Space Efficient Signature Schemes from the RSA Assumption

Shota Yamada*[1], Goichiro Hanaoka[2] and Noboru Kunihiro[1]

[1] The University of Tokyo. {yamada@it., kunihiro@} k.u-tokyo.ac.jp
[2] National Institute of Advanced Industrial Science and Technology (AIST).
hanaoka-goichiro@aist.go.jp

Abstract. Signature schemes from the RSA assumption are very important because of their highly reliable security. Despite their importance, only a few digital signature schemes from the RSA assumption are currently known. Thus, improvement of efficiency in this area seems to be very important. In this paper, we propose various signature schemes from the RSA assumption. First, we propose a scheme that simultaneously provides the shortest signatures and public key length among the known schemes. Compared with the known best schemes, the signature size is the same as that of the scheme proposed recently by Hofheinz, Jager, and Kiltz, whereas the public key size is about the half that of the Hohenberger-Waters scheme. The drawback of the scheme is its heavy signing and verification algorithms. Second, we also propose a scheme whose public key is longer than our first scheme, but the signing and verification cost is more efficient. The scheme can be seen as a generalization of our first scheme and the Hofheinz-Jager-Kiltz scheme. Finally, we propose a scheme whose signing and verification algorithms are more efficient than our first and second schemes, whereas the signature size is longer. All these schemes are constructed based on a new observation about the relation between m-time signature schemes and short signature schemes.

Key words: Short signature, m-time signature, RSA assumption

1 Introduction

1.1 Background

Construction of a digital signature scheme with existential unforgeability under chosen message attack (EUF-CMA) [9] in the standard model is a main research topic in cryptography. In particular, the construction of a short signature from a mild assumption has been extensively studied. Earlier studies proposed various efficient signature schemes in the standard model from various assumptions, such as the strong q-DH assumption [2, 11], the q-DH assumption [10, 20], the strong RSA assumption [8, 3, 7, 11], and the CDH assumption [19]. Many of these

* The first author is supported by a JSPS Research Fellowship for Young Scientists.

schemes rely on the q-type assumption or the strong type assumption, except for Waters' scheme [19]. Even though these assumptions seem reasonable, it is desired to construct a signature scheme from a better studied, weaker assumption (such as the RSA assumption), to obtain high confidentiality in security. A digital signature scheme from the RSA assumption whose signatures are short enough, is not known even in the stateful setting until the recent work by Hohenberger and Waters [12]. Subsequently, they proposed a signature scheme from the RSA assumption in the stateless setting using a new technique [13]. Very recently, Hofheinz, Jager, and Kiltz showed that even shorter signature schemes can be obtained using a programmable hash function [10]. Despite of their importance, no (stateless) signature schemes from the RSA assumption are known, except for the schemes proposed in the above two papers. Improvement in efficiency for RSA based signature schemes seems very important as a step to obtaining a truly efficient, reliable signature scheme in the future. In this paper, we propose various novel signature schemes from the RSA assumption. For example, we propose a scheme that achieves the shortest signature size and public key size simultaneously.

1.2 Our Approach

As an approach to constructing short signature schemes, we focus on the fact that a one-time signature scheme and a weakly secure signature scheme yield a fully-fledged signature scheme. This is a variant of the generic construction proposed in [18]. As we will discuss in a later section, the idea can be (informally) generalized to the combination of an m-time signature scheme and a weakly secure scheme. This is the first time that this idea has been explicitly discussed. Even though the idea is not formal, the idea is conceptually of interest since it often leads to constructions of short signature schemes. For example, recent generic constructions of short signature schemes from the programmable hash function [10] and its variant [20] can be seen as the realizations of the idea. Based on this idea, we construct various novel signature schemes from the RSA assumption. Conceptually, we take two steps to construct a scheme. First, we construct an m-time signature scheme from the RSA assumption. We then combine it with a weakly secure signature scheme from the RSA assumption proposed by Hohenberger-Waters [13]. According to this strategy, we obtain various new schemes from the RSA assumption.

