New Factoring-Based Cryptographic Primitives

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

[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
ferradi.houda@lab.ntt.co.jp
[3] DIENS, École normale supérieure, CNRS, PSL University
45 rue d'Ulm, 75230, Paris CEDEX 05, France
david.naccache@ens.fr

Abstract. This paper describes new p^2q-based OWFs and 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.

The signature algorithms are derived from a new OWF whose inversion is as provably as difficult as factoring p^2q. By opposition to the DLP, Rabin or RSA, which assume that the target modulus is built into the OWF, the new OWF does not require any built-in parameters except the modulus' size.

In a first (polynomial-time but impractical) signature scheme the signer generates $k \simeq 200$ 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_is match the message digest.

In a second variant, the resulting public-key is 300 times longer than RSA's and a typical signature is 1600-bytes long.

Given of their very different design the new signature schemes seem to be an overlooked "missing species" in the corpus of known signature algorithms.

We stress that we did not manage to prove the security of the proposed signature schemes nor find any attacks against them.

1 Introduction

To construct secure signature scheme or public-key cryptosystem, one fundamental building block is *one-way function*. 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, theoretically that means there cannot exist a probabilistic (or deterministic) machine that can invert f in polynomial time. It is conjectured that the existence of a OWF implies that $\mathcal{P} \neq \mathcal{NP}$. Conversely, in the current state of complexity theory (i.e.,

$\mathcal{P} = \mathcal{NP}$?) it still unknown whether $\mathcal{P} \neq \mathcal{NP}$ implies the existence of OWFs. For that reason even if some OWF candidates (e.g., the Discrete Logarithm Problem or Factoring) are known to be NP-complete, this does not necessarily imply their one-wayness.

1.1 Cryptography Modulo p^2q

p^2q moduli have found a few applications in cryptography since the mid 1980s. The most notable of which are probably the EISGN signature scheme and its variants [OS85,FOM91,OFM98,Gra03,SPMLS02], Okamoto–Uchiyama's cryptosystem [OU98,SST05], Schmidt-Samoa's cryptosystem [SS06] or constructions such as [STTT03].

As stated in [???08,MQSW08]...[compléter]

Reste aussi à citer les autres réf qui sont dans mers.bib dans leur contexte.

Using p^2q moduli does not seem to render factoring significantly easier. [BDHG99] shows that it is easy to factor $N = p^r q$ when $r \simeq \log p$. This LLL-based approach [LLL82] does not apply to the context of this paper where $r = 2$ (moduli of the form $p^r q$ are rather rarely used in cryptographic constructions, e.g. [Tak98]). We also refer the reader to [May04].

[MF17] presents two different approaches to factor p^2q. The first approach relies on Coppersmith's method [Cop97] and factors p^2q in $O(q^{0.31})$ time. As a second approach, uses [BDHG99] and achieves $O(\sqrt[3]{q})$ assuming that p and q are of the same size. Both approaches lag far behind the GNFS.

2 Preliminaries & Notations

We start by recalling notations used in this paper.

2.1 Number-Theoretic Definitions

$\mathbb{P}$ will stand for set of primes. To distinguish the first primes from the large moduli p_i used in this paper we will denote $\bar{p}_1 = 2, \bar{p}_2 = 3, \bar{p}_3 = 5, \ldots$

$\mathbb{P}[a, b]$ will denote the set $\{\bar{p}_a, \bar{p}_{a+1}, \ldots, \bar{p}_b\}$.

$k\#$ will represent the product of the first k primes starting with $\bar{p}_1 = 2$.

$\texttt{NextPrime}(x)$ is the function associating to $x \in \mathbb{N}$ the smallest greater or equal to x.

A boldface variable $\boldsymbol{x}$ will denote a set of elements identified by that variable, i.e. $\boldsymbol{x} = (x_0, \ldots, x_{k-1})$.

