations thereof has been studied in a number of papers, including [9,10,7,34,18,37]. In particular, the results of Peralta in [34] indicate that the probability of

$$\left(\left(\tfrac{a_0}{p}\right), \left(\tfrac{a_1}{p}\right), \ldots, \left(\tfrac{a_{k-1}}{p}\right)\right)$$

matching any particular sequence $(\epsilon_0, \epsilon_1, \ldots, \epsilon_{k-1}) \in \{-1, 1\}^k$ is in the range $\frac{1}{2^k} \pm O(kp^{-1/2})$.

This section considers a related problem. It relies on a new notion that we call *Jacobi imprint*. In essence, the imprint is an integer formed of bits representing the sequence of Jacobi symbols where -1's are replaced by 1's and 1's by 0's.

Definition 3 (Jacobi Imprint). *For an integer a and $\boldsymbol{n} = (n_0, \ldots, n_{k-1}) \in \mathbb{N}^k$ such that* $\gcd(a, n_j) = 1$ *for* $0 \leq j \leq k-1$*, the* Jacobi imprint $\mathfrak{I}_{\boldsymbol{n}}(a)$ *is given by*

$$\mathfrak{I}_{\boldsymbol{n}}(a) = \sum_{j=0}^{k-1} \left\{ \frac{a}{n_j} \right\} 2^j \quad \textit{where} \quad \left\{ \frac{a}{n_j} \right\} = \frac{1 - \left(\frac{a}{n_j}\right)}{2} \ .$$

(At times we will interchangeably use $\mathfrak{I}_{\boldsymbol{n}}(a)$ to denote the integer $\mathfrak{I}_{\boldsymbol{n}}(a)$ or its binary representation.)

FUNCTION $\mathcal{F}_0$. Let $\boldsymbol{q} = (q_0, \ldots, q_{k-1})$ be a set of k distinct (odd) primes and let $Q = \prod_{j=0}^{k-1} q_j$. Consider the function $\mathcal{F}_0$ given by

$$\mathcal{F}_0 \colon \mathfrak{D} \subset \mathbb{Z}_Q^* \to \mathbb{N}, x \mapsto \mathcal{F}_0(x) = \mathfrak{I}_{\boldsymbol{q}}(x) \ .$$

We argue that an appropriate selection for the domain of $\mathcal{F}_0$ and the number of primes q_j's turns $\mathcal{F}_0$ into a one-way function.

Of course, $\mathfrak{D}$ cannot be the whole group $\mathbb{Z}_Q^*$. Otherwise, given a challenge $\hat{y} = \mathcal{F}_0(\hat{x})$, an attacker could execute Algorithm 1.

Algorithm 1: Finding a (large) pre-image

Data: $\hat{y} = \sum_{j=0}^{k-1} \hat{y}_j \, 2^j$ with $\hat{y}_j \in \{0, 1\}$ and $\boldsymbol{q} = (q_0, \ldots, q_{k-1})$
Result: $x \in \mathbb{Z}_Q^*$ such that $\mathcal{F}_0(x) = \hat{y}$
for $0 \leq j \leq k-1$ **do**
  $r_j \xleftarrow{\$} \mathbb{Z}_{q_j}^*$ such that $\{\frac{r_j}{q_j}\} = \hat{y}_j$
end
$x \leftarrow \mathsf{CRT}(\boldsymbol{r}, \boldsymbol{q})$ where $\boldsymbol{r} = (r_0, \ldots, r_{k-1})$
return x

This algorithm yields outputs that are smaller than $Q = \prod_{j=0}^{k-1} q_j$. An obvious way to prevent an attacker to successfully run Algorithm 1 would be to restrict $\mathfrak{D}$ to entries smaller than a given bound B.

But there is another way to tackle the problem of finding pre-images to $\mathcal{F}_0$. Let $\mathcal{Z}$ be the set of k-bit integers in $\mathbb{N}$. Now if we regard an imprint in $\mathcal{Z}$ as an element of $(\mathbb{Z}_2)^k$ (that is, if we look at its binary representation), we see that $\mathcal{F}_0$ induces a group homomorphism from $(\mathbb{Z}_Q^*, \cdot)$ to $(\mathcal{Z}, \oplus)$:

$$\mathcal{F}_0(x_1 \cdot x_2 \bmod Q) = \mathcal{F}_0(x_1) \oplus \mathcal{F}_0(x_2) \,, \quad \forall x_1, x_2 \in \mathbb{Z}_Q^* \ .$$

Therefore, an attacker could generate a set of ℓ "small" primes p_i's (with $p_i \nmid Q$) and compute the corresponding imprint $z_i = \mathcal{F}_0(p_i)$, for $1 \leq i \leq \ell$. It suffices then for the attacker to use linear algebra modulo 2 (i.e., Gaussian elimination) to find a subset of the z_i's having the target imprint $\hat{y}$ as an xor:[5]

$$\hat{y} = \varepsilon_1 z_1 \oplus \cdots \oplus \varepsilon_\ell z_\ell \quad \text{with } \varepsilon_i \in \{0, 1\} \ .$$

A pre-image is given by

$$x = \prod_{\substack{1 \leq i \leq \ell \\ \varepsilon_i = 1}} p_i \ ,$$

which is valid provided that $x < B$. This second attack is avoided by limiting $\mathfrak{D}$ to primes.

Furthermore, each prime q_j in $\boldsymbol{q}$ imposes a condition on the pre-image. The birthday paradox suggests to choose the number k of primes q_j's to be at least 2κ, where κ is the security parameter.

All in all, we recommend to select $k = 2\kappa$ and $\mathfrak{D} = \left\{ x \in \mathbb{P} \mid x < B \text{ with } B \ll Q \text{ where } Q = \prod_{j=0}^{k-1} q_j \right\}$.

[5] If a solution $\varepsilon_1, \ldots, \varepsilon_\ell$ does not exist, refresh the p_j's as necessary.

FROM $\mathcal{F}_0$ TO $\mathcal{F}_1$. We use function $\mathcal{F}_0$ as a starting point to define a (conjectured) *trapdoor one-way function*. The resulting function $\mathcal{F}_1$ has the extra property that it can be inverted when it is given a trapdoor as an additional input. To insert a trapdoor, we replace the primes q_j's with RSA-like moduli of the form $n_j = {p_j}^2 q_j$. This does not affect the output value since $\mathfrak{I}_{\boldsymbol{n}}(x) = \mathfrak{I}_{\boldsymbol{q}}(x)$ for all x such that $\gcd(x, n_j) = 1$ for $0 \leq j \leq k-1$. The trapdoor is $\boldsymbol{q}$.

We conjecture:

Assumption 1. *Let κ denote a security parameter. Let also $k = k(\kappa)$ and $\ell = \ell(\kappa)$. Define $\mathfrak{D} = \left\{x \in \mathbb{P} \mid x < 2^{k\ell}\right\}$ and*

$$\mathcal{F}_1 \colon \mathfrak{D} \to \mathbb{N}, x \mapsto \mathcal{F}_1(x) = \mathfrak{I}_{\boldsymbol{n}}(x)$$

where $\boldsymbol{n} = (n_0, \ldots, n_{k-1})$ is a set of k pairwise co-prime moduli of the form $n_j = {p_j}^2 q_j$ for ℓ-bit primes p_j and q_j, $0 \leq j \leq k-1$. For every polynomial-time algorithm $\mathcal{A}$, the success probability

$$\Pr\left[\hat{x} \xleftarrow{\$} \mathfrak{D}; \mathcal{A}(\mathcal{F}_1(\hat{x})) = x \mid \mathcal{F}_1(x) = \mathcal{F}_1(\hat{x})\right]$$

is negligible.

