New Online/Offline Signature Schemes Without Random Oracles

Kaoru Kurosawa[1] and Katja Schmidt-Samoa[2]

[1] Department of Computer and Information Sciences, Ibaraki University, Japan
[2] Fachbereich Informatik, Technische Universität Darmstadt, Germany

Abstract. In this paper, we propose new signature schemes provably secure under the strong RSA assumption in the standard model. Our proposals utilize Shamir-Tauman's generic construction for building EF-CMA secure online/offline signature schemes from trapdoor commitments and less secure basic signature schemes. We introduce a new natural intractability assumption for hash functions, which can be interpreted as a generalization of second pre-image collision resistance. Assuming the validity of this assumption, we are able to construct new signature schemes provably secure under the strong RSA assumption without random oracles. In contrast to Cramer-Shoup's signature scheme based on strong RSA in the standard model, no costly generation of prime numbers is required for the signer in our proposed schemes. Moreover, the security of our schemes relies on weaker assumptions placed on the hash function than Gennaro, Halevi and Rabin's solution.
Keywords: Online/offline signatures, trapdoor hash, strong RSA assumption, division intractability

1 Introduction

Digital signatures are intended to replace handwritten signatures in the electronic world. The security goal here is authenticity, *e.g.*, the proof of authorship of messages. Besides obvious applications in electronic commerce, digital signatures are important building blocks for various kinds of cryptographic protocols, and traditional public key infrastructures rely on digital signatures for certifying public keys.

Until 1999, all provably secure solutions for efficient digital signature schemes relied on the random oracle methodology [BR93]. In the random oracle model (ROM), all parties (the legitimate ones as well as the adversary) have black-box access to functions which behave like truly random functions. Under this idealized assumption it became possible to develop cryptosystems that are both efficient and provably secure. In concrete implementations, however, truly random functions are out of reach and the random oracles are replaced by concrete objects like cryptographic hash functions. Thus it is obvious that even a rigorously analyzed security proof in the random oracle model does not guaranty security in the real world. As a real world adversary may exploit some weaknesses of the hash functions used, a proof in the ROM can only exclude *generic* attacks against the scheme. Even worse, recently published results show separations between the random oracle scenario and standard model as there exist cryptosystems provably secure in the ROM that nevertheless are breakable when implemented with any concrete realization [CGH98,CGH04].

Then, in 1999, Cramer and Shoup on the one hand and Gennaro, Halevi and Rabin on the other hand independently came up with practical solutions for digital signature schemes provably secure without random oracles, *i.e.*, in the standard model [CS99,GHR99]. Interestingly, the security of both proposals relies on the same intractability assumption, namely the hardness of the flexible RSA problem, also known as the strong RSA assumption. However, none of these solutions is free from disadvantages. The major drawback of the Cramer-Shoup scheme—referred to as CS scheme in the following—is that the signer is required to generate a prime number for producing a signature. According to heuristics given in [CS99], the costs for prime number generation are one third of the total signing costs on average. The most crucial disadvantage of the Gennaro-Halevi-Rabin scheme—referred to as GHR scheme in the following—is that its security relies on a strong non-standard assumption placed on the hash function used. Gennaro *et al.* prove the existence of suitable hash functions under the strong RSA assumption by constructing a concrete implementation, however, when utilizing this fully proved hash function the entire system becomes less efficient than the CS scheme. Our aim in this paper is to overcome both drawbacks.

On the first glance, the CS scheme and the GHR scheme seem quite different. But in the light of more recent results about generic constructions of provably secure signature schemes, one may observe a common design principle (here, we consider the fully proved GHR scheme): In both cases, first a commitment to the message is constructed, followed by signing the commitment with a "weak" signature scheme. For the first step a *trapdoor* commitment scheme is utilized, which enables the simulator in the security proof to answer signature queries based on previously computed commitments. Although the weak basic signature schemes are different in CS and GHR, both make use of prime numbers to permit the reduction of the flexible RSA problem to the security of the basic signature scheme[1]. In 2001, Shamir and Tauman universalized this approach and proposed a generic construction for online/offline signature schemes [ST01]. As now the mechanisms to enhance the security of "weak" signature schemes by the means of trapdoor commitments are better understood, it seems worthwhile to revisit the CS and GHR schemes.

Our goal is to get rid off the need for prime number generation as well as off the strong assumption placed on the hash function. Therefore, the GHR basic signature scheme seems to be a more promising candidate to start with because

[1] In the GHR scheme, the hash function used in the basic signature scheme has to satisfy a rather strong assumption. Gennaro *et al.* show that a trapdoor commitment scheme combined with a collision resistant hash function producing prime digests only is a possible implementation for the hash function. Thus, formally the task of prime number generation is assigned to the hash function here.

prime number generation is incorporated directly in its CS pendant. An analysis of this scheme reveals that the weak security conditions necessary for a Shamir-Tauman-like construction can be fulfilled if the utilized hash function possesses a property that is similar but intuitively less demanding than its analog in the in the GHR framework. To be more concrete, Gennaro *et al.* introduced the notion of a *division-intractable* family of hash functions $\mathcal{H}$, which briefly states that given $H \in \mathcal{H}$, it is infeasible to find values $X_1, \ldots, X_n, Y$ such that $H(Y)$ divides the product $\prod_{i=1}^{n} H(X_i)$. In contrast, our construction only requires what we call *weak division-intractability*, meaning that given $H \in \mathcal{H}$ *and* $X_1, \ldots, X_n$, it is infeasible to find Y such that $H(Y)$ divides the product $\prod_{i=1}^{n} H(X_i)$. Thus, the values X_i are not longer under the attacker's control. Note that our newly defined property relaxes Gennaro *et al.*'s notion of (strong) division intractability in exactly the same way as second pre-image resistance relaxes collision resistance.

2 Preliminaries

Throughout this paper, we use the following notations:
For any positive integer N we write $\mathbb{Z}_N$ for the ring of residue classes modulo N, and $\mathbb{Z}_N^\times$ for its multiplicative group. $|N|_2$ denotes the bit-length of N, and we write $[N]^k$ for the integer corresponding to the k most significant bits of N.
As usual, a probability $\Pr(k)$ is called *negligible* if $\Pr(k)$ decreases faster than the inverse of any polynomial in k, *i.e.* $\forall c \exists k_c (k > k_c \Rightarrow \Pr(k) < k^{-c})$. In contrast, a probability $\Pr(k)$ is called *overwhelming*, if $1 - \Pr(k)$ is negligible.
We abbreviate *probabilistic polynomial time* by PPT.