Definition 1 (Quadratic Residues). *Let $p \in \mathbb{P}$ be odd. $a \in \mathbb{N}$ is a quadratic residue modulo p ($a \in QR_p$) if a is congruent to a perfect square modulo p. Otherwise, a is a quadratic nonresidue modulo p ($a \in QNR_p$).*

Definition 2 (Legendre Symbol). *Let $p \in \mathbb{P}$ be odd. The Legendre symbol is a function of $a \in \mathbb{N}$ and p defined as*

$$\left(\frac{a}{p}\right) = \begin{cases} 1 & \text{if } a \in QR_p \text{ and } a \not\equiv 0 \bmod p, \\ -1 & \text{if } a \in QNR_p, \\ 0 & \text{if } a \equiv 0 \bmod p. \end{cases}$$

☞ Computing a Legendre symbol is simple:

$$\left(\frac{a}{p}\right) \equiv a^{\frac{p-1}{2}} \bmod p$$

Several other arithmetic rules and identities allow to considerably speed-up computation and avoid resorting to a modular exponentiation.

The Jacobi symbol is a natural generalization of the Legendre symbol:

Definition 3 (Jacobi Symbol). *$\forall (a, n) \in \mathbb{N}^2$, the Jacobi symbol (a/n) is defined as the product of the Legendre symbols corresponding to the prime factors of n:*

$$\left(\frac{a}{n}\right) = \prod_{i=1}^{k} \left(\frac{a}{p_i}\right)^{\alpha_i}$$
$$\text{where } n = \prod_{i=1}^{k} p_i^{\alpha_i} \text{ is the prime factorization of } n$$

☞ Following the normal convention for the empty product, $(a/1) = 1$.

☞ Interestingly, factoring n is not required for computing (a/n).

☞ Legendre and Jacobi symbols are indistinguishable when n is an odd prime.

☞ The Legendre symbol allows to check if $a \in \mathrm{QR}_p$, whereas the Jacobi symbol does not allow checking this property.

Definition 4 (Jacobi Imprint). *For $a \in \mathbb{N}$ and $\boldsymbol{n} \in \mathbb{N}^k$, the Jacobi Imprint $\mathcal{J}_{\boldsymbol{n}}(a) \in \mathbb{N}$ is the integer:*

$$\mathcal{J}_{\boldsymbol{n}}(a) = 2^{k-1} - \frac{1}{2} + \sum_{i=0}^{k-1} 2^{i-1} \left(\frac{a}{n_i}\right)$$

In essence, the imprint is an integer formed of bits representing the sequence of Jacobi symbols where -1s are replaced by 0s and 1s are left unchanged[4].

Jacobi imprints are usually considered to be pseudo-random [SS07] and were used as such in a handful of cryptographic constructions (e.g. [**?**]). At times we will indistinctively use $\mathcal{J}_{\boldsymbol{n}}(a)$ to design the integer $\mathcal{J}_{\boldsymbol{n}}(a)$ or the set of its bits.

Letting $\boldsymbol{n} = (n_0, \ldots, n_{k-1}) \in \mathbb{N}^k$ and $\boldsymbol{a} = (a_0, \ldots, a_{k-1}) \in \mathbb{N}^k$ we denote by $\mathsf{CRT}(\boldsymbol{a}, \boldsymbol{n})$ the Chinese Remainder function returning the smallest $a \in \mathbb{N}$ such that $a_i = a \bmod n_i$ for $0 \leq i \leq k-1$.

[4] We ignore the case 0 inapplicable to the constructions described in this paper.

Definition 5 (General Residues Problem (GRP)). *Given $k \in \mathbb{N}$ and a binary sequence $\boldsymbol{s} = (s_0, \ldots, s_{k-1})$ find the smallest $x \in \mathbb{N}$ such that:*

$$\mathcal{J}_{\mathbb{P}[1,k]}(x) = \boldsymbol{s}$$

If we are not interested in the smallest possible x then longer solutions can be constructed efficiently. To do so, start by generating[5] the k imprints $\mathcal{J}_{\mathbb{P}[1,k]}(r_i)$ for small prime r_is and express the target $\boldsymbol{s}$ as a xor of the $\mathcal{J}_{\mathbb{P}[1,k]}(r_i)$s using linear algebra modulo 2.

Because linear algebra results in a solution of Hamming density $\frac{1}{2}$, the expected solution will be of size $\simeq \sqrt{\#k}$.

houda

David, c'est quoi le role de ce probleme dans le papier?