Note that finding a pre-image to $\hat{y} = \mathcal{F}_1(\hat{x})$ is easy given the trapdoor $\boldsymbol{q} = (q_0, \ldots, q_{k-1})$:

1. Run Algorithm 1 and obtain x such that $\mathfrak{I}_{\boldsymbol{q}}(x) = \hat{y}$;
2. Update x as $x \leftarrow xu^2 \bmod Q$ with $u \xleftarrow{\$} \mathbb{Z}_Q^*$ until x is prime;
3. Return x.

Clearly, the so-obtained x is a valid pre-image: $x \in \mathfrak{D}$ and $\mathcal{F}_1(x) = \hat{y}$.

Remark 1. By definition, the Jacobi imprint $\mathfrak{I}_{\boldsymbol{n}}(x)$ requires x to be co-prime with n_j for $0 \leq j \leq k-1$. Strictly speaking, the domain $\mathfrak{D}$ should therefore exclude the primes p_j and q_j. However, since primes p_j and q_j are ℓ-bit primes—where $\ell = \ell(\kappa)$—the probability to output an x such that $\gcd(x, n_j) \neq 1$ for some $0 \leq j \leq k-1$ is negligible when the prime factorization of the n_j's is unknown.

4 Signatures Modulo p^2q

We are now ready to formally describe a first signature scheme. We prove that it meets the EUF-CMA security level in the random oracle model.

4.1 Description

Our basic signature scheme is a tuple of algorithms $\Sigma = (\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify})$, which we define as follows:

Key generation The key generation algorithm KeyGen takes as input a security parameter 1^κ and defines parameters k and ℓ. It selects a collision-resistant hash function $H \colon \{0,1\}^* \to \{0,1\}^k$. It also produces k pairs (p_j, q_j) of ℓ-bit primes and forms the moduli $n_j = {p_j}^2 q_j$. The public parameters are $\mathsf{pp} = (k, \ell, H)$. The public key is $\mathsf{pk} = \{n_j\}_{0 \leq j \leq k-1}$ while the private key is $\mathsf{sk} = \{q_j\}_{0 \leq j \leq k-1}$. The outputs are pk and sk (and pp).

Signing The signing algorithm Sign takes as inputs a message $m \in \{0,1\}^*$ and the secret key sk. The signature on message m proceeds as follows:

1. Compute $H(m) = \sum_{j=0}^{k-1} h_j \, 2^j$ with $h_j \in \{0,1\}$;
2. Pick up at random k ℓ-bit integers r_j such that

$$\left\{\frac{r_j}{q_j}\right\} = h_j \,, \quad \text{for } 0 \leq j \leq k-1 \,;$$

3. Compute
$$R = \mathsf{CRT}(\boldsymbol{r}, \boldsymbol{q})$$
with $\boldsymbol{r} = (r_0, \ldots, r_{k-1})$ and $\boldsymbol{q} = (q_0, \ldots, q_{k-1})$;
4. Set $Q = \prod_{j=0}^{k-1} q_j$ and choose at random an integer $u \in \mathbb{Z}_Q^*$ such that
$$\sigma \coloneqq R\,u^2 \bmod Q \quad \in \mathbb{P}\,;$$
5. Return σ.

Verification The verifying algorithm Verify takes as inputs the public pk, a message m, and a signature σ on message m. It checks whether
$$\text{(i) } \sigma \in \mathbb{P}\,, \quad \text{(ii) } \sigma < 2^{\ell k}\,, \quad \text{(iii) } \mathfrak{I}_{\boldsymbol{n}}(\sigma) = H(m)$$
where $\boldsymbol{n} = (n_0, \ldots, n_{k-1})$. Verify returns 1 (i.e., the signature is accepted) if and only if the three conditions above are fulfilled. Otherwise, Verify returns 0.

The next proposition shows that the signature scheme is correct: for $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^\kappa)$ and any message $m \in \{0,1\}^*$, $\mathsf{Verify}\big(\mathsf{pk}, m, \mathsf{Sign}(m, \mathsf{sk})\big) = 1$.

Proposition 1 (Correctness). *Signature scheme Σ is correct.*

Proof. Let $(\{n_j\}, \{q_j\})$ and σ the respective outputs of KeyGen and Sign, with message m as input. By construction, σ is prime and $\sigma = R\,u^2 \bmod Q < 2^{\ell k}$. Moreover, since $\sigma \equiv r_j u^2 \pmod{q}_j$ $(0 \le j \le k-1)$, it follows that
$$\mathfrak{I}_{\boldsymbol{q}}(\sigma) = \sum_{j=0}^{k-1} \left\{ \frac{r_j u^2}{q_j} \right\} 2^j = \sum_{j=0}^{k-1} \left\{ \frac{r_j}{q_j} \right\} 2^j.$$
Finally, since $n_j = {p_j}^2 q_j$, we have
$$\left\{ \frac{r_j}{n_j} \right\} = \left\{ \frac{r_j}{q_j} \right\},$$
and so $\mathfrak{I}_{\boldsymbol{n}}(\sigma) = \mathfrak{I}_{\boldsymbol{q}}(\sigma) = H(m)$. □

4.2 Security Proof

Theorem 1. *Signature scheme Σ is* EUF-CMA *secure assuming the hardness of inverting $\mathcal{F}_1$, in the random oracle model.*

Proof. The security proof is by contradiction. Suppose we are given as a challenge an output $\hat{s}$ of the function $\mathcal{F}_1$. We assume that there exists a polynomial-time adversary $\mathcal{A}$ that is able to produce an existential signature forgery with non-negligible success probability. Adversary $\mathcal{A}$ is allowed to make q_H queries to random oracle H and q_s queries to signing oracle Sign. We then use $\mathcal{A}$'s forgery to invert $\mathcal{F}_1$; i.e., to find a pre-image to $\hat{s}$.

Specifically, suppose that the received challenge is the k-bit integer
$$\hat{s} \leftarrow \mathcal{F}_1(x) = \mathfrak{I}_{\boldsymbol{n}}(x) \quad \text{with } \boldsymbol{n} = (n_0, \ldots, n_{k-1})$$
for moduli n_j of the form $n_j = {p_j}^2 q_j$ where p_j's and q_j's are ℓ-bit primes; $0 \le j \le k-1$. The simulator sets the public key to $\mathsf{pk} = \{n_j\}_{0 \le j \le k-1}$. It also selects a collision-resistant hash function H mapping to $\{0,1\}^k$. The public key pk as well as public parameters $\mathsf{pp} \coloneqq (k, \ell, H)$ are given to $\mathcal{A}$.

The simulator needs to answer the oracle queries made by $\mathcal{A}$. It maintains a history list of tuples $(m_i, \mathfrak{h}_i, \sigma_i)$, Hist$[H]$, that keeps track of the hash queries; Hist$[H]$ is initialized to $\emptyset$. It also maintains a counter i initialized to 0 and chooses at random an index $i^* \in [1, \ldots, q_H]$.