2.1 Digital Signature Schemes

A digital signature scheme is denoted by $\Omega = (\mathsf{G}_{sign}, \mathsf{Sign}, \mathsf{Verify})$. G_{sign} is a PPT algorithm which on input a security parameter generates $(\mathrm{sk}, \mathrm{vk})$, where vk and sk are the secret signing and the public verification key, respectively. Sign is a PPT algorithm which produces a signature σ on input a message m and the secret key sk. Verify is a polynomial time algorithm which checks the validity of (m, σ) by using vk, say $\mathsf{Verify}(\mathrm{vk}, m, \sigma) = \mathsf{valid}$ or $\mathsf{invalid}$. It is required that $\mathsf{Verify}(\mathrm{vk}, m, \sigma) = \mathsf{valid}$ holds if and only if σ is a possible outcome of $\mathsf{Sign}(\mathrm{sk}, m)$. For brevity, we also write $\mathsf{Sign}_{\mathrm{sk}}(m)$ instead of $\mathsf{Sign}(\mathrm{sk}, m)$ and $\mathsf{Verify}_{\mathrm{vk}}(m, \sigma)$ instead of $\mathsf{Verify}(\mathrm{vk}, m, \sigma)$.

In the following, we review security notions for digital signature schemes. All the notions below have been introduced by Goldwasser, Micali and Rivest [GMR88].

The standard security notion of signature schemes is existential unforgeability under adaptive chosen message attacks (EF-CMA). Here, the attacker is allowed to query the signing oracle adaptively.

Definition 1 (EF-CMA). *A digital signature scheme* $\Omega = (\mathsf{G}_{sign}, \mathsf{Sign}, \mathsf{Verify})$ *is said to be* existentially unforgeable under adaptive chosen message attacks *if* *for any PPT adversary $\mathcal{A}$ the following probability is negligible in ℓ:*

$$\Pr\left[\begin{array}{l}(\mathrm{sk},\mathrm{vk}) \hookleftarrow \mathsf{G}_{sign}(1^\ell),\\ \mathit{FOR}\ i=1,\ldots,k:\\ \qquad \{m_i \hookleftarrow \mathcal{A}(\mathrm{vk},m_1,\sigma_1,\ldots,m_{i-1},\sigma_{i-1});\ \sigma_i \hookleftarrow \mathsf{Sign}_{\mathrm{sk}}(m_i)\}\,,\\ (m^*,\sigma^*) \hookleftarrow \mathcal{A}(\mathrm{vk},m_1,\sigma_1,\ldots,m_k,\sigma_k):\\ m^* \notin \{m_1,\ldots,m_t\} \wedge \mathsf{Verify}_{\mathrm{vk}}(m^*,\sigma^*) = \mathsf{valid}\end{array}\right].$$

In this paper, we call a signature scheme is *adaptively secure* if it is EF-CMA.

A much weaker security notion is existential unforgeability against random message attacks, *a.k.a.* known message attacks (EF-KMA). Here, the adversary is just given the verification key and a list of randomly generated valid message/signature pairs without any control over the messages.

Definition 2 (EF-KMA). *A digital signature scheme $\Omega = (\mathsf{G}_{sign}, \mathsf{Sign}, \mathsf{Verify})$ is said to be* existentially unforgeable under known message attacks *if for any PPT adversary $\mathcal{A}$ the following probability is negligible in ℓ:*

$$\Pr\left[\begin{array}{l}(\mathrm{sk},\mathrm{vk}) \hookleftarrow \mathsf{G}_{sign}(1^\ell),\\ \mathit{FOR}\ i=1,\ldots,k:\ \{m_i \hookleftarrow \mathcal{M},\ \sigma_i \hookleftarrow \mathsf{Sign}_{\mathrm{sk}}(m_i)\}\,,\\ (m^*,\sigma^*) \hookleftarrow \mathcal{A}(\mathrm{vk},m_1,\sigma_1,\ldots,m_k,\sigma_k):\\ m^* \notin \{m_1,\ldots,m_t\} \wedge \mathsf{Verify}_{\mathrm{vk}}(m^*,\sigma^*) = \mathsf{valid}\end{array}\right].$$

In this paper, we call EF-KMA secure signature schemes *weakly secure*.

2.2 Trapdoor Commitment Schemes

A trapdoor commitment scheme is defined by $\mathcal{TC} = (\mathsf{G}_{TC}, \mathsf{Tcom}, \mathsf{Topen})$, where Topen is Twopen or Tsopen as shown below. G_{TC} is a PPT algorithm which generates $(\mathrm{pk}, \mathrm{tk})$, where pk is the public key and tk is the trapdoor. Associated to $\mathcal{TC}$ are the spaces of messages $\mathcal{M}$, randomness $\mathcal{R}$ and commitments $\mathcal{C}$.

Tcom is the algorithm that computes a commitment to m as $x = \mathsf{Tcom}(\mathrm{pk}, m, r)$, where $r \in \mathcal{R}$ is a random nonce. To open the commitment x, the sender reveals m, r and the receiver recomputes x.

Twopen is the algorithm that weakly opens a commitment in any desired way with the trapdoor tk. For given m, r and a target message m', it outputs $r' = \mathsf{Twopen}(\mathrm{tk}, m, r, m')$ such that $x = \mathsf{Tcom}(\mathrm{pk}, m, r) = \mathsf{Tcom}(\mathrm{pk}, m', r')$.

Hence, the trapdoor holder is able to create a "dummy commitment" and later open this commitment to any message of his choice.

However, for some applications a strictly stronger property turns out to be useful; namely, the owner of the trapdoor key should be able to open a commitment arbitrarily even without knowledge of the pre-image values r, m. We call this mechanism *strong trapdoor opening*[2] and the corresponding schemes *strong*. In such a strong trapdoor commitment scheme there exists an algorithm Tsopen such that for a given commitment x and a target message m it outputs $r = \mathsf{Tsopen}(\mathrm{tk}, m, x)$ with $x = \mathsf{Tcom}(\mathrm{pk}, m, r)$.