Definition 6 (Approximate GCD Problem (AGCDP)). *Given a set of n integers of the form $x_i = q_i p + r_i$, where $p \in \mathbb{Z}$ and $q_i, r_i \xleftarrow{\$} \mathbb{Z}$. Find p.*

2.2 Cryptographic Definitions

We remind the following cryptographic notions and conventions:

Classically, $x \xleftarrow{\$} S$ denotes an x uniformly drawn from the set S.

λ denotes a security parameter (all the other parameters are function of λ).

PPT stands for probabilistic algorithms running in polynomial time.

$H : \{0,1\}^* \to \{0,1\}^k$ will denote a public hash-function. Typically, $k = 200$.

Definition 7 (Negligibility). *A function in n is negligible, denoted by $\mathsf{negl}(n)$, if $\forall p(x) \in \mathbb{Z}[x]$, $\exists N \in \mathbb{N}$, such that, $\forall n \geq N \Rightarrow \mathsf{negl}(n) \leq \frac{1}{p(n)}$.*

Definition 8 (One-Way Function (OWF)). *A polynomial-time computable function $f : \{0,1\}^* \to \{0,1\}^n$ is a one-way function, if $\forall \mathcal{A} \in PPT$ there is a negligible function negl such that : $\Pr_{x \in \{0,1\}^n}[\mathsf{Invert}_{\mathsf{Adv},f}(n) = 1] \leq \mathsf{negl}(n)$, where $\mathsf{Invert}_{\mathsf{Adv},f}(n)$ is the following experiment:*

$\mathsf{Invert}_{\mathsf{Adv},f}(n)$:
$x \xleftarrow{\$} \{0,1\}^n$
$y \leftarrow f(x)$
$\tilde{x} \leftarrow \mathcal{A}(1^\kappa, y)$
If $f(\tilde{x}) = y$ return(1) else return(0)

☞ A OWF f is a one-way permutation if f is bijective and length-preserving.

[5] for $0 \leq i \leq k-1$

2.3 Security model

We recall the strong [6] EUF-CMA security notion:

Definition 9 (Strong EUF-CMA Security). *A signature scheme Σ is* secure against existential forgeries in a chosen-message attack *(strongly* EUF-CMA*-secure) if the advantage of any PPT adversary $\mathcal{A}$ against the* EUF-CMA *game defined in Figure 1 is negligible:* $\mathsf{Adv}^{\mathsf{EUF}}_{\mathcal{A},\Sigma}(\kappa) = \Pr\left[\mathsf{EUF}^{\mathcal{A}}_{\Sigma}(\kappa) = 1\right] \in \mathsf{negl}(\kappa)$.

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

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

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

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

3 A Provably Secure One-Way Function

The first contribution of this paper is a new, provably secure, OWF. To understand the intuition behind the proposed function we build it in three steps. Each version perfects the previous and addresses a specific design limitation.

houda
David est ce que tu peux un nom au fcts F_1 et F_2 c'est plus simple pour l'appeler plus tard dans le papier

3.1 $\mathcal{F}_1$: A OWF Processing Prime Inputs

Algorithm 1 The First Version $\mathcal{F}_1$
Input: $x \in \mathbb{P}$ and $k \in \mathbb{N}$
Output: $s \in \mathbb{N}$
  $s \leftarrow \mathcal{J}_{\mathbb{P}[1,k]}(x)$
  return s

$\mathcal{F}_1$ operates on prime inputs only and simply return the imprint:

$$\mathcal{F}_1(x) = \mathcal{J}_{\mathbb{P}[1,k]}(x) = (\left(\frac{x}{2}\right), \left(\frac{x}{3}\right), \ldots, \left(\frac{x}{\bar{p}_{k-1}}\right))$$

[6] In contrast to the *weak* version, the adversary is allowed to forge for a message that they have queried before, provided that their forgery is *not* an oracle response.

for some parameter k that will be precised later.