More concretely, we obtain three signature schemes based on the approach described above. In section 4, we propose a signature scheme that provides the shortest signature size and public key size simultaneously. Compared to currently known best schemes, the signature size of our scheme is 1074 bits, which is the same as that of $\mathsf{Sig}_{\mathsf{RSA}}[\mathsf{H}_{\mathsf{cfs}}]$ in [10], whereas the public key size is about 2000 bits, which is about half the size of the scheme by Hohenberger and Waters [13]. The drawback of the scheme is its heavy signing and verification algorithms. To compensate it, in section 5, we also propose another scheme whose signature size is the same as our first scheme, and the public key is longer than for that scheme, but the signing and verification cost is more efficient. The scheme is equipped

with parameters u_1, u_2 and we can adjust the trade-off between public key length and signing and verification cost. The scheme can be seen as a generalization of our first scheme and $\mathsf{Sig}_{\mathsf{RSA}}[\mathsf{H}_{\mathsf{cfs}}]$ in [10]. In fact, we can obtain a slight variant of these schemes as a special case of the scheme. Finally, in section 6, we propose a scheme whose signing and verification cost is more efficient than the our first and second schemes whereas the public key and signature sizes are larger than for our first scheme. The structure of the scheme can be seen as a hybrid of our first scheme and $\mathsf{Sig}_{\mathsf{RSA}}[\mathsf{H}_{\mathsf{rand}}]$ in [10].

Finally, we note that we have also constructed a stateful version of our second and third schemes. The scheme is more efficient than corresponding stateless version of the scheme except for the signature size, which is slightly larger.

2 Preliminaries

For $\lambda \in \mathbb{N}$, 1^λ denotes the string of λ ones, with λ expressing the security parameter throughout this paper. $[d]$ denotes the set $\{1, 2, \ldots, d\}$. Moreover, $|x|$ and $|S|$ denote, respectively, the length of bitstring x, and the size of set S. If S is a set, $s \xleftarrow{\$} S$ denotes the action of uniform randomly selecting an element of S. Given algorithm $\mathcal{A}$, we write $z \xleftarrow{\$} \mathcal{A}(x, y, \ldots)$ to indicate that $\mathcal{A}$ is a (probabilistic) algorithm that outputs z on input $(x, y, \ldots)$.

2.1 Digital Signature and its EUF-CMA Security

A digital signature scheme is defined by the three algorithms, $\mathsf{Gen}, \mathsf{Sign}$, and Verify. The key generation algorithm Gen generates a keypair $(PK, sk) \xleftarrow{\$} \mathsf{Gen}(1^\lambda)$ for a secret key sk and a public key PK. The signing algorithm Sign inputs a message and the secret key, and returns a signature $\sigma \xleftarrow{\$} \mathsf{Sign}(sk, M)$ of the message. The verification algorithm Verify takes a public key and a message with a corresponding signature as input, and returns $\top$ or $\bot$, indicating "accept" or "reject", respectively. We require the usual correctness properties.

We recall the EUF-CMA experiment played by a challenger and a forger $\mathcal{F}$. First, the challenger runs $(PK, sk) \xleftarrow{\$} \mathsf{Gen}(1^\lambda)$ and $\mathcal{F}$ is given PK. Proceeding adaptively, $\mathcal{F}$ requests signatures on messages $M_1, \ldots, M_q \in \{0,1\}^*$ under PK. The challenger responds to each query with a signature $\sigma_i \xleftarrow{\$} \mathsf{Sign}(sk, M_i)$. Eventually, $\mathcal{F}$ outputs the pair (M^*, σ^*). We say that the adversary wins the game if $\mathsf{Verify}(M^*, \sigma^*, PK) = \top$ and $M^* \notin \{M_1, \ldots, M_q\}$. We say that $\mathcal{F}$ (t, q, ϵ)-breaks the EUF-CMA security of the signature if $\mathcal{F}$ runs in time t, makes at most q signing queries, and has success probability ϵ. We say that the signature scheme is EUF-CMA secure if ϵ is negligible for any probabilistic polynomial-time algorithm $\mathcal{F}$.

2.2 Prime Numbers, the RSA-assumption, and Generalized Birthday Bounds

For $x \in \mathbb{N}$ let $\pi(x)$ denote the number of primes between 0 and x. The following lemma is a direct consequence of Chebyshev's bounds on $\pi(x)$.