Answering hash queries When $\mathcal{A}$ submits a message m to H, the simulator checks whether m was already queried:
- If $m \notin \text{Hist}[H]$ then i is incremented: $i \leftarrow i+1$. Next, the simulator sets $m_i \leftarrow m$ and depending on the value of i:
  - if $i = i^*$, it sets $\mathfrak{h}_i \leftarrow \hat{s}$ and $\sigma_i \leftarrow \perp$;
  - if $i \neq i^*$, it generates a random ℓk-bit prime σ_i and sets $\mathfrak{h}_i \leftarrow \mathfrak{I}_{\boldsymbol{n}}(\sigma_i)$.

  Tuple $(m_i, \mathfrak{h}_i, \sigma_i)$ is appended to Hist$[H]$: Hist$[H] \leftarrow$ Hist$[H] \cup (m_i, \mathfrak{h}_i, \sigma_i)$.
- If $m \in \text{Hist}[H]$, the simulator finds the index i such that $m = m_i$ and recovers the corresponding value $\mathfrak{h}_i$.

The simulator returns $\mathfrak{h}_i$ as the hash value of input message m.

Answering signature queries Without loss of generality, we assume that when $\mathcal{A}$ calls signing oracle Sign with a message m, it has already submitted m to hash oracle H (observe that the simulator can always call internally H). Therefore, there exists an index i such that $m = m_i$ in Hist$[H]$. The simulator recovers the corresponding value for σ_i. There are two cases:
- If $\sigma_i \neq \perp$ then the simulator returns σ_i as a valid signature on input message m;
- Otherwise the simulator fails and stops.

The number of queries to the hash oracle being polynomial, with non-negligible probability, the adversary will return a signature forgery on its i^*-th query to H; i.e., on message m_{i^*}. Letting σ_{i^*} the corresponding signature returned by $\mathcal{A}$, we see that σ_{i^*} is a solution to the challenge since $\mathfrak{I}_{\boldsymbol{n}}(\sigma_{i^*}) = H(m_{i^*}) = \hat{s}$. □

4.3 Toy Example ($k = 8$)

Picking the secret primes

	$j=0$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$	$j=7$
p_j	59069	54139	52639	53813	49871	41269	53653	40361
q_j	62989	32917	36583	48383	36653	34963	52517	38971

we have the public moduli

$$n_0 = 219777865328629 \qquad n_1 = 096480757993357 \qquad n_2 = 101366529455143$$
$$n_3 = 140109376837127 \qquad n_4 = 091160286242573 \qquad n_5 = 059546546811643$$
$$n_6 = 151177768427453 \qquad n_7 = 063484161219691$$

and the value $Q = \prod_{i=0}^{7} q_i = 962535482083430844430189085476678516 1$.

Consider a message whose digest is $\boldsymbol{h} = (h_0, \ldots, h_7)$ and draw r_j's as:

	$j=0$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$	$j=7$
h_j	1	0	1	1	0	1	1	0
r_j	64863	58999	47120	50684	37458	57079	43135	56942

We get $\mathsf{CRT}(\boldsymbol{r}, \boldsymbol{q}) = 139578625155923187878976453585864 1198$.
By selecting $u = 215226682070986629514007750468780345 9$, we obtain the signature

$$\sigma = 113754256158676123077058534525609284 1 \in \mathbb{P} \ .$$

5 Generalized Signatures

The Legendre symbol tells whether an integer is a square modulo a prime p. Given an integer a and an odd prime p, if $p \nmid a$, there exists a unique integer j modulo 2 such that $a^{(p-1)/2} \equiv (-1)^j \pmod{p}$. To obtain the analogue to a higher power r, the rational integers need to be extended so that they include an r^{th} root of unity, namely $e^{2\pi i/r}$.

5.1 Cyclotomic Integers and Higher-Order Residuosity

We start by reviewing some classical results on cyclotomic fields. We refer the reader to [20] and [45] for further introductory background.

Fix $\zeta := \zeta_r$ a primitive r^{th} root of unity; i.e., ζ is a root of $X^r - 1$ and $X^s \neq 1$ for $0 < s < r$. Adjoining ζ to the field $\mathbb{Q}$ of rationals defines the *cyclotomic field* $\mathbb{Q}(\zeta)$. It is the splitting field of $X^r - 1$; its Galois group $\mathrm{Gal}(\mathbb{Q}(\zeta)/\mathbb{Q})$ is isomorphic to $\mathbb{Z}_r^*$, with $k \bmod r$ corresponding to the map $\sigma_k \colon \zeta \mapsto \zeta^k$; see [20, Proposition 13.2.1] or [45, Theorem 2.5]. The ring of integers of $\mathbb{Q}(\zeta)$ is $\mathbb{Z}[\zeta] \cong \mathbb{Z}[X]/(\Phi_r)$ where Φ_r is the r^{th} *cyclotomic polynomial*; see [45, Theorem 2.6].

The elements α of $\mathbb{Z}[\zeta]$ are written as

$$\alpha = \sum_{0 \le j < \varphi(r)} a_j \, \zeta^j \quad \text{with } a_j \in \mathbb{Z}$$

where φ denotes Euler's totient function. The norm of $\alpha \in \mathbb{Z}[\zeta]$ is the rational integer given by $\mathrm{N}(\alpha) = \prod_{k \in \mathbb{Z}_r^*} \sigma_k(\alpha)$. We assume that $\mathbb{Z}[\zeta]$ is *norm-Euclidean*.[6]

The elements of norm ± 1 in $\mathbb{Z}[\zeta]$ are called *units*. Two elements $\alpha, \beta \in \mathbb{Z}[\zeta]$ that are equal up to multiplication by a unit $\upsilon \in \mathbb{Z}[\zeta]$ (i.e., $\alpha = \upsilon\beta$) are said to be *associates*; we write $\alpha \sim \beta$. A non-unit element $\pi \in \mathbb{Z}[\zeta]$ is a *prime in* $\mathbb{Z}[\zeta]$ if, for any $\alpha, \beta \in \mathbb{Z}[\zeta]$, $\pi \mid \alpha\beta$ implies $\pi \mid \alpha$ or $\pi \mid \beta$. If r is a prime power (i.e., $r = q^\ell$ for some rational prime q and $\ell \ge 1$) then $(1 - \zeta)$ is a prime in $\mathbb{Z}[\zeta]$ and $\mathrm{N}(1 - \zeta) = q$; otherwise, $(1 - \zeta)$ is a unit in $\mathbb{Z}[\zeta]$.

Let π be a prime in $\mathbb{Z}[\zeta]$, with $\gcd(\mathrm{N}(\pi), r) = 1$. For every $\alpha \in \mathbb{Z}[\zeta]$ such that $\pi \nmid \alpha$, we have $\alpha^{\mathrm{N}(\pi)-1} \equiv 1 \pmod{\pi}$. Further, $\langle \zeta \rangle$ is a subgroup of order r of $(\mathbb{Z}[\zeta]/(\pi))^*$, it follows that $r \mid (\mathrm{N}(\pi) - 1)$ and

$$\alpha^{\frac{\mathrm{N}(\pi)-1}{r}} \equiv \zeta^j \pmod{\pi} \quad \text{for some } j \in \mathbb{Z}_r \ .$$

This defines the r^{th}-power residue symbol.