Assume that we are given an attacker $m = \mathcal{A}_1(\mathcal{F}_1(m))$ inverting $\mathcal{F}_1$ for a k which is large enough to characterize m beyond any reasonable doubt (a notion formalized later), we use $\mathcal{A}_1$ to factor any $n = p^2q$ by calling:

$$\mathcal{A}_1(\mathcal{F}_1(n)) = \mathcal{A}_1(\mathcal{J}_{\mathbb{P}[1,k]}(n)) = \mathcal{A}_1(\mathcal{J}_{\mathbb{P}[1,k]}(p^2q)) = \mathcal{A}_1(\mathcal{J}_{\mathbb{P}[1,k]}(q)) = q$$

Indeed $\mathcal{F}_1(n) = \mathcal{F}_1(p^2q) = \mathcal{F}_1(q)$ and because we assume that k suffices to characterize q, $\mathcal{A}$ has no choice but to factor the target n.

3.2 $\mathcal{F}_2$: A OWF Processing All Inputs

Algorithm 2 The Second Version $\mathcal{F}_2$

Input: $x \in \mathbb{N}$, $k \in \mathbb{N}$
Output: $s \in \mathbb{N}$
$\hat{x} \leftarrow \texttt{NextPrime}(x)$
$t \leftarrow \hat{x} - x$
$s \leftarrow \mathcal{J}_{\bar{\mathbf{p}}[\boldsymbol{kt},\boldsymbol{k(t+1)-1}]}(\hat{x}) = (\left(\frac{\hat{x}}{\bar{p}_{kt}}\right), \ldots, \left(\frac{\hat{x}}{\bar{p}_{k(t+1)-1}}\right))$
return s

A OWF operating only on prime inputs is of little use. We hence extend $\mathcal{F}_1$ to a function $\mathcal{F}_2$ operating on all inputs while preserving the existence of $\mathcal{A}_1$.

Let $\hat{x} = \texttt{NextPrime}(x)$ and $t = \hat{x} - x \geq 0$.

We modify the OWF's definition to:

$$\mathcal{F}_2(x) = (\left(\frac{\hat{x}}{\bar{p}_{kt}}\right), \ldots, \left(\frac{\hat{x}}{\bar{p}_{k(t+1)-1}}\right))$$

We see that $x \in \mathbb{P} \Rightarrow \mathcal{F}_1(m) = \mathcal{F}_2(m)$. When $x \notin \mathbb{P}$ then $\mathcal{F}_2$ produces a shifted output sequence.

Assuming that we are given an $\mathcal{A}_2$ inverting $\mathcal{F}_2$, we use $\mathcal{A}_2$ to factor p^2q, relying only on the $\mathcal{A}_2$'s capacity act as $\mathcal{A}_1$ on prime inputs.

3.3 $\mathcal{F}_3$: Quantifying "... Beyond Any Reasonable Doubt..."

It remains to fix the function's only parameter k.

Let $k = \Delta + \log_2 q$ for some $\Delta \in \mathbb{N}$.

[faire le calcul d'entropie de H(q|)]

4 A New Signature Scheme

We are now ready to formally describe a first signature scheme.

4.1 Definition and Security of a new Signature

Definition 10. *Our signature scheme is a tuple of algorithms (***KeyGen***,* **Sign***, and* **Ver***), which we define as follows:*

- **KeyGen**(pp)*: The key generation algorithm* **KeyGen** *takes as input the security parameter* 1^κ *and outputs a secret key* $\mathsf{sk} = \{p_i, q_i\}$ *and a public key* $\mathsf{pk} = \boldsymbol{q}$ *computed as follows:*
  The signer generates k *public moduli* $n_i = p_i^2 q_i$ *while keeping their factors secret. For the sake of simplicity, we assume that all secret factors (i.e. the* p_i *and the* q_i*) are* ℓ*-bits long.*
- **Sign**$(m, \mathsf{sk} = \boldsymbol{q})$*: The signing algorithm* **Sign** *takes as inputs the message* m *and the secret key* $\boldsymbol{q}$ *and proceed as follows:*
  The signer hashes $H(m) = (h_0, \ldots, h_{k-1}) \in \{0,1\}^k$ *and picks* k *random* ℓ*-bit integers* r_i *such:*

$$2h_i - 1 = \left(\frac{r_i}{q_i}\right) \quad \textit{for} \quad 0 \leq i \leq k+1$$