Lemma 1. $\frac{x}{\log_2 x} < \pi(x) < \frac{2x}{\log_2 x}$

We say that a prime p is a safe prime, if $p = 2p' + 1$ and p' is also prime. Let p and q be two randomly chosen r'-bit safe primes, and let $N = pq$. Let $e \in \mathbb{Z}_{\phi(N)}$ be a random odd prime with $e \neq p', q'$. We say that an algorithm $\mathcal{A}$ (t, ϵ)-breaks the RSA assumption, if $\mathcal{A}$ runs in time t and $\Pr[y^{1/e} \xleftarrow{\$} \mathcal{A}(N, e, y)] \geq \epsilon$. As discussed in the previous papers [12, 13, 10], the definition above is equivalent to a more standard version of the RSA assumption where $e \in \mathbb{Z}_{\phi(N)}$ is a random integer relatively prime to $\phi(N)$ with only polynomial loss in reduction cost. We say that an algorithm $\mathcal{A}$ (t, ϵ)-breaks the RSA assumption, if $\mathcal{A}$ runs in time t and non-negligible ϵ.

We denote with $\mathbb{QR}_N$ the group of quadratic residues modulo N. We recall the following lemmas which is needed for the security proof of our constructions.

Lemma 2. *([17],[3]) There is an efficient algorithm that, on input $y, z \in \mathbb{Z}_N$ and integer $e, f \in \mathbb{Z}$ such that $\gcd(e, f) = 1$ and $z^e \equiv y^f \mod N$, computes $x \in \mathbb{Z}_N$ satisfying $x^e \equiv y \mod N$.*

Lemma 3. *([10]) Let A be a set with $|A| = a$. Let $X_1, \ldots, X_q$ be q independent random variables, taking uniformly random values from A. Then the probability that there exists $m + 1$ pairwise distinct indices $i_1, \ldots, i_{m+1}$ such that $X_{i_1} = \cdots = X_{i_{m+1}}$ is upper bounded by $\frac{q^{m+1}}{a^m}$.*

3 Overview of the Idea of Our Constructions

Here, we explain an underlying idea of our constructions. It is known that the combination of a weakly secure signature scheme and a one-time signature scheme yields an EUF-CMA secure signature scheme. It can be seen as a variant of the generic construction of an EUF-CMA secure signature from a weakly secure signature scheme and a chameleon hash function [14, 18]. It would be interesting to consider what would happen if we used an m-time signature scheme instead of a one-time signature scheme in the above. Even in this case, we can obtain an analogous construction of a signature scheme as we explain below.

The public key of the scheme is $(pk_w, vk_m^{(1)}, \ldots, vk_m^{(2^\eta)})$ where pk_w is the public key of the weakly secure signature scheme and all $vk_m^{(s)}$ $(s \in [2^\eta])$ are verification keys of the m-time signature. The secret key of the scheme is $(sk_w, sk_m^{(1)}, \ldots, sk_m^{(2^\eta)})$ where sk_w is the secret key corresponding to pk_w and $sk_m^{(s)}$ $(s \in [2^\eta])$ are secret keys corresponding to $vk_m^{(s)}$. To sign a message M, a signer first picks a random bit string s with length η by $s \xleftarrow{\$} [2^\eta]$. Then, the signer computes signature σ_w on "message" s by the signing algorithm of the weakly secure signature scheme. The signer also computes signature σ_m on M for $vk_m^{(s)}$ using $sk_m^{(s)}$. The final signature is $\sigma = (\sigma_w, \sigma_m, s)$. The verification algorithm simply checks the validity of σ_w and σ_m.

In fact, the above idea does not work without change. This is simply because 2^η is exponentially large and the above construction needs an exponential

number of public keys. Nevertheless, the idea has a potential advantage over the previous generic constructions. That is, if we take larger m (for example $m = 4$), then we can take smaller η. Concretely, we can take $\eta = \lg(q) + \lambda/m$ where q is the upper bound of the number of signing queries issued by the adversary and λ is the security parameter [10]. Since the size of the signature is $|s|(= \eta) + |\sigma_w| + |\sigma_m|$, this considerably reduces the size of the signature.

The generic construction of short signature schemes from a $(m, 1)$-programmable hash function proposed by [11, 10] can be seen as a realization of the above informal idea. In fact, one can obtain an m-time signature from $(m, 1)$-programmable hash function as suggested in [10]. Since m-time signatures form a wider class than $(m, 1)$-programmable hash functions, we can obtain various short signature schemes that cannot be captured by the generic construction by [11, 10].