Definition 4. *Fix ζ a primitive r^{th} root of unity. Let $\alpha, \pi \in \mathbb{Z}[\zeta]$ with π prime and $\gcd(\mathrm{N}(\pi), r) = 1$. The r^{th}-power residue symbol is defined by*

$$\left[\frac{\alpha}{\pi}\right]_r = \begin{cases} \alpha^{(\mathrm{N}(\pi)-1)/r} \bmod \pi & \text{if } \pi \nmid \alpha \,, \\ 0 & \text{otherwise} \,. \end{cases}$$

Let $\alpha, \beta, \pi \in \mathbb{Z}[\zeta]$ with π prime and $\gcd(\mathrm{N}(\pi), r) = 1$. It is easily verified from the definition that the following properties are satisfied:

$$\left[\frac{\alpha\beta}{\pi}\right]_r = \left[\frac{\alpha}{\pi}\right]_r \left[\frac{\beta}{\pi}\right]_r , \qquad \left[\frac{\alpha}{\pi}\right]_r = \left[\frac{\alpha \bmod \pi}{\pi}\right]_r \ .$$

Furthermore, in a way similar to the Jacobi symbol for quadratic residuosity, the r^{th}-power residue symbol naturally generalizes.

Definition 5. *Fix ζ a primitive r^{th} root of unity. Let $\alpha, \lambda \in \mathbb{Z}[\zeta]$ with λ non-unit and $\gcd(\mathrm{N}(\lambda), r) = 1$. Then, writing $\lambda = \prod_j \pi_j^{e_j}$ for primes π_j in $\mathbb{Z}[\zeta]$, if α and λ are co-prime, the symbol $\left[\frac{\alpha}{\lambda}\right]_r$ is defined by*

$$\left[\frac{\alpha}{\lambda}\right]_r = \prod_j \left[\frac{\alpha}{\pi_j}\right]_r^{e_j} \ .$$

Moreover, $\left[\frac{\alpha}{\upsilon}\right]_r = 1$ for every unit $\upsilon \in \mathbb{Z}[\zeta]$.

[6] A ring R is said *norm-Euclidean* or *Euclidean with respect to the norm* N if for every $\alpha, \beta \in R$, $\beta \neq 0$, there exist $\eta, \rho \in R$ such that $\alpha = \beta\,\eta + \rho$ and $\mathrm{N}(\rho) < \mathrm{N}(\beta)$.

The notion of Jacobi imprint generalizes to higher powers. To ease the notation, we extend the brace symbol as follows:

$$\left\{\frac{\alpha}{\lambda}\right\}_r = j \quad \text{with } j \in \mathbb{Z}_r$$

where $\left\{\frac{\alpha}{\lambda}\right\}_r = j$ if and only if $\left[\frac{\alpha}{\lambda}\right]_r = \zeta^j$. Note that Definition 3 corresponds to the case $r = 2$.

Definition 6 (r^{th}-order Imprint). *For an integer $\alpha \in \mathbb{Z}[\zeta]$ and a vector $\boldsymbol{\lambda} = (\lambda_0, \ldots, \lambda_{k-1}) \in \mathbb{Z}[\zeta]^k$, such that α and λ_j (with $0 \le j \le k-1$) are co-prime, the* r^{th}-order imprint of α w.r.t. $\boldsymbol{\lambda}$ *is the integer $\mathfrak{I}_{\boldsymbol{\lambda}}^{(r)}(\alpha) \in \mathbb{Z}$ given by*

$$\mathfrak{I}_{\boldsymbol{\lambda}}^{(r)}(\alpha) = \sum_{j=0}^{k-1} \left\{\frac{\alpha}{\lambda_j}\right\}_r r^j \quad .$$

5.2 Parameter Selection

As discussed in the introduction, the main threat for factoring-related cryptosystems comes from NFS and its variants. A factor of 512 bits is really beyond the scope of ECM. The current state of the art teaches that moduli could be selected of the form $p_j{}^r q_j$ with r chosen to have a balanced resistance against both types of factoring algorithms.

The next table lists different types of security level and the commonly-accepted corresponding size for the modulus and its (smallest) prime factor. See e.g. [3].

Table 1. Key lengths and bit security.

Type	Bit-security level	Prime factors (bit size)	Modulus (bit size)
Legacy	80	160	1024
Basic	112	224	2048
Normal	128	256	3072
High	192	384	7680
Very high	256	512	15360

Depending on the security level, this suggests to select r in the set $\{5, 8, 11, 19, 29\}$ or, more generally, as a parameter ranging from 5 (i.e., $\approx \frac{1024}{160} - 1$) up to 29 (i.e., $\frac{15360}{512} - 1$).

If ζ_r is an r^{th} primitive root of unity, the ring $\mathbb{Z}[\zeta_r]$ is not necessarily norm-Euclidean. It is known that the rings $\mathbb{Z}[\zeta_r]$ with

$$r = 1, 3, 4, 5, 7, 8, 9, 11, 12, 13, 15, 16, 20, 24$$

are norm-Euclidean [22, §8]; see also [26]. Moreover, since $\mathbb{Z}[\zeta_{2r}] = \mathbb{Z}[\zeta_r]$ for r odd, the values

$$r = 2, 6, 10, 14, 18, 22, 26, 30$$

also produce norm-Euclidean rings. Each possible value for r gives rise to a signature scheme. Of particular interest are the following new species in the signature zoo:

Pentapus[7]	$r = 5$	legacy security;
Octapus	$r = 8$	basic security;
Hendecapus	$r = 11$	normal security;
Octadecapus	$r = 18$	high security;
Icosapus	$r = 20$	high security+;
Triacontapus	$r = 30$	very high security.

[7] Pentapus is an endangered species.

6 Octapus

The p^2q signature scheme given in Section 4 extends to any value of $r > 2$ (provided that $\mathbb{Z}[\zeta_r]$ is norm-Euclidean). As an illustration, we detail the Octapus signature scheme, which is an adaptation to the case $r = 8$.

Throughout this section, we let $\zeta \coloneqq \zeta_8 = \frac{\sqrt{2}}{2}(1+i)$ denote a primitive 8^{th} root of unity. Let also $\epsilon = 1+\sqrt{2} = 1+\zeta+\zeta^{-1}$. The field $\mathbb{Q}(\zeta) = \mathbb{Q}(i,\sqrt{2})$ is biquadratic and its group of units is $\langle \zeta, \epsilon \rangle$. The Galois group of $\mathbb{Q}(\zeta)/\mathbb{Q}$ contains the four automorphisms $\sigma_k \colon \zeta \mapsto \zeta^k$ with $k \in \{1,3,5,7\}$. For an element $\alpha \in \mathbb{Z}[\zeta]$, we write $\alpha_k = \sigma_k(\alpha)$. The (absolute) norm of α is given by $\mathrm{N}(\alpha) = \alpha_1\alpha_3\alpha_5\alpha_7$.

6.1 Description

The Octapus signature scheme, $(\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify})$, is defined as follows.

Key generation KeyGen takes as input a security parameter 1^κ and defines parameters k and ℓ. It selects a collision-resistant hash function $H \colon \{0,1\}^* \to (\mathbb{Z}_8)^k$. It also produces k pairs (π_j, ψ_j) of primes in $\mathbb{Z}[\zeta]$, where $\mathrm{N}(\pi_j)$ and $\mathrm{N}(\psi_j)$ are ℓ-bit long, and forms the moduli $\nu_j = {\pi_j}^8\psi_j$. The outputs are $\mathsf{pp} = (k, \ell, H)$, $\mathsf{pk} = \{\nu_j\}_{0\le j\le k-1}$, and $\mathsf{sk} = \{\psi_j\}_{0\le j\le k-1}$.