[2] In [ST01], this property is referred to as *inversion property*.

The existence of (strong or weak) trapdoor opening algorithms Topen implies that the receiver cannot obtain any information about m given x.

The security of trapdoor commitment schemes requires that without knowledge of the trapdoor key it should be hard to find collisions. Moreover, randomness r obtained by invoking the trapdoor opening algorithm should be indistinguishable from properly generated r. Again, we simplify the notation by writing the keys as indices.

Definition 3. *We say that a trapdoor commitment scheme $\mathcal{TC} = (\mathsf{G}_{TC}, \mathsf{Tcom}, \mathsf{Topen})$ is secure if the following properties hold:*

Collision resistance: *For any PPT $\mathcal{A}$ the following probability is negligible in ℓ:*

$$\Pr\left[\begin{array}{l}(\mathrm{pk}, \mathrm{tk}) \leftarrow \mathsf{G}_{TC}(1^\ell), \mathcal{A}(\mathrm{pk}) = (r, m, r', m'), \\ m \neq m' \wedge \mathsf{Tcom}_{\mathrm{pk}}(r, m) = \mathsf{Tcom}_{\mathrm{pk}}(r', m')\end{array}\right].$$

Uniformity: *The outcome of* Topen *is computationally indistinguishable from uniform in $\mathcal{R}$ provided that*

- *in case of weak altering the input r is uniformly distributed in $\mathcal{R}$, resp.*
- *in case of strong altering the following holds: for any $m \in \mathcal{M}$ the distribution of the input x is computationally indistinguishable from the distribution of $\mathsf{Tcom}_{\mathrm{pk}}(m, r)$, where r is uniformly distributed in $\mathcal{R}$.*

2.3 Hash Functions

A hash function is an efficiently computable procedure that maps strings of arbitrary length to strings of fixed length. The sequence $\mathcal{H} = (H_k)_{k\in\mathbb{N}}$ is called a family of hash functions if each H_k is a collection of hash functions with output length k. Analog to signature and trapdoor commitment schemes, collections of hash functions can also be defined via a key generation algorithm, but for better readability, we utilize less formal notations below. Within the scope of this paper, the most important security properties of hash functions are the standard requirements (second pre-image) collision resistance (dating back to Damgård [Dam87]) and the non-standard ones weak/strong division intractability (the strong version introduced by Gennaro, Halevi and Rabin [GHR99], the weak version introduced and defined below in the present paper).

Definition 4 ((Second pre-image) collision resistance). *A family $\mathcal{H} = (H_k)_{k\in K}$ of hash functions is said to be*

collision resistant *if for any PPT adversary $\mathcal{A}$, the following probability is negligible in k:*

$$\Pr_{H\in H_k}[\mathcal{A}(H) = (X, Y) : X \neq Y \wedge H(X) = H(Y)],$$

second pre-image collision resistant *if for any PPT adversary $\mathcal{A}$, the following probability is negligible in k:*

$$\Pr_{H\in H_k, X}[\mathcal{A}(H, X) = (Y) : X \neq Y \wedge H(X) = H(Y)].$$

It is obvious that collision resistance implies second pre-image collision resistance.

Definition 5 (Weak/strong division intractability). *A family $\mathcal{H} = (H_k)_{k\in\mathbb{N}}$ of hash functions is said to be*

strongly division intractable *if for any PPT adversary $\mathcal{A}$, the following probability is negligible in k:*

$$\Pr_{H\in H_k}\left[\begin{array}{l}\mathcal{A}(H) = (X_1, X_2, \ldots, X_n, Y) : \\ Y \notin \{X_1, X_2, \ldots, X_n\} \wedge H(Y) \text{ divides } \prod_{i=1}^{n} H(X_i)\end{array}\right],$$

weakly division intractable *if for any PPT adversary $\mathcal{A}$, the following probability is negligible in k for any n which is polynomially bounded by k:*

$$\Pr_{H\in H_k, X_1,\ldots,X_n}\left[\begin{array}{l}\mathcal{A}(H, X_1, X_2, \ldots, X_n) = Y : \\ Y \notin \{X_1, X_2, \ldots, X_n\} \wedge H(Y) \text{ divides } \prod_{i=1}^{n} H(X_i)\end{array}\right].$$

Note that our newly defined property of weak division intractability relaxes Gennaro *et al.*'s notion of (strong) division intractability in exactly the same way as second pre-image collision resistance lessens full collision resistance. Moreover, while division intractability obviously implies collision resistance, it is also easy to see that weak division intractability implies second pre-image collision resistance. The opposite directions, however, are not true. We will discuss the relationship between strong and weak division intractability further in Section 5.

2.4 Intractability Assumptions

Our proposed online/offline signature schemes rely on the following standard intractability assumptions:

Claim (Blum Factorization Assumption). Given $N = pq$ for two random primes p, q with $|p|_2 \approx |q|_2$ and $p = q = 3 \bmod 4$, it is hard to factor N.

The integer N from the preceeding assumption is called a *Blum integer*. If N is a Blum integer, then squaring is a permutation on the group $\mathrm{QR}(N)$ of quadratic residues modulo N.

Claim (p^2q Factorization Assumption). Given $N = p^2q$ for two random primes p, q with $|p|_2 \approx |q|_2$, it is hard to factor N.

The following assumption has been first described by Barić and Pfitzmann [BP97].

Claim (Strong RSA Assumption). Given $N = pq$ for two random primes p, q and a randomly chosen $s \in \mathbb{Z}_N^\times$, it is hard to find values $r \in \mathbb{Z}_N^\times$ and $e > 1$ such that $r^e = s \bmod N$.

In the preceeding claim, the tuple (N, s) is called an *instance of the flexible RSA problem.* In the rest of this paper, we sometimes use special moduli such as Blum integers or products of safe primes[3]. In this case, the Strong RSA Assumption has to be understood with respect to these kind of moduli.

We now state a useful lemma, which is proved, for example, in [CL02].