  Then, the signer generates an ℓ*-bit random* $\rho \in \mathbb{N}$ *such that:*

$$s = \mathit{CRT}(\boldsymbol{r}, \boldsymbol{q}) + \rho\pi \in \mathbb{P} \quad \textit{where} \quad \pi = \prod_{i=0}^{k-1} q_i$$

  And returns s *as the signature of* m*.*
- **Ver**$(s, m, \mathsf{pk} = \boldsymbol{n})$*: To verify a signature* s*, the verification algorithm* **Ver** *takes as inputs the message* m*, the public key* $\mathsf{pk} = \boldsymbol{n}$ *and proceeds as follows: Return 1 if a signature* s *satisfies three criteria:*

$$s \in \mathbb{P} \quad \textit{and} \quad s < 2^{(k+1)\ell+1} \quad \textit{and} \quad \mathcal{J}_{\boldsymbol{n}}(s) = H(m)$$

  , 0 otherwise.

Correctness: Reste à faire.

4.2 Security proof

Todo: Existential unforgeability under adaptive chosen message attacks assumptions: $n = p^2 q$ and H is collision-resistant.

Moi j'ai aucune idée. Eric ou Houda prenez en charge sinon il n'y qu'à dire que prouver la sécurité est un problème ouvert.

houda

David il faut plutot faire une preuve de securité par reduction: si un attaquant arrive à casser la signature dans le modele UFCMA (forge existentielle sous une attaque à messages choisis) alors arrive à resoudre à inverser la fonction à sens unique F_1.

Theorem 1 (Existential unforgeability). *Our scheme is provably* EUF-CMA*-secure assuming the hardness of inverting the* F_1*, in the ROM.*

Proof. To prove this result, we will show a reduction from an efficient EUF-CMA forger to an efficient F_1 inverter. For this goal we first show a sequence of indistinguishability results between the output distributions of:

- h
- g

4.3 Toy Example ($k = 8$)

Picking the secret primes:

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

We get the public moduli:

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

and the value:

$$\pi = \prod_{i=0}^{7} q_i = 962535482083430844430189085476678516 1$$

Consider a message whose digest is $\{h_0, \ldots, h_7\}$ and pick as r_is:

	$i=0$	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$
h_i	0	1	0	0	1	0	0	1
r_i	64863	58999	47120	50684	37458	57079	43135	56942

We get:

$$\mathsf{CRT}(\boldsymbol{r}, \boldsymbol{q}) = 139578625155923187878976453585864 1198$$

And by selecting $\rho = 56195$ we obtain the signature:

$$s = \mathsf{CRT}(\boldsymbol{r}, \boldsymbol{q}) + \rho\pi = 540898209943035522259423546348155350763593 \in \mathbb{P}$$

5 Shorter Signatures Without Prime Generation

5.1 Key Generation, Signing and Verifying

Definition 11. *We define the Shorter Signatures Without Prime Generation as follows:*

- **KeyGen**(1^κ)*: As in Section 4.1. In addition, the signer generates a secret* $\mathsf{sk} = w$ *and publishes* $\eta = \pi - w$ *as part of the public-key* pk*. The size of* w *will be described later.*
- **Sign**(m, w, r_i)*: The signer hashes* $H(m) = (h_0, \ldots, h_{k-1}) \in \{0,1\}^k$ *and constructs* k *random* ℓ*-bit integers* r_i *(the choice of the* r_i *will be precised later, to ease understanding the reader can temporarily assume that the* r_i *are random).*

  Using linear algebra modulo 2 find a subset of the $\mathcal{J}_{\boldsymbol{n}}(r_i)$ *having* $H(m)$ *as a xor*[7]*:*

$$\varepsilon_0\ \mathcal{J}_{\boldsymbol{n}}(r_0) \oplus \varepsilon_1\ \mathcal{J}_{\boldsymbol{n}}(r_1) \oplus \ldots \oplus \varepsilon_{k-1}\ \mathcal{J}_{\boldsymbol{n}}(r_{k-1}) = H(m) \quad \textit{where} \quad \varepsilon_i \in \{0,1\}$$

[7] If a solution $\varepsilon_0, \ldots, \varepsilon_{k-1}$ does not exist, refresh the r_i as necessary.