Signing On input a message $m \in \{0,1\}^*$ and sk, Sign does the following:

1. Compute $H(m) = \sum_{j=0}^{k-1} h_j\, 8^j$ with $h_j \in \mathbb{Z}_8$;
2. Pick at random k integers $\rho_j \in \mathbb{Z}[\zeta]$ of ℓ-bit norm such that
$$\left\{\frac{\rho_j}{\psi_j}\right\} = h_j\,, \quad \text{for } 0 \le j \le k-1\,;$$
3. Compute
$$\varrho = \mathsf{CRT}(\boldsymbol{\rho}, \boldsymbol{\psi})$$
with $\boldsymbol{\rho} = (\rho_0, \ldots, \rho_{k-1})$ and $\boldsymbol{\psi} = (\psi_0, \ldots, \psi_{k-1})$;
4. Set $\Psi = \prod_{j=0}^{k-1} \psi_j$ and choose at random an integer $\upsilon \in (\mathbb{Z}[\zeta]/(\Psi))^*$ such that
$$\sigma \coloneqq \varrho\,\upsilon^8 \bmod \Psi \text{ is prime in } \mathbb{Z}[\zeta]\,;$$
5. Return σ.

Verification On input σ, m and pk, Verify checks whether
$$\text{(i) } \sigma \text{ is prime}\,, \quad \text{(ii) } \mathrm{N}(\sigma) < 2^{\ell k}\,, \quad \text{(iii) } \mathfrak{I}^{(8)}_{\boldsymbol{n}}(\sigma) = H(m)$$
and, if so, accepts the signature.

Remark 2. The primes π_j's and ψ_j's must be chosen of norm of ℓ bits for an ℓ sized for the factoring problem over the rational integers. Indeed, suppose an attacker is given as a challenge $\nu = \pi\psi$, a product of two primes in $\mathbb{Z}[\zeta]$. The goal of the attacker is to recover π and ψ.

The norm of ν satisfies $\mathrm{N}(\nu) = \mathrm{N}(\pi)\,\mathrm{N}(\psi) \coloneqq pq$ for two ℓ-bit rational primes $p, q \equiv 1 \pmod 8$. If ℓ were chosen too small so that the problem of factoring the product of two rational ℓ-bit primes becomes feasible, the attacker could factor $\mathrm{N}(\nu)$ and recover p and q. Once p and q are found, its remaining task is to find $\pi, \psi \in \mathbb{Z}[\zeta]$ with $\mathrm{N}(\pi) = p$ and $\mathrm{N}(\psi) = q$. This can be efficiently achieved by generalizing Cornacchia's algorithm [6, Algorithm 1.5.2] to eighth roots, as done in [8, § 1.2] for cubic roots. The first step is to solve for r over $\mathbb{F}_p^*$ the equation $r^4 + 1 = 0 \pmod p$. Next, to consider the integer $\rho \coloneqq r - \zeta \in \mathbb{Z}[\zeta]$, whose norm is a multiple of p. Hence, the computation of $\gcd(\rho, p)$ yields $\pi \in \mathbb{Z}[\zeta]$—remember that $\mathbb{Z}[\zeta]$ is norm-Euclidean, and $p = \pi\pi_3\pi_5\pi_7$ where $\pi_k = \sigma_k(\pi)$. And similarly for q.

6.2 Evaluating Octic Residue Symbols

Octapus requires the evaluation of the 8^{th}-power residue symbol. We develop such an algorithm below. Algorithms for computing r^{th}-power residue symbols have only been devised for $r \in \{2,3,4,5,7\}$. See [47,8], [46,8], [39] and [5] for the cases $r = 3$, 4, 5 and 7, respectively. As noted in [5], as r grows, the technical details become increasingly complicated. An excellent account on the octic reciprocity can be found in [23, Chapter 9]. See also [16].

An element $\alpha = a_0 + a_1\zeta + a_2\zeta^2 + a_3\zeta^3 \in \mathbb{Z}[\zeta]$ is said to be *primary* if $\alpha \equiv 1 \pmod{2+2\zeta}$ or, equivalently, if

$$\begin{cases} a_0 + a_1 + a_2 + a_3 \equiv 1 \pmod 4\,, \\ a_1 \equiv a_2 \equiv a_3 \equiv 0 \pmod 2\,. \end{cases}$$

Proposition 2. *Let $\alpha \in \mathbb{Z}[\zeta]$ such that $(1+\zeta) \nmid \alpha$. Then there is a unit $\upsilon \in \mathbb{Z}[\zeta]$ such that $\alpha = \upsilon\,\alpha^*$ with α^* primary.*

Proof. Let $\alpha = a_0 + a_1\zeta + a_2\zeta^2 + a_3\zeta^3$. The condition $(1+\zeta) \nmid \alpha$ implies $a_0 + a_1 + a_2 + a_3 \equiv 1 \pmod 2$.

1. Suppose first that $a_0 \not\equiv a_2 \pmod 2$ (and thus $a_1 \equiv a_3 \pmod 2$). Noting that $\alpha \sim \alpha\,\zeta^{-2} = a_2 + a_3\zeta - a_0\zeta^2 - a_1\zeta^3$, we can assume that $a_0 \equiv 1 \pmod 2$ and $a_2 \equiv 0 \pmod 2$.
   (a) If $a_1 \equiv a_3 \equiv 0 \pmod 2$ then $\alpha = a_0 + a_1\zeta + a_2\zeta^2 + a_3\zeta^3$ with $a_0 \equiv 1 \pmod 2$ and $a_1 \equiv a_2 \equiv a_3 \equiv 0 \pmod 2$.
   (b) If $a_1 \equiv a_3 \equiv 1 \pmod 2$, we replace α with $\alpha\,\epsilon^{-1}$ and get

$$\alpha\,\epsilon^{-1} = \underbrace{(-a_0 + a_1 - a_3)}_{\equiv 1 \pmod 2} + \underbrace{(a_0 - a_1 + a_2)}_{\equiv 0 \pmod 2}\zeta + \underbrace{(a_1 - a_2 + a_3)}_{\equiv 0 \pmod 2}\zeta^2 + \underbrace{(-a_0 + a_2 - a_3)}_{\equiv 0 \pmod 2}\zeta^3 \ .$$

   By possibly multiplying by $-1 = \zeta^{-4}$ yields a primary element.
2. Suppose now that $a_0 \equiv a_2 \pmod 2$ (and $a_1 \not\equiv a_3 \pmod 2$). Then multiplying α by ζ^{-1} yields $\alpha\,\zeta^{-1} = a_1 + a_2\zeta + a_3\zeta^3 - a_0\zeta^3$. We so obtain a case similar to Case 1.

Consequently, in all cases, α can be expressed as $\alpha = \upsilon\,\alpha^*$ with α^* primary and $\upsilon = \zeta^k\epsilon^l$ for some $0 \le k \le 7$ and $l \in \{0,1\}$. □

The main result is the octic reciprocity law; see [23, Theorem 9.19].