Lemma 1 *Let $N = pq$ be the product of two distinct safe primes $p = 2p'+1, q = 2q' + 1$. Given $s, t \in \mathrm{QR}(N)$ along with $0 < a < b$ such that $s^b = t^a \bmod N$ and $\gcd(a, b) < a$, one can efficiently compute values $r, e > 1$ such that $r^e = s \bmod N$.*

Proof. By using extended Euclidean algorithm, we can efficiently find $u, v \in \mathbb{Z}$ such that $au + bv = \gcd(a, b) =: c$. In particular, we have $(a/c)u + (b/c)v = 1$.

Without loss of generality, we may assume $\gcd(c, p'q') = 1$, because otherwise we can factor N (either directly from the knowledge of p' resp. q', or by applying Miller's algorithm [Mil75] on a multiple of $\varphi(N) = 4p'q'$). Therefore, from $s^b = t^a \bmod N$, we conclude $s^{b/c} = t^{a/c} \bmod N$, leading to

$$s = s^{(a/c)u+(b/c)v} = s^{(a/c)u}t^{(a/c)v} = (s^u t^v)^{(a/c)} \bmod N.$$

Hence, we obtain $e = a/c$ and $r = s^u t^v \bmod N$. □

Note that as one quarter of the elements of $\mathbb{Z}_N^\times$ are quadratic residues, we have that if the Strong RSA Assumption is true at all, then it is also true for instances (N, s) where s is randomly chosen from $\mathrm{QR}(N)$. Thus efficiently finding t, a, b given N, s as in Lemma 1 above violates the Strong RSA Assumption.

2.5 Online/Offline Signature Schemes

The notion of online/offline signatures was introduced by Even *et al.* [EGM96]. In such schemes, the online phase of the signing algorithm is made very fast due to the precomputation performed in the offline phase before the message actually to be signed is known.

In 2001, Shamir and Tauman improved this generic construction [ST01]. Informally, their new approach can be described as using the well-known hash-then-sign paradigm, where the ordinary hash function is replaced by a trapdoor commitment scheme: Let $\Omega = (\mathsf{G}_{sign}, \mathsf{Sign}, \mathsf{Verify})$ and $\mathcal{TC} = (\mathsf{G}_{TC}, \mathsf{Tcom}, \mathsf{Topen})$ be a weakly secure signature scheme and a trapdoor commitment scheme, respectively. The key generation algorithm of the entire online/offline signature scheme runs both individual key generation algorithms $\mathsf{G}_{sign}, \mathsf{G}_{TC}$, and the signer is given the secret signing key sk as well as the secret trapdoor key tk. The public key is (vk, pk), where vk is the verification key of Ω and pk is the public key of $\mathcal{TC}$.

Offline phase: Choose a dummy message $\tilde{m}$ and a random number $\tilde{r}$. Compute hash $= \mathsf{Tcom}_{\mathrm{pk}}(\tilde{m}, \tilde{r})$, $\sigma = \mathsf{Sign}_{\mathrm{sk}}(\mathrm{hash})$ and store $(\tilde{m}, \tilde{r}, \sigma)$.

[3] A prime p is called a *safe prime* if $(p - 1)/2$ is also prime.

Online phase: Given a message m, first retrieve $(\tilde{m}, \tilde{r}, \sigma)$ from memory. Then, by using tk, find r such that $\mathsf{Tcom}_{\mathrm{pk}}(m, r) = \mathsf{Tcom}_{\mathrm{pk}}(\tilde{m}, \tilde{r})$ holds. Output (σ, r) as the signature of m.

Verification is straightforward, as by construction σ is a valid hash-then-sign signature of m.

Fortunately, this generic construction also enhances the security of the basic signature scheme: If Ω is existentially unforgeable against generic message attacks (EF-GMA), then the online-offline scheme as described above is adaptively secure (EF-CMA). Moreover, if $\mathcal{TC}$ also allows *strongly* trapdoor opening, then Ω is only required to be existentially unforgeable under known message attacks (EF-KMA).

Therefore, Shamir and Tauman's construction might also be useful in environments where the distinction between online and offline costs is not an issue. In this case, the composed signature algorithm simply consists of committing to the message and signing the commitment, and there is no need for the signer to know the trapdoor key. The ability of arbitrarily opening commitments is only required in the security proof to enable the simulator to respond to the signature queries. In the following, we call this construction the *commit-then-sign approach*. As mentioned in the Introduction, the CS scheme also follows this design principle[4].

Remark 1. The technique of commiting to a message with a trapdoor commitment scheme and signing the commitment has also been used by Krawczyk and Rabin for introducing chameleon signatures [KR00]. In contrast to the approach above, in a chameleon signature scheme the *recipient* is the trapdoor holder. Whilst in case of Shamir/Tauman, the intended goal is efficient online signing and a security enhancement of the basic signature scheme, the aim of chameleon signatures is to distract the receiver of a signature from revealing the signed message to any third party.

3 The Primitives

In this section, we present the building blocks for our proposed full signature schemes. As noted above, the basic primitives are a (strong) trapdoor commitment scheme and a weakly secure signature scheme.

3.1 A Trapdoor Commitment Scheme with Strongly Trapdoor Opening Based on Factoring

We propose a factorization-based trapdoor commitment scheme $\mathcal{TC}_{2^k} = (\mathsf{G}_{TC}, \mathsf{Tcom}, \mathsf{Topen})$ resting on the 2^k identification scheme of Shoup [Sho99] as follows:

[4] In fact, Cramer and Shoup also proposed an online/offline version of their scheme by providing the signer with the trapdoor key. Thus, Shamir and Tauman's idea is not new.

G_{TC}**:** Let ℓ be a security parameter. Choose two ℓ-bit prime numbers p and q such that $p = q = 3 \bmod 4$. Let $N = pq$. Pick $v \in QR(N)$ randomly and define a parameter k such that 2^k grows faster than any polynomial in ℓ. The public key consists of (N, v, k) and the trapdoor key is (p, q).

Tcom**:** To commit to a message $m \in \{0, \ldots, 2^{k-1} - 1\}$, the commiter chooses a random value $r \in \mathbb{Z}_N^\times$ and computes $\mathsf{Tcom}_{\mathrm{pk}}(r, m) = r^{2^k} v^m \bmod N$.

Topen**:** Given a target message m and a commitment x, the strong trapdoor opening algorithm computes $r \in \mathbb{Z}_N^\times$ such that $x = r^{2^k} v^m \mod N$. Weak trapdoor opening is realized by $\mathsf{Twopen}_{\mathrm{tk}}(m, r, m') = r' = r v^{(m-m')2^{-k}} \bmod N$.