Based on the above idea, we construct three short signature schemes from the RSA assumption which are presented in section 4, 5, and 6. Specifically, signature size of our first scheme in section 4 is the same as that of the best known scheme [10], and furhtermore its public key size is significantly shorter than that of [10]. Moreover, our second scheme in section 5 yields the same signature size and better computational efficiency by admitting larger public key size, and our third scheme in section 6 yields further better computational efficiency by only slightly increasing signature size and public key size (compared with our first scheme).

4 Our First Scheme

4.1 Basic Idea

As we discussed in the precious section, one possible approach to constructing a (fully-fledged) short signature scheme is to combine an m-time signature scheme and weakly secure signature scheme. We use the weakly secure signature scheme proposed by [13] in this paper.

One possible choice of m-time signature would be the RSA-based m-time signature considered in [4]. In fact, the construction of $\mathsf{Sig}_{\mathsf{RSA}}[\mathsf{H}_{\mathsf{cfs}}]$ proposed in [10] is closely related to the m-time signature in [4]. Since this choice of an m-time signature leads to a signature scheme with huge public key size, we do not use the scheme here. Instead, we construct a new m-time signature and propose a fully-fledged short signature scheme based on it.

Here, we explain our m-time signature. The verification key of our scheme vk_m consists of the odd primes $e_1, \ldots, e_d$, the product of large two primes $N = pq$, and $h \in \mathbb{Z}_N^*$. Let S be a map $S : \mathcal{M} \to 2^{[d]}$ where $\mathcal{M} = \{0,1\}^l$ is the message space. We assume that for all $M^*, M_1, \ldots, M_m \in \mathcal{M}$, it holds that $S(M^*) \not\subseteq \cup_{i=1}^m S(M_i)$ if $M^* \notin \{M_1, \ldots, M_m\}$. We remark that a map S with such a property can be constructed from an m-cover free family [6, 15]. The signature on a message M is $\sigma_m = h^{1/\prod_{i \in S(M)} e_i}$. The verification algorithm checks whether $\sigma_m^{\prod_{i \in S(M)} e_i} \stackrel{?}{=} h$.

We now explain how to combine our m-time signature with weakly secure scheme in [13]. We need an exponentially large number of verification keys

$vk_m^{(1)}, \ldots, vk_m^{(2^\eta)}$ where $vk_m^{(s)} = (e_1^{(s)}, \ldots, e_d^{(s)})$ if we apply the idea described in the previous section straightforwardly. We can resolve this problem by a technique from [12, 13]. That is, we prepare a pseudorandom function F that can be computed publicly and let $e_j^{(s)} = \mathsf{F}(s||j)$ where $s||j$ is the concatenation of s and j. Since all $\{e_1^{(s)}, \ldots, e_d^{(s)}\}_{s \in [2^\eta]}$ can be computed from F, we do not need these elements in the public key. The public key size of our scheme becomes very short by this idea. Another problem to consider is that the signature $\sigma = (\sigma_m, \sigma_w, s)$ is still longer than that of previous schemes. In the construction below, we reduce the signature length by using the algebraic structure of σ_m and σ_w. As a result, we obtain a signature scheme that achieves the shortest signature length and public key length simultaneously among the signature schemes from the RSA assumption [13, 10].

4.2 Construction

Let S be a map $S : \mathcal{M} \to 2^{[d]}$ where $\mathcal{M} = \{0,1\}^l$ is the message space. We assume that for all $M^*, M_1, \ldots, M_m \in \mathcal{M}$ it holds that $S(M^*) \not\subseteq \cup_{i=1}^m S(M_i)$ if $M^* \notin \{M_1, \ldots, M_m\}$. Such S can be constructed using a cover free family [6, 15] if $d \geq 16m^2 l$. See Appendix A for the details. We define the scheme as follows.

$\mathsf{Gen}(1^\lambda)$**:** It picks two large safe r'-bit primes p and q, and sets $N = pq$. Then it chooses a random key K for the pseudorandom function $\mathsf{PRF} : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^r$ and picks $c \xleftarrow{\$} \{0,1\}^r$, where $r = \lfloor \log_2 N \rfloor - 1$. These values define a function F as $\mathsf{F}(z) = \mathsf{PRF}_K(\mu, z) \oplus c$ where μ, called the resolving index of z, denotes the smallest positive integer such that $\mathsf{PRF}_K(\mu, z) \oplus c$ is an odd prime. Here $\oplus$ denotes the bit-wise XOR operation, and we interpret r-bit string returned by F as an integer in the obvious way. Finally, it picks $h \xleftarrow{\$} \mathbb{Z}_N^*$. The public key is $PK = (N, h, K, c)$, the secret key is $sk = (PK, p, q)$.