Theorem 2 (Octic Reciprocity). *Let α and λ be co-prime primary elements of $\mathbb{Z}[\zeta]$. Let N_1, N_2 and N_3 respectively denote the relative norms of the extensions $\mathbb{Q}(\zeta)/\mathbb{Q}(i)$, $\mathbb{Q}(\zeta)/\mathbb{Q}(\sqrt{-2})$ and $\mathbb{Q}(\zeta)/\mathbb{Q}(\sqrt{2})$; and write $\mathrm{N}_1(\alpha) = a(\alpha)^2 + b(\alpha)^2$, $\mathrm{N}_2(\alpha) = c(\alpha)^2 + 2d(\alpha)^2$, $\mathrm{N}_3(\alpha) = e(\alpha)^2 - 2f(\alpha)^2$, and similarly for λ. Then*[8]

$$\left[\frac{\alpha}{\lambda}\right]_8 = \left[\frac{\lambda}{\alpha}\right]_8 (-1)^{\frac{\mathrm{N}(\alpha)-1}{8}\frac{\mathrm{N}(\lambda)-1}{8}}\zeta^{\frac{d(\lambda)f(\alpha)-d(\alpha)f(\lambda)}{4}} \ .$$

Moreover,

$$\left[\tfrac{1-\zeta}{\alpha}\right]_8 = \zeta^{\frac{5a-5+5b+18d+b^2-2bd+d^4/2}{8}}\,, \qquad \left[\tfrac{\zeta}{\alpha}\right]_8 = \zeta^{\frac{a-1+4b+2bd+2d^2}{4}}\,,$$
$$\left[\tfrac{1+\zeta}{\alpha}\right]_8 = \zeta^{\frac{a-1+b+6d+b^2+2bd+d^4/2}{8}}\,, \qquad \left[\tfrac{\epsilon}{\alpha}\right]_8 = \zeta^{\frac{d-3b-bd-2d^2}{2}}\,,$$
$$\left[\tfrac{1+\zeta+\zeta^2}{\alpha}\right]_8 = \zeta^{\frac{a-1-2b+2d-2d^2}{4}}\ .$$

□

[8] We note that a factor $-\frac{1}{4}$ is missing in the expression given in [23, Theorem 9.19].

Letting $\alpha = a_0 + a_1\zeta + a_2\zeta^2 + a_3\zeta^3$, a direct calculation shows that $\alpha_1\alpha_5 = (a_0{}^2 - a_2{}^2 + 2a_1a_3) + (-a_1{}^2 + a_3{}^2 + 2a_0a_2)i$, $\alpha_1\alpha_3 = (a_0{}^2 - a_1{}^2 + a_2{}^2 - a_3{}^2) + (a_0a_1 + a_0a_3 - a_1a_2 + a_2a_3)\sqrt{-2}$, and $\alpha_1\alpha_7 = (a_0{}^2 + a_1{}^2 + a_2{}^2 + a_3{}^2) + (a_0a_1 - a_0a_3 + a_1a_2 + a_2a_3)\sqrt{2}$ [23, Exerc. 5.21]. This yields $a(\alpha) = a_0{}^2 - a_2{}^2 + 2a_1a_3$, $b(\alpha) = -a_1{}^2 + a_3{}^2 + 2a_0a_2$,[9] $d(\alpha) = a_0a_1 + a_0a_3 - a_1a_2 + a_2a_3$, and $f(\alpha) = a_0a_1 - a_0a_3 + a_1a_2 + a_2a_3$.

As stated, the reciprocity law requires α and λ being primary. Suppose that α is such that $(1+\zeta) \nmid \alpha$, but is not necessarily primary. Then from Proposition 2, we can write $\alpha = \zeta^k\epsilon^l\,\alpha^*$ for some $0 \le k \le 7$ and $l \in \{0, 1\}$, with α^* primary. We note $\alpha^* = \text{primary}(\alpha)$ and $(k, l) = \nu(\alpha)$. Likewise, suppose that λ is such that $(1+\zeta) \nmid \lambda$ and is not necessarily primary. Then $\lambda = \zeta^{k'}\epsilon^{l'}\lambda^*$ with $\lambda^* = \text{primary}(\lambda)$ and $(k', l') = \nu(\lambda)$.

We assume $(1+\zeta) \nmid \lambda$. Putting all together, when $(1+\zeta) \nmid \alpha$, we have:

$$\begin{aligned}
\left[\frac{\alpha}{\lambda}\right]_8 &= \left[\frac{\alpha}{\lambda^*}\right]_8 = \left[\frac{\zeta^k}{\lambda^*}\right]_8\left[\frac{\epsilon^l}{\lambda^*}\right]_8\left[\frac{\alpha^*}{\lambda^*}\right]_8 && \text{by Proposition 2}\\
&= \zeta^{\frac{k\left(a(\lambda^*)-1+4b(\lambda^*)+2b(\lambda^*)d(\lambda^*)+2d(\lambda^*)^2\right)}{4}}\zeta^{\frac{l\left(d(\lambda^*)-3b(\lambda^*)-b(\lambda^*)d(\lambda^*)-2d(\lambda^*)^2\right)}{2}}\\
&\qquad \left[\frac{\lambda^*}{\alpha^*}\right]_8\zeta^{\frac{(\mathrm{N}(\alpha^*)-1)(\mathrm{N}(\lambda^*)-1)}{16}+\frac{d(\lambda^*)f(\alpha^*)-d(\alpha^*)f(\lambda^*)}{4}} && \text{by Theorem 2}\\
&= \left[\frac{\lambda^* \bmod \alpha^*}{\alpha^*}\right]_8\zeta^{k\,\mathcal{K}(\lambda^*)+l\,\mathcal{L}(\lambda^*)+\mathcal{J}(\alpha^*,\lambda^*)} \pmod 8
\end{aligned}$$

where $\mathcal{K}(\lambda^*) = \frac{1}{4}\left[a(\lambda^*)-1+4b(\lambda^*)+2b(\lambda^*)d(\lambda^*)+2d(\lambda^*)^2\right]$, $\mathcal{L}(\lambda^*) = \frac{1}{2}\left[d(\lambda^*)-3b(\lambda^*)-b(\lambda^*)d(\lambda^*)-2d(\lambda^*)^2\right]$ and $\mathcal{J}(\alpha^*, \lambda^*) = \frac{1}{16}\left[(\mathrm{N}(\alpha^*)-1)(\mathrm{N}(\lambda^*)-1) + 4d(\lambda^*)f(\alpha^*) - 4d(\alpha^*)f(\lambda^*)\right]$. When $(1+\zeta) \mid \alpha$, we have:

$$\begin{aligned}
\left[\frac{\alpha}{\lambda}\right]_8 &= \left[\frac{\alpha}{\lambda^*}\right]_8 = \left[\frac{\alpha/(1+\zeta)}{\lambda^*}\right]_8\left[\frac{1+\zeta}{\lambda^*}\right]_8\\
&= \left[\frac{\alpha/(1+\zeta)}{\lambda^*}\right]_8\zeta^{\mathcal{I}(\lambda^*)} \pmod 8 && \text{by Theorem 2}
\end{aligned}$$

where $\mathcal{I}(\lambda^*) = \frac{1}{8}\left(a(\lambda^*) - 1 + b(\lambda^*) + 6d(\lambda^*) + b(\lambda^*)^2 + 2b(\lambda^*)d(\lambda^*) + d(\lambda^*)^4/2\right)$. These two observations lead to Algorithm 2.