We have the following theorem:

Theorem 1. *Under the Blum Factorization Assumption the above construction $\mathcal{TC}_{2^k}$ is a strong trapdoor commitment scheme secure in the sense of Definition 3.*

Proof. The correctness of the trapdoor opening algorithms is obvious.

To prove the collision resistance, we assume that $\mathcal{A}$ is a PPT collision finder. We then construct a PPT algorithm I which can factor Blum integers N as follows: On input N, I chooses a such that

$$\left(\frac{a}{N}\right) = -1$$

randomly, where $\left(\frac{\cdot}{\cdot}\right)$ denotes the Jacobi symbol. I computes $v = a^2 \bmod N$ and runs $\mathcal{A}$ on input (N, v). A eventually outputs $(m, r), (m', r')$ such that $m \neq m', \mathsf{Tcom}(r, m) = \mathsf{Tcom}(r', m')$. It holds that

$$r^{2^k} v^m = r'^{2^k} v^{m'} \bmod N.$$

Therefore, we obtain that

$$(r/r')^{2^k} = v^{m'-m} \bmod N.$$

Wlog, assume that $m' > m$ and let $m' - m = u2^t$, where u is odd. Then $t < k-1$. Let $z = r/r' \bmod N$. Now

$$z^{2^k} = v^{u2^t} = (a^2)^{u2^t} \bmod N.$$

Since $p = q = 3 \bmod 4$, we have

$$(z^{2^{k-t-1}})^2 = (a^u)^2 \bmod N.$$

From $k - t - 1 > 0$, we have

$$\left(\frac{z^{2^{k-t-1}}}{N}\right) = 1.$$

On the other hand,

$$\left(\frac{a^u}{N}\right) = -1$$

because u is odd. Therefore, we can factor N with probability 1 by computing $\gcd(a^u - z^{2^{k-t-1}}, N)$.

Finally, we note that for each message $m \in \{0, \ldots, 2^{k-1} - 1\}$ and for each commitment $x \in \mathrm{QR}(N)$ there are exactly four $r \in \mathbb{Z}_N^\times$ with $x = \mathsf{Tcom}(r, m)$. Consequently, uniformity holds for both trapdoor opening algorithms. $\square$

Remark 2. If weak altering is sufficient, we define $v^{2^{-k}} \bmod N$ as the trapdoor key.

As we will see, combined with a weakly secure signature scheme, $\mathcal{TC}_{2^k}$ yields an adaptively secure commit-then-sign scheme as described in Section 2.5. However, as the opening algorithms require a modular exponentiation, it is not reasonable to use $\mathcal{TC}_{2^k}$ as a building block for a full online/offline signature scheme.

For the construction of schemes with real online/offline properties, trapdoor commitments with extremely fast weak trapdoor opening are required. A variant of the following scheme $\mathcal{TC}_{p^2q} = (\mathsf{G}_{TC}, \mathsf{Tcom}, \mathsf{Topen})$ has recently been proposed by Schmidt-Samoa and Takagi [SST05]:

G_{TC}: Let ℓ be a security parameter. Randomly choose two ℓ-bit primes p, q with $p \nmid q-1, q \nmid p-1$ and compute $N = p^2q$. Define a parameter k minimal with respect to $2^k > pq\sqrt{p}$, and a parameter l maximal with respect to $lpq < 2^k$. The public key is $\mathrm{pk} = (N, k)$, and the trapdoor key is $\mathrm{tk} = (p, q, l)$.

Tcom: To commit to a message $m \in \{0, \ldots, [N]^{|N|_2 - k} - 1\}$, a value $r \in \{0, \ldots, 2^k - 1\}$ is chosen uniformly at random and $\mathsf{Tcom}(r, m) = (2^k m + r)^N \bmod N$ is computed, where $[N]^{|N|_2 - k}$ stands for the integer corresponding to the $|N|_2 - k$ most significant bits of N.

Topen: Given a target message m and a commitment x, the strong trapdoor opening algorithm first computes $aux = x^{1/N} - 2^k m \bmod pq$. Then, $0 \le s < l$ is chosen uniformly at random, and the output r is computed as $r = aux + spq$.
Weak trapdoor opening on the input m, r, m' is realized by first computing $aux = 2^k(m - m') + r \bmod pq$, and then proceeding as before.

Theorem 2 ([SST05]). $\mathcal{TC}_{p^2q} = (\mathsf{G}_{TC}, \mathsf{Tcom}, \mathsf{Topen})$ *is a secure trapdoor commitment scheme in the sense of Definition 3.*

Remark 3. In the original scheme from [SST05], the randomness is chosen from $\mathbb{Z}_{pq}$. In this case, however, a polynomial number of trapdoor openings reveals a logarithmic number of the most significant bits of the secret pq. Although this is not a thread in the light of current factoring achievements (lattice methods like [Cop97] require the knowledge of the $\mathcal{O}((pq)^{1/3})$ most significants bit of pq to factor p^2q), we slightly modified the scheme as described above. Now, the randomness is sampled from the set $\{0, \ldots, 2^k - 1\}$, and the r constructed by the opening algorithms Topen is uniformly distributed over the set $\{0, \ldots, lpq - 1\}$. These distributions are statistically close (a simple computation shows that the distance is upperbounded by $2/\sqrt{p}$). This modification also ensures that the simulator in the commit-then-sign security proof is able to compute commitments properly.

Note that weak trapdoor opening only requires a modular addition, a short integer multiplication, and a bit-shift, and therefore can be computed extremely fast.

3.2 A Weakly Secure Signature Scheme Based on Strong-RSA

In this section, we analyze a simple RSA-type hash-then-sign signature scheme. The proposed scheme is essentially the same as Gennaro, Halevi and Rabin introduced in [GHR99]. In that paper, Gennaro *et al.* proved that when instantiated with a so-called *suitable* hash function, their scheme is adaptively secure (EF-CMA) under the Strong RSA Assumption. The most crucial demands on a suitable hash function are (strong) division intractability, which can be achieved by forcing the output to be a prime, and the property that collision finding does not help solving the flexible RSA problem, *i.e.*, the two associated intractability assumptions should be unrelated in a sense. The latter requirement is dealt with by implementing the hash function as a trapdoor commitment scheme. In the following, we prove that if we relax the hash requirement to *weak* division intractability, then the signature scheme is still weakly secure.
Let us now describe the basic signature scheme $\Omega_{\mathsf{S-RSA}} = (\mathsf{G}_{sign}, \mathsf{Sign}, \mathsf{Verify})$.