The signature is:

$$s = w + \tau = w + \prod_{\varepsilon_i = 1} r_i$$

– $\mathbf{Ver}(m, \mathsf{pk} = \eta)$*: To be valid, a signature* s *must satisfy two criteria:*

$$s < 2^{k\ell+1} \quad \textit{and} \quad \mathcal{J}_{\boldsymbol{n}}(\eta + s) = H(m)$$

Selecting w and the r_i The choice of w and the r_i must take into account several considerations.

Avoid small factors in the r_i: First of all, note that if the r_i are chosen randomly then half of them are expected to be even. Consequently, τ is expected to end by $\cong k/4$ trailing zeros and reveal the $k/4$ bits of w. Similarly, divisibility by $3, 5, 7, \ldots$ is expected to reveal more information on w. This means that if countermeasures are not implemented, π is not protected by the classical approximate GCD problem but by a modified problem in which the attacker knows the LSB and the MSB of π whose middle bits remain masked. To prevent this leakage, one can select $r_i \in \mathbb{P}$ but this slows-down the signature process although, as we will see next, we recommend to take $\ell \simeq 200$. A heuristic way to limit the leakage consists in constructing the r_i such that $\gcd(r_i, u\#) = 1$ for some bound u.

Avoid collisions between the r_i: The size of the r_is is also important. Note that the difference between two signatures (in $\mathbb{Z}$) yields an integer of the form:

$$\zeta = \prod_{\varepsilon'_i = 1} r'_i - \prod_{\varepsilon_i = 1} r_i$$

If, for some i, j, we obtain a collision $r'_i = r_j$ then by factoring (or gcd-ing) ζs obtained from different signatures the attacker can progressively discover the r_is and retrieve w (and hence π). We hence require the r_is to be sufficiently long so as not collide and recommend $\ell \simeq 200$.

Another security parameter is the hamming weight of $\boldsymbol{\varepsilon}$. Because half of the ε_is are null, τ is expected to be $\cong k\ell/2$ bits long. Because in the event where $\tau < w$ information about w will leak (the MSBs of w), here as well the signer must ascertain that $\tau > w$ before releasing the signature.

Finally, a direct attack on the key consists in averaging signatures (interestingly, even for diverse messages). Indeed, assuming that the r_i are random:

$$\lim_{t \to \infty} \overline{s}_t = \lim_{t \to \infty} \frac{1}{t} \sum_{i=1}^{t} s_i = w + \frac{(2^\ell - 1)^k}{2} = \mu$$

We do not regard this as a threat given that convergence is very slow:

$$\Pr(|\overline{s}_t - \mu| < \varepsilon) = 1 - \Pr(|\overline{s}_t - \mu| \geqslant \varepsilon) \geqslant 1 - \frac{\sigma^2}{t\varepsilon^2} \simeq 1 - \frac{2^{2k\ell}}{12\ t\ \varepsilon^2}$$

To reveal w's j-th digit we need $\varepsilon = 2^{j-1}$ and hence:

$$\frac{2^{2k\ell}}{12t2^{2(j-1)}} \simeq 1 \Rightarrow \log_2 t \simeq 2(\ell k - j) - 1.58$$

In other words, an exponential number of signatures is required to filter-out a linear number of bits of w.

[Houda, Eric, refaites les calculs et vérifiez ça SVP. Je pense que pour la borne sup il faut à mon avis utiliser la Zelen's inequality:

$$\Pr(X - \mu \geq u\sigma) \leq \left[1 + u^2 + \frac{\left(u^2 - u\theta_3 - 1\right)^2}{\theta_4 - \theta_3^2 - 1}\right]^{-1}$$

$$\text{with } \; u \geq \frac{\theta_3 + \sqrt{\theta_3^2 + 4}}{2}, \;\; \text{and } \; \theta_m = \frac{\eta_m}{\sigma}$$

where η_m is the m-th moment. In the present case (uniform distribution), skewness (and hence θ_3) is 0 which yields $u \geq 1$ and:

$$\Pr(X - \mu \geq u\sigma) \leq \left[1 + u^2 + \frac{\left(u^2 - 1\right)^2}{\theta_4 - 1}\right]^{-1}$$

$$\eta_4 = \frac{6(b-a+1)^2 + 1}{5(b-a+1)^2 - 1} = \frac{6(2^\ell) + 1}{5(2^\ell) - 1} \simeq \frac{6}{5}$$

$$\sigma = \frac{(b-a+1)^2 - 1}{12} = \frac{2^{2k\ell} - 1}{12} \Rightarrow \theta_4 = \frac{\eta_4}{\sigma} = \frac{6}{5} \times \frac{12}{2^{2k\ell} - 1} \simeq 0$$

$$\Pr(X - \mu \geq u\sigma) \leq \left[1 + u^2 + \left(u^2 - 1\right)^2\right]^{-1} = \frac{1}{u^2(3 - u^2)}$$

]

5.2 Toy Example ($k = 8$)

Consider the same $\boldsymbol{p}, \boldsymbol{q}, \boldsymbol{n}, \pi, \boldsymbol{h}$ as in the previous example.

We select $w = \texttt{91116}$ hence:

$$\eta = \pi - w = \texttt{962535482083430844430189085476669404 5}$$

Picking the following r_is:

	$i=0$	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$
r_i	13	31	47	17	91	57	23	67

We get the imprints $\mathcal{J}_{\boldsymbol{n}}(r_j)$ matrix I below [8]:

[8] j is indexing the r_j and i is indexing the n_i

	$i=0$	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$	$\mathcal{J}_{\boldsymbol{n}}(r_j)$
$j=0$	0	0	0	0	0	1	1	0	01001100
$j=1$	0	1	0	0	1	0	1	1	01001011
$j=2$	0	0	1	0	1	1	0	1	00101101
$j=3$	0	1	0	0	0	1	0	1	01000101
$j=4$	1	1	1	0	1	1	1	1	11101111
$j=5$	0	0	1	1	0	1	1	1	00110111
$j=6$	1	0	1	0	1	1	0	1	10101101
$j=7$	0	1	1	1	0	1	0	0	01110100

Gaussian elimination modulo 2 yields:

$$I^{\intercal}\boldsymbol{\varepsilon} == h \Rightarrow \boldsymbol{\varepsilon} = (0,0,1,0,1,0,0,1)$$

Indeed:

$$h = 10110110 = \mathcal{J}_{\boldsymbol{n}}(r_2) \oplus \mathcal{J}_{\boldsymbol{n}}(r_4) \oplus \mathcal{J}_{\boldsymbol{n}}(r_7) = \begin{cases} 00101101 \\ \quad \oplus \\ 11101111 \\ \quad \oplus \\ 01110100 \end{cases}$$

This yields the signature:

$$s = 47 \times 91 \times 67 + 91116 = 377675$$

For which:

$$\mathcal{J}_{\boldsymbol{n}}(\eta + s) = \mathcal{J}_{\boldsymbol{n}}(\eta + 377675) = 10110110 = h$$

6 Security

Tout est à faire. Eric tu as des idées?

7 Open Questions

Except efficiency, proving or refuting the security of the new signature scheme is an intriguing open question. A second interesting research direction is the generalizing of the construction to higher residues, e.g. following [BLS13] or using Eisenstein integers and cubic residue characters. Despite all our attempts we did not manage to extend the idea to classical RSA moduli (i.e. $n = pq$).

8 A faire

We now show formally that the factoring assumption of the modulus of the form $n = p^2q$ implies the hardness of OWFs described in section 3.

Theorem 2. *If the factoring problem of $n = p^2q$ is hard, then $\mathcal{F}_1$ is a one-way function.*

Theorem 3. *If the factoring problem of $n = p^2q$ is hard, then $\mathcal{F}_2$ is a one-way function.*

References

???08. ??? Security of $n = p^2q$ in ESIGN. `www.cryptrec.go.jp/exreport/cryptrec-ex-1010-2001.pdf`, 2008. [Online; accessed 1-January-2019].