Algorithm 2: Computing $\left[\frac{\alpha}{\lambda}\right]_8$

Data: $\alpha, \lambda \in \mathbb{Z}[\zeta]$ with α and λ co-prime, and $(1+\zeta) \nmid \lambda$
Result: $\left[\frac{\alpha}{\lambda}\right]_8 \in \{\pm 1, \pm i, \pm\zeta, \pm i\zeta\}$
$\lambda \leftarrow \text{primary}(\lambda)$; $j \leftarrow 0$
while $\mathrm{N}(\alpha) \neq 1$ **do**
  if $(1+\zeta) \mid \alpha$ **then**
    $\alpha \leftarrow \alpha/(1+\zeta)$
    $j \leftarrow j + \mathcal{I}(\lambda) \pmod 8$
  else
    $(k, l) \leftarrow \nu(\alpha)$; $\alpha \leftarrow \text{primary}(\alpha)$
    $j \leftarrow j + k\,\mathcal{K}(\lambda) + l\,\mathcal{L}(\lambda) + \mathcal{J}(\alpha, \lambda) \pmod 8$
    $(\alpha, \lambda) \leftarrow (\lambda \bmod \alpha, \alpha)$
  end
end
$(k, l) \leftarrow \nu(\alpha)$; $\alpha \leftarrow \text{primary}(\alpha)$
$[u_0, u_1, u_2, u_3] \leftarrow \alpha \bmod 8$; $k \leftarrow k + u_0 - 1$; $l \leftarrow l + u_3$
$j \leftarrow j + k\,\mathcal{K}(\lambda) + l\,\mathcal{L}(\lambda) \pmod 8$
return ζ^j

[9] The first formula listed in [23, Exerc. 5.21] actually corresponds to $-b$.

At the end of the while-loop, α is transformed into a primary unit, say υ^*. Letting $\upsilon^* \bmod 8 = u_0 + u_1\zeta + u_2\zeta^2 + u_3\zeta^3 := [u_0, u_1, u_2, u_3]$, it turns out that the possible values are $[1,0,0,0]$, $[1,4,0,4]$, $[5,6,0,2]$, $[5,2,0,6]$, respectively corresponding to $\left[\frac{\upsilon^*}{\lambda^*}\right]_8 = \left[\frac{1}{\lambda^*}\right]_8$, $\left[\frac{\epsilon^4}{\lambda^*}\right]_8$, $\left[\frac{\zeta^4\epsilon^2}{\lambda^*}\right]_8$, $\left[\frac{\zeta^4\epsilon^6}{\lambda^*}\right]_8$.

The correctness of the algorithm is a consequence of the fact that $\mathbb{Z}[\zeta]$ is norm-Euclidean: when α is replaced by $\lambda \bmod \alpha$, its norm decreases. Also, when α is divided by $(1+\zeta)$, its norm is divided by 2 since $\mathrm{N}(1+\zeta) = 2$. Therefore, in all cases, the norm of α is decreasing and eventually becomes 1.

Remark 3. Letting $\alpha = a_0 + a_1\zeta + a_2\zeta^2 + a_3\zeta^3$, the condition $(1+\zeta) \mid \alpha$ simply amounts to verify whether $a_0 + a_1 + a_2 + a_3 \equiv 0 \pmod 2$; in this case, $\alpha/(1+\zeta) = \frac{1}{2}(a_0 + a_1 - a_2 + a_3) + \frac{1}{2}(-a_0 + a_1 + a_2 - a_3)\zeta + \frac{1}{2}(a_0 - a_1 + a_2 + a_3)\zeta^2 + \frac{1}{2}(-a_0 + a_1 - a_2 + a_3)\zeta^3$.

7 Concluding Remarks

In this paper, we have introduced a formal definition and construction of a new family of one-way functions and signature schemes. They are related to the hardness of factoring moduli of the form $n = p^r q$. Since our constructions rely on newly introduced assumptions, further cryptanalytic efforts are demanded in order to get more confidence about their exact security.

Acknowledgments The third author is grateful to Franz Lemmermeyer for providing a copy of [16].

References

1. Bach, E., Shallit, J.: Algorithmic Number Theory, Volume I: Efficient Algorithms. MIT Press (1996), https://mitpress.mit.edu/books/algorithmic-number-theory-volume-1
2. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: Denning, D.E., et al. (eds.) 1st ACM Conference on Computer and Communications Security. pp. 62–73. ACM Press (1993). doi:10.1145/168588.168596
3. BlueKrypt: Cryptographic key length recommendations (Jun 2018), https://www.keylength.com
4. Boneh, D., Durfee, G., Howgrave-Graham, N.: Factoring $N = p^r q$ for large r. In: Wiener, M.J. (ed.) Advances in Cryptology – CRYPTO '99. Lecture Notes in Computer Science, vol. 1666, pp. 326–337. Springer (1999). doi:10.1007/3-540-48405-1_21
5. Caranay, P.C., Scheidler, R.: An efficient seventh power residue symbol algorithm. International Journal of Number Theory **6**(8), 1831–1853 (2010). doi:10.1142/s1793042110003770
6. Cohen, H.: A Course in Computational Algebraic Number Theory, Graduate Texts in Mathematics, vol. 138. Springer (1993). doi:10.1007/978-3-662-02945-9
7. Damgård, I.: On the randomness of Legendre and Jacobi sequences. In: Goldwasser, S. (ed.) Advances in Cryptology – CRYPTO '88. Lecture Notes in Computer Science, vol. 403, pp. 163–172. Springer (1990). doi:10.1007/0-387-34799-2_13
8. Damgård, I.B., Frandsen, G.S.: Efficient algorithms for the gcd and cubic residuosity in the ring of Eisenstein integers. Journal of Symbolic Computation **39**(6), 643–652 (2005). doi:10.1016/j.jsc.2004.02.006
9. Davenport, H.: On the distribution of quadratic residues (mod p). Journal of the London Mathematical Society **s1-6**(1), 49–54 (1931). doi:10.1112/jlms/s1-6.1.49
10. Davenport, H.: On the distribution of quadratic residues (mod p). Journal of the London Mathematical Society **s1-8**(1), 46–52 (1933). doi:10.1112/jlms/s1-8.1.46
11. Diffie, W., Hellman, M.: New directions in cryptography. IEEE Transactions on Information Theory **22**(6), 644–654 (1976). doi:10.1109/TIT.1976.1055638
12. Ding, C., Pei, D., Salomaa, A.: Chinese Remainder Theorem: Applications in Computing, Coding, Cryptography. World Scientific (1996). doi:10.1142/9789812779380_0004
13. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) Advances in Cryptology – CRYPTO '86. Lecture Notes in Computer Science, vol. 263, pp. 186–194. Springer (1987). doi:10.1007/3-540-47721-7_12