G_{sign}**:** On input a security parameter ℓ, choose two safe ℓ-bit primes p, q. Set $N = pq$ and randomly select $y \in \mathrm{QR}(N)$. Finally, pick a weakly division intractable hash function H from a family of hash functions. We assume that H always outputs odd integers[5]. The public key consists of N, y and H; the secret key is p, q.

Sign**:** To sign a message $m \in \{0,1\}^*$, first compute the hash $e = H(m)$. Then, with knowledge of p and q, compute an e-th root of y modulo N:

$$\sigma = y^{\frac{1}{e}} \bmod N.$$

Then, σ is the signature of m.

Verify**:** Given (m, σ), output valid if $\sigma^{H(m)} = y \bmod N$ holds and $\mathsf{invalid}$, otherwise.

Note that the signing algorithm can compute an appropriate root modulo N with overwhelming probability because N is a product of safe primes. Namely, four is the only small factor of $\varphi(N)$ and thus, any odd element not co-prime with $\varphi(N)$ reveals the factorization of N.

[5] This can be easily achieved by setting the least significant output bit to one.

Theorem 3. *Provided the Strong RSA Assumption is valid, the basic signature scheme $\Omega_{\mathsf{S-RSA}}$ above is existentially unforgeable under known message attacks (EF-KMA).*

Proof. Let $\mathcal{F}$ be a EF-KMA adversary against $\Omega_{\mathsf{S-RSA}}$. We construct an attacker $\mathcal{A}$ against the Strong RSA Assumption, which uses $\mathcal{F}$ as a subroutine. $\mathcal{A}$ is given a quadratic instance (N, s) of the flexible RSA problem for safe moduli, *i.e.*, N is a product of two safe primes and s is a quadratic residue modulo N. $\mathcal{A}$ picks dummy messages $m_1, \ldots, m_k$ at random and defines

$$y = s^{\prod_{i=1}^{k} H(m_i)} \bmod N.$$

Moreover, $\mathcal{A}$ computes

$$\sigma_j = s^{\prod_{i=1, i \neq j}^{k} H(m_i)} \bmod N$$

for $j = 1, \ldots, k$. Observe that, by construction, σ_j is a valid signature on m_j. $\mathcal{A}$ gives the forger $\mathcal{F}$ the public key N, y as well as the signature/message pairs $(m_1, \sigma_1), \ldots, (m_k, \sigma_k)$. Eventually, $\mathcal{F}$ outputs a forgery (m, σ). Validity of this forgery implies

$$\sigma^{H(m)} = y = s^{\prod_{i=1}^{k} H(m_i)} \bmod N.$$

As H is weakly division intractable and $m \notin \{m_1, \ldots, m_k\}$, we must have $\gcd(\prod_{i=1}^{k} H(m_i), H(m)) < H(m)$. Thus, by applying Lemma 1, $\mathcal{A}$ can efficiently find values $r, e > 1$ with $r^e = s \bmod N$. Consequently, $\mathcal{A}$ could break the Strong RSA Assumption if the advantage of $\mathcal{F}$ were non-negligible. □

4 New Adaptively Secure Signatures Based on Strong-RSA

In this section, we eventually combine the primitives described in the section above using Shamir-Tauman's approach. As mentioned before, we utilize the trapdoor commitment $\mathcal{TC}_{2^k}$ to enhance the weak security of the basic signature scheme $\Omega_{\mathsf{S-RSA}}$ to full adaptive security, whereas the usage of $\mathcal{TC}_{p^2 q}$ additionally provides online/offline functionality. The reason why we have introduced $\mathcal{TC}_{2^k}$ is that its underlying intractability assumption (Blum Factorization) is implied by the Strong RSA Assumption, and thus we can base the entire construction on the latter only.

In the following, we assume that H is a hash function that always outputs odd integers. Our first proposal is as follows:

G_{sign}: Choose two safe primes p_1, q_1 as well as two primes p_2, q_2 with $p_2 = q_2 = 3 \bmod 4$. Set $N_1 = p_1 q_1,\ N_2 = p_2 q_2$ and randomly select $y \in \mathrm{QR}(N_1),\ v \in \mathrm{QR}(N_2)$. Define a parameter k such that 2^k grows faster than any polynomial in the security parameter. The public key consists of N_1, N_2, y, v, k; the secret key is p_1, q_1.

Sign: To sign a message $m \in \{0, \ldots, 2^{k-1} - 1\}$, first commit to m by choosing a random value $r \in \mathbb{Z}_{N_2}^{\times}$ and computing $x = r^{2^k} v^m \bmod N_2$. Then build the hash $e = H(x)$. Finally, with knowledge of p_1 and q_1, construct an e-th root of y modulo N_1:

$$\sigma = y^{\frac{1}{e}} \bmod N_1.$$

Output (σ, r) as the signature of m.

Verify: Given (m, σ, r), first compute $x = r^{2^k} v^m \bmod N_2$. Output valid if $\sigma^{H(x)} = y \bmod N_1$ holds and invalid, otherwise.

Theorem 4. *If H is weakly division intractable and the Strong RSA Assumption is valid, then the signature scheme above is existentially unforgeable under adaptive chosen message attacks (EF-CMA).*

Proof. From Theorem 1 we have that the commitment scheme utilized in the construction above is a secure trapdoor commitment scheme which allows strongly trapdoor opening. Theorem 1 states that the basic signature scheme used to sign the commitments is weakly secure under the Strong RSA Assumption. The generation of different moduli ensures that the underlying problems are unrelated, *i.e.*, even with knowledge of p_2, q_2, which enables to open the commitments in any desired way, it is still assumed to be infeasible to solve the flexible RSA problem with respect to N_1. Thus, from the results of Shamir and Tauman [ST01], the assertion follows.

A direct proof without using the result from Shamir and Tauman is given in the full version of this paper [KSS06]. □

Now we replace the commitment scheme to achieve online/offline functionality.