BDHG99. Dan Boneh, Glenn Durfee, and Nick Howgrave-Graham. Factoring $n = p^rq$ for large r. In *Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '99, pages 326–337, Berlin, Heidelberg, 1999. Springer-Verlag.

BLS13. Dan Boneh, Rio LaVigne, and Manuel Sabin. Identity-based encryption with e-th residuosity and its incompressibility. *In: Autumn 2013 TRUST Conference. Washington DC. Poster presentation.*, October 2013.

Cop97. Don Coppersmith. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. *J. Cryptol.*, 10(4):233–260, September 1997.

FOM91. Atsushi Fujioka, Tatsuaki Okamoto, and Shoji Miyaguchi. ESIGN: An efficient digital signature implementation for smart cards. In Donald W. Davies, editor, *Advances in Cryptology — EUROCRYPT '91*, pages 446–457, Berlin, Heidelberg, 1991. Springer Berlin Heidelberg.

Gra03. Louis Granboulan. How to repair ESIGN. In *Proceedings of the 3rd International Conference on Security in Communication Networks*, SCN'02, pages 234–240, Berlin, Heidelberg, 2003. Springer-Verlag.

LLL82. Arjen Klaas Lenstra, Hendrik Willem Lenstra, and László Lovász. Factoring polynomials with rational coefficients. *Math. Ann.*, 261:515–534, 1982.

May04. Alexander May. Secret exponent attacks on RSA-type schemes with moduli $n = p^rq$. In Feng Bao, Robert Deng, and Jianying Zhou, editors, *Public Key Cryptography – PKC 2004*, pages 218–230, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

MF17. Nathan Manohar and Ben Fisch. Factoring $n = p^2q$ (Final project reports, CS359C). Stanford University, 2017.

MQSW08. Alfred Menezes, Minghua Qu, Doug Stinson, and Yongge Wang. Evaluation of security level of cryptography: ESIGN signature scheme. `www.ipa.go.jp/security/enc/CRYPTREC/fy15/doc/1053_esign.pdf`, 2008. [Online; accessed 1-January-2019].

OFM98. Tatsuaki Okamoto, Eiichiro Fujisaki, and Hikaru Morita. TSH-ESIGN: Efficient digital signature scheme using trisection size hash (submission to P1363a), 1998.

OS85. Tatsuaki Okamoto and Akira Shibaishi. A digital signature scheme based on quadratic inequalities. In *Proceeding of Symposium on Security and Privacy, IEEE*, pages 123—132, 1985.

OU98. Tatsuaki Okamoto and Shigenori Uchiyama. A new public-key cryptosystem as secure as factoring. In Kaisa Nyberg, editor, *Advances in Cryptology — EUROCRYPT'98*, pages 308–318, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg.

SPMLS02. Jacques Stern, David Pointcheval, John Malone-Lee, and Nigel P. Smart. Flaws in applying proof methodologies to signature schemes. In Moti Yung, editor, *Advances in Cryptology — CRYPTO 2002*, pages 93–110, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

SS06. Katja Schmidt-Samoa. A new Rabin-type trapdoor permutation equivalent to factoring. *Electron. Notes Theor. Comput. Sci.*, 157(3):79–94, May 2006.

SS07. András Sárközy and Cameron L. Stewart. On pseudorandomness in families of sequencesderived from the Legendre symbol. *Periodica Mathematica Hungarica*, 54(2):163–173, June 2007.

SST05. Katja Schmidt-Samoa and Tsuyoshi Takagi. Paillier's cryptosystem modulo p^2q and its applications to trapdoor commitment schemes. In Ed Dawson and Serge Vaudenay, editors, *Progress in Cryptology – Mycrypt 2005*, pages 296–313, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

STTT03. Hisayoshi Sato, Tsuyoshi Takagi, Satoru Tezuka, and Kazuo Takaragi. Generalized powering functions and their application to digital signatures. In Chi-Sung Laih, editor, *Advances in Cryptology - ASIACRYPT 2003*, pages 434–451, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

Tak98. Tsuyoshi Takagi. Fast RSA-type cryptosystem modulo p^kq. In *Proceedings of the 18th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '98, pages 318–326, London, UK, UK, 1998. Springer-Verlag.