14. Fujioka, A., Okamoto, T., Miyaguchi, S.: ESIGN: An efficient digital signature implementation for smart cards. In: Davies, D.W. (ed.) Advances in Cryptology – EUROCRYPT '91. Lecture Notes in Computer Science, vol. 547, pp. 446–457. Springer (1991). doi:10.1007/3-540-46416-6_38
15. Goldreich, O.: Foundations of Cryptography, Volume 1: Basic Tools. Cambridge University Press (2001). doi:10.1017/CBO9780511546891
16. Goldscheider, F.: Das Reziprozitätsgesetz der achten Potenzreste. Wissenschaftliche Beilage zum Programm des Luisenstädtischen Realgymnasiums **96**, 1–29 (1889), https://zbmath.org/?q=an%3A21.0178.02
17. Goldwasser, S., Micali, S., Rivest, R.L.: A digital signature scheme secure against adaptive chosen-message attacks. SIAM Journal of Computing **17**(2), 281–308 (1988). doi:10.1137/0217017
18. Goubin, L., Mauduit, C., Sárközy, A.: Construction of large families of pseudo-random binary sequences. Journal of Number Theory **106**(1), 56–69 (2004). doi:10.1016/j.jnt.2003.12.002
19. Granboulan, L.: How to repair ESIGN. In: Cimato, S., Persiano, G., Galdi, C. (eds.) Security in Communication Networks (SCN 2002). Lecture Notes in Computer Science, vol. 2576, pp. 234–240. Springer (2003). doi:10.1007/3-540-36413-7_17
20. Ireland, K., Rosen, M.: A Classical Introduction to Modern Number Theory, Graduate Texts in Mathematics, vol. 84. Springer, 2nd edn. (1990). doi:10.1007/978-1-4757-2103-4
21. Katz, J.: Digital Signatures. Springer (2010). doi:10.1007/978-0-387-27712-7
22. Lemmermeyer, F.: The Euclidean algorithm in algebraic number fields. Expositiones Mathematicae **13**(5), 385–416 (1995), http://www.rzuser.uni-heidelberg.de/~hb3/publ/survey.pdf, updated version, Feb. 14, 2004
23. Lemmermeyer, F.: Reciprocity Laws: From Euler to Eisenstein. Springer Monographs in Mathematics, Springer (2000). doi:10.1007/978-3-662-12893-0
24. Lenstra, A.K., Lenstra, Jr., H.W. (eds.): The Development of the Number Field Sieve, Lecture Notes in Mathematics, vol. 1554. Springer (1993). doi:10.1007/BFb0091534
25. Lenstra, A.K., Lenstra, Jr., H.W., Lovász, L.: Factoring polynomials with rational coefficients. Mathematische Annalen **261**(4), 515–534 (1982). doi:10.1007/BF01457454
26. Lenstra, Jr., H.W.: Euclid's algorithm in cyclotomic fields. Journal of the London Mathematical Society (2) **10**(4), 457–465 (1975). doi:10.1112/jlms/s2-10.4.457
27. Lenstra, Jr., H.W.: Factoring integers with elliptic curves. Annals of Mathematics **126**(3), 649–673 (1987). doi:10.2307/1971363
28. Manohar, N., Fisch, B.: Factoring $n = p^2q$. Final project report CS359C, Stanford University (2017), https://crypto.stanford.edu/cs359c/17sp/projects/NathanManoharBenFisch.pdf
29. May, A.: Secret exponent attacks on RSA-type schemes with moduli $n = p^r q$. In: Bao, F., Deng, R.H., Zhou, J. (eds.) Public Key Cryptography – PKC 2004. Lecture Notes in Computer Science, vol. 2947, pp. 218–230. Springer (1999). doi:10.1007/978-3-540-24632-9_16
30. Menezes, A., Qu, M., Stinson, D., Wang, Y.: Evaluation of security level of cryptography: ESIGN signature scheme. External Evaluation Report ex-1053-2000, CRYPTREC (Jan 15, 2001), https://www.cryptrec.go.jp/exreport/cryptrec-ex-1053-2000.pdf
31. Okamoto, T., Fujisaki, E., Morita, H.: TSH-ESIGN: Efficient digital signature scheme using trisection size hash. Submission to IEEE P1363a (Nov 1998), http://security.nknu.edu.tw/crypto/tsh_esign.pdf, [Online; accessed 7-February-2019]
32. Okamoto, T., Uchiyama, S.: A new public-key cryptosystem as secure as factoring. In: Nyberg, K. (ed.) Advances in Cryptology – EUROCRYPT '98. Lecture Notes in Computer Science, vol. 1403, pp. 308–318. Springer (1998). doi:10.1007/BFb0054135
33. Okamoto, T., Shibaishi, A.: A fast signature scheme based on quadratic inequalities. In: 1985 IEEE Symposium on Security and Privacy. pp. 123–133. IEEE Computer Society (1985). doi:10.1109/SP.1985.10026
34. Peralta, R.: On the distribution of quadratic residues and nonresidues modulo a prime number. Mathematics of Computation **58**(197), 433–440 (1992). doi:10.1090/S0025-5718-1992-1106978-9
35. Peralta, R., Okamoto, E.: Faster factoring of integers of a special form. IEICE Transactions on Fundamentals of Electronics, Communications, and Computer Sciences **E79-A**(4), 489–493 (1996), http://www.cs.yale.edu/homes/peralta/papers/SpeedEcm.ps
36. Rivest, R.L., Shamir, A., Adleman, L.M.: A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM **21**(2), 120–126 (1978). doi:10.1145/359340.359342
37. Sárközy, A., Stewart, C.L.: On pseudorandomness in families of sequences derived from the Legendre symbol. Periodica Mathematica Hungarica **54**(2), 163–173 (2007). doi:10.1007/s-10998-007-2163-9
38. Sato, H., Takagi, T., Tezuka, S., Takaragi, K.: Generalized powering functions and their application to digital signatures. In: Laih, C.S. (ed.) Advances in Cryptology – ASIACRYPT 2003. Lecture Notes in Computer Science, vol. 2894, pp. 443–451. Springer (2003). doi:10.1007/978-3-540-40061-5_28

39. Scheidler, R., Williams, H.C.: A public-key cryptosystem utilizing cyclotomic fields. Designs, Codes and Cryptography **6**(2), 117–131 (1995). doi:10.1007/BF01398010
40. Schmidt-Samoa, K.: A new Rabin-type trapdoor permutation equivalent to factoring. Electronic Notes in Theoretical Computer Science **157**(3), 79–94 (2006). doi:10.1016/j.entcs.2005.09.039
41. Schmidt-Samoa, K., Takagi, T.: Paillier's cryptosystem modulo p^2q and its applications to trapdoor commitment schemes. In: Dawson, E., S, V. (eds.) Progress in Cryptology – Mycrypt 2005. Lecture Notes in Computer Science, vol. 3715, pp. 296–313. Springer (2005). doi:10.1007/11554868_21
42. Schnorr, C.P.: Efficient signature generation by smart cards. Journal of Cryptology **4**(3), 161–174 (1991). doi:10.1007/BF00196725
43. Stern, J., Pointcheval, D., Malone-Lee, J., Smart, N.P.: Flaws in applying proof methodologies to signature schemes. In: Yung, M. (ed.) Advances in Cryptology – CRYPTO 2002. Lecture Notes in Computer Science, vol. 2442, pp. 93–110. Springer (2002). doi:10.1007/3-540-45708-9_7
44. Takagi, T.: Fast RSA-type cryptosystem modulo p^kq. In: Krawczyk, H. (ed.) Advances in Cryptology – CRYPTO '98. Lecture Notes in Computer Science, vol. 1462, pp. 318–326. Springer (1998). doi:10.1007/BFb0055738
45. Washington, L.C.: Introduction to Cyclotomic Fields, Graduate Texts in Mathematics, vol. 83. Springer, 2nd edn. (1997). doi:10.1007/978-1-4612-1934-7
46. Weilert, A.: Fast computation of the biquadratic residue symbol. Journal of Number Theory **96**(1), 133–151 (2002). doi:10.1006/jnth.2002.2783
47. Williams, H.C.: An M^3 public-key encryption scheme. In: Williams, H.C. (ed.) Advances in Cryptology – CRYPTO '85. Lecture Notes in Computer Science, vol. 218, pp. 358–368. Springer (1986). doi:10.1007/3-540-39799-X_26