G_{sign}: Choose two safe primes p_1, q_1 as well as two primes p_2, q_2 with $p_2 \nmid q_2 - 1, q_2 \nmid p_2 - 1$. Set $N_1 = p_1 q_1,\ N_2 = p_2^2 q_2$ and randomly select $y \in \mathrm{QR}(N_1)$. Define a parameter k minimal with respect to $2^k > pq\sqrt{p}$, and a parameter l maximal with respect to $lpq < 2^k$. The public key consists of N_1, N_2, y, k; the secret key is p_1, q_1, p_2, q_2, l.

Sign:
1. Offline phase: Pick a dummy message $\tilde{m} \in \{0, \ldots, [N_2]^{|N_2|_2 - k} - 1\}$, and commit to $\tilde{m}$ by choosing a random value $\tilde{r} \in \mathbb{Z}_{p_2 q_2}$ and computing $x = (2^k \tilde{m} + \tilde{r})^{N_2} \bmod N_2$. Then build the hash $e = H(x)$. Finally, with knowledge of p_1 and q_1, construct an e-th root of y modulo N_1:

$$\sigma = y^{\frac{1}{e}} \bmod N_1.$$

   Store $\sigma, \tilde{m}, \tilde{r}$.
2. Online phase: To finish the signature generation when the message m to be signed is known, first retrieve $\sigma, \tilde{m}, \tilde{r}$ from memory. Then compute $aux = 2^k(\tilde{m} - m) + \tilde{r} \bmod p_2 q_2$. Finally, $0 \leq s < l$ is chosen uniformly at random, and r is computed as $r = aux + spq$. Output (σ, r) as the signature of m.

Verify: Given (m, σ, r), first compute $x = (2^k m + r)^{N_2} \bmod N_2$. Output valid if $\sigma^{H(x)} = y \bmod N_1$ holds and invalid, otherwise.

The following theorem can be proved exactly as the theorem above because the commitment scheme utilized in the construction above is a secure trapdoor commitment scheme [SST05] which allows strongly trapdoor opening.

Theorem 5. *Assume the Strong RSA Assumption and the p^2q Factorization Assumption are valid. If H is weakly division intractable, then the signature scheme above is existentially unforgeable under adaptive chosen message attacks (EF-CMA).*

Remark 4. In the schemes above, we restricted the message spaces according to the requirements of the trapdoor commitment schemes. Extensions to arbitrary message spaces are possible when utilizing families of collision-resistant hash functions.

5 Comparison

In this section, we compare our proposals with the CS and GHR schemes[6]. Under the assumption that weak, resp. strong division intractable hash functions exist, neither GHR nor our proposals require the signer to perform costly prime number generations as in (modified) CS.

We next discuss why we regard weak division intractability as more reasonable than its strong pendant. First note that a random oracle is weakly as well as strongly division intractable. Assuming a hash function behaving like a random oracle, Coron and Naccache analyzed the complexity of an attack against strong division intractability [CN00]. The outline of their proposed attack is to find a smooth hash value first, and then, for each of its (small) prime divisors p, to search for another hash value divisible by p. Based on theoretical results on the density of smooth numbers, Coron and Naccache show that the running time of this attack is sub-exponential in the digest length. Thus, they recommend a digest length of at least 1024 bits, which is twice as large as suggested by Gennaro *et al.* in [GHR99]. We want to point out that this attack does not work against weak division intractability where the adversary has no control over the hash values that should be divided.

We conducted some experiments to investigate weak division intractability (of random oracles) heuristically. For each pair (n, k), we performed 200 experiments: n k-bit numbers were chosen uniformly at random, and we counted the number of random k-bit numbers x to pick, until x divides the product of the others. The measured data suggests that the expected value of numbers x to pick is lower bounded by $n^{-1.5}2^{k/3}$ for n chosen polynomial in k. Table 1 shows the results of some of these experiments. For each pair (n, k), the table contains three entries: the first one is the evaluation of $n^{-1.5}2^{k/3}$, the second one is the mean of all performed experiments, and the third one is the second-smallest number

[6] In [CL02], Camenisch and Lysyanskaya also propose a signature scheme based on strong RSA in the standard model. As their scheme is less efficient as CS–it has other qualities instead–we exclude it from our considerations.

appearing in the 200 experiments (an entry “-” indicates that no experiments have been performed at all, whilst the index “+” denotes that the respective data is based on less than 200 experiments).

$n \diagdown k$	20			40			60			80			$100^{(+)}$		
k	1	56	1	41	21555	606	2256	18490671	93702	–			–		
$k^{1.5}$	<1	9	1	3	415	4	105	33631	229	5566	3141452	14240	–		
k^2	<1	3.5	1	<1	44	1	5	941	11	208	28883	431	10823	493263	13613
$k^{2.5}$	<1	2.2	1	<1	14	1	<1	135	2	8	1383	7	342	13749	1289
k^3	<1	1.5	1	<1	5	1	–			–			–		

Table 1. Experiments on weak division intractability

If the assumed bound $n^{-1.5}2^{k/3}$ is correct, than the probability that for n fixed uniformly distributed k-bit integers a randomly chosen k-bit integer divides the product of the others is upperbounded by $n^{1.5}2^{-k/3}$. That is, asymptotically this probability is independent from n (provided that n is polynomial in k). For a more practical-oriented interpretation, recall that in our schemes the number n describes the number of the adversary's signature queries. It is common to upperbound this number by 2^{30}. Therefore, we assume that moderate digest lengths, say 256-512 bits, are reasonable for our proposals. We leave the theoretical investigation of weak division-intractability as further work.

Gennaro *et al.* showed how to build strongly division intractable hash functions from collision resistant hash functions essentially by forcing the output to be a prime. Although this approach is not of practical relevance (because in this case CS is clearly more efficient), note that to achieve weak division intractability in that way only second pre-image collision resistant hash functions instead of collision resistant ones were required.

We finally compare the computational efficiency. For a fair comparison, in case of GHR we refer to the variant where suitability of the hash function is achieved by combining a division intractable hash function with a trapdoor commitment scheme. Referring to the computational costs for the modular exponentiations, the differences between all schemes are within a small margin. There is one full modular exponentiation needed for signing in the basic signature scheme, but this task can be significantly sped up by using standard techniques like Chinese remaindering and efficient exponentiation based on precomputation. For the latter, comb methods like [LL94] can be applied because the base of the exponentiation is fixed (this is immediate in our proposals and in the GHR scheme, whilst it requires appropriately chosen verification keys and additional secret keys in the CS scheme and in its modification proposed by Fischlin [Fis03]). In addition, all schemes require a short exponentiation for commiting to the message[7]. In Fischlin's modification of CS, this short exponentiation is eliminated at the expense of a slightly more costly full exponentiation and an increased length of the verification key. Verification requires two short exponentiation in CS, one short plus one short double exponentiation in Fischlin's modified CS and in our first proposal, and one short plus one full exponentiation in our second proposal. The verification costs for GHR depend on the trapdoor commitment used.

6 Conclusion

In this paper we utilized a Shamir-Tauman-like framework to construct new signature schemes based on the strong RSA assumption. Our proposals are existentially unforgeable under adaptive chosen message attacks in the standard model. As in the well-known Gennaro-Halevi-Rabin scheme, we utilized a hash function with a special property, namely division intractability. However, we significantly relaxed this requirement such that for our proposal *weak* division intractability is sufficient. The relation between weak and strong division intractability can be compared to the relation between second pre-image resistance and collision resistance. This newly defined property may be of independent interest. In contrast to the Cramer-Shoup signature scheme based on strong RSA, in our schemes there is no need for the signer to generate a fresh prime number for each message to be signed.

Acknowledgments

The authors wish to thank anonymous referees for useful comments. The second author was supported by the Japanese Society for Promotion of Science (JSPS) for doing research in Japan.

References

[BP97] N. Barić and B. Pfitzmann. Collision-free accumulators and fail-stop signature schemes without trees. In Walter Fumy, editor, *EUROCRYPT*, volume 1233 of *Lecture Notes in Computer Science*, pages 366 – 377, Berlin, 1997. Springer-Verlag.

[BR93] M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *Proc. of the 1st ACM Conference on Computer and Communications Security (CCS)*, pages 62–73. ACM Press, 1993.

[CGH98] R. Canetti, O. Goldreich, and S. Halevi. The random oracle methodology, revisited (preliminary version). In *Proc. of the 30th Annual ACM Symposium on Theory of Computing (STOC '98)*, pages 209–218, New York, NY, USA, 1998. ACM Press.

[7] This exponentiation is full in our second proposal, but there only the offline costs are affected.

[CGH04] R. Canetti, O. Goldreich, and S. Halevi. On the random-oracle methodology as applied to length-restricted signature schemes. In Moni Naor, editor, *TCC*, volume 2951 of *Lecture Notes in Computer Science*, pages 40–57. Springer, 2004.

[CL02] J. Camenisch and A. Lysyanskaya. A signature scheme with efficient protocols. In Stelvio Cimato, Clemente Galdi, and Giuseppe Persiano, editors, *SCN*, volume 2576 of *Lecture Notes in Computer Science*, pages 268–289. Springer, 2002.

[CN00] J.-S. Coron and D. Naccache. Security analysis of the Gennaro-Halevi-Rabin signature scheme. In Bart Preneel, editor, *EUROCRYPT*, volume 1807 of *Lecture Notes in Computer Science*, pages 91–101. Springer, 2000.

[Cop97] D. Coppersmith. Small solutions to polynomial equations, and low exponent rsa vulnerabilities. *J. Cryptology*, 10(4):233–260, 1997.

[CS99] R. Cramer and V. Shoup. Signature schemes based on the strong RSA assumption. In *ACM Conference on Computer and Communications Security*, pages 46–51, 1999.

[Dam87] I. Damgård. Collision free hash functions and public key signature schemes. In David Chaum and Wyn L. Price, editors, *EUROCRYPT*, volume 304 of *Lecture Notes in Computer Science*, pages 203–216. Springer, 1987.

[EGM96] S. Even, O. Goldreich, and S. Micali. On-line/off-line digital signatures. *Journal of Cryptology*, 9(1):35–67, 1996.

[Fis03] M. Fischlin. The Cramer-Shoup strong-RSA signature scheme revisited. In Yvo Desmedt, editor, *Public Key Cryptography*, volume 2567 of *Lecture Notes in Computer Science*, pages 116–129. Springer, 2003.

[GHR99] R. Gennaro, S. Halevi, and T. Rabin. Secure hash-and-sign signatures without the random oracle. In Jacques Stern, editor, *EUROCRYPT*, volume 1592 of *Lecture Notes in Computer Science*, pages 123–139. Springer, 1999.

[GMR88] S. Goldwasser, S. Micali, and R. L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM J. Comput.*, 17(2):281–308, 1988.

[KR00] H. Krawczyk and T. Rabin. Chameleon signatures. In *Proc. of the Symposium on Network and Distributed Systems Security (NDSS)*. The Internet Society, 2000.

[KSS06] K. Kurosawa and K. Schmidt-Samoa. New online/offline signature schemes without random oracles. Cryptology ePrint Archive, 2006. http://eprint.iacr.org/.

[LL94] C. H. Lim and P. J. Lee. More flexible exponentiation with precomputation. In Yvo Desmedt, editor, *CRYPTO*, volume 839 of *Lecture Notes in Computer Science*, pages 95–107. Springer, 1994.

[Mil75] G. L. Miller. Riemann's hypothesis and tests for primality. In *Proc. of the 7th annual ACM symposium on Theory of computing (STOC '75)*, pages 234–239, New York, NY, USA, 1975. ACM Press.

[Sho99] Victor Shoup. On the security of a practical identification scheme. *J. Cryptology*, 12(4):247–260, 1999.

[SST05] K. Schmidt-Samoa and T. Takagi. Paillier's cryptosystem modulo p^2q and its applications to trapdoor commitment schemes. In Ed Dawson and Serge Vaudenay, editors, *Mycrypt*, volume 3715 of *Lecture Notes in Computer Science*, pages 296–313. Springer, 2005.

[ST01] A. Shamir and Y. Tauman. Improved online/offline signature schemes. In Joe Kilian, editor, *CRYPTO*, volume 2139 of *Lecture Notes in Computer Science*, pages 355–367. Springer, 2001.