In the following, we define $\mathsf{P} : \{0,1\}^\eta \to \mathbb{N}$ as $\mathsf{P}(s)$ for $\mathsf{P}(s) = \prod_{i=1}^\eta \mathsf{F}(s|_i)$ where $s|_i$ is the i-th prefix of s, i.e., the bit string consisting of the first i bits of s. We also define $s|_0 = \emptyset$, where $\emptyset$ is the empty string, for technical reasons. We define another function $\mathsf{Q} : \{0,1\}^\eta \times 2^{[d]} \to \mathbb{N}$ as $\mathsf{Q}(s, S) = \prod_{i \in S} \mathsf{F}(s||i)$ where S is a subset of $[d]$ and $s||i$ denotes concatenation of a bit string s and $i \in [d]$. In this case, we regard i as a bit string.

$\mathsf{Sign}(sk, M)$**:** It first picks random $s \xleftarrow{\$} \{0,1\}^\eta$ and computes $\mathsf{F}(t)$ for $t \in (\cup_{i \in [\eta]} \{s|_i\}) \cup (\cup_{i \in S(M)} \{s||i\})$. If the resolving index of t is more than r^2 or $\mathsf{F}(t)$ divides $\phi(N)$ for some $t \in (\cup_{i \in [\eta]} \{s|_i\}) \cup (\cup_{i \in S(M)} \{s||i\})$, then it outputs $((p, q), s)$. [3] Otherwise it computes

$$\sigma = h^{1/\mathsf{P}(s)\mathsf{Q}(s, S(M))},$$

where inverse of $\mathsf{P}(s)\mathsf{Q}(s, S(M))$ is computed modulo the order $\phi(N) = (p-1)(q-1)$ of the multiplicative group $\mathbb{Z}_N^*$. The signature is (σ, s).

[3] The probability of these events happen is negligible as proven in the security proof of the scheme. Thus this step can be ignored in practice.

$\mathsf{Verify}(M,(\sigma,s),PK)$**:** Given a signature (σ,s), it first checks whether resolving index of t is more than r^2 or $2\mathsf{F}(t)+1$ divides N (which is equivalent to $\mathsf{F}(t)$ divides $\phi(N)$) for some $t \in (\cup_{i\in[\eta]}\{s|_i\}) \cup (\cup_{i\in S(M)}\{s||i\})$. If it holds, it outputs $\top$ if $\sigma=(p,q)$ and otherwise $\bot$. Next, it returns $\top$ if

$$\sigma^{\mathsf{P}(s)\mathsf{Q}(s,S(M))} = h.$$

Otherwise it returns $\bot$.

Correctness. The correctness can be verified by the following equation:

$$\sigma^{\mathsf{P}(s)\mathsf{Q}(s,S(M))} = h^{\mathsf{P}(s)\mathsf{Q}(s,S(M))/\mathsf{P}(s)\mathsf{Q}(s,S(M))} = h.$$

4.3 Security

In this subsection, we prove the following theorem which establishes the security of the scheme.

Theorem 1. *Let* PRF *be a* (t'',ϵ'')*-secure pseudo-random function. Suppose there exists a forger* $\mathcal{F}$ *who* (t,q,ϵ)*-breaks the EUF-CMA security of the above scheme. Then there exists an adversary* $\mathcal{A}$ *that* (t',ϵ')*-breaks the RSA assumption with* $t \approx t'$ *and* $\epsilon \leq (q+1)\eta d\big(4r^2\epsilon' + 3\epsilon'' + \frac{r(q+1)^2(\eta+d)^2}{2^{r-1}}\big) + \frac{q^{m+1}}{2^{m\eta}}$.

In the following, let M_k denote the k-th query to the signing oracle, and let (σ_k, s_k) denote the reply. Let (M^*,σ^*,s^*) be the forgery output of $\mathcal{F}$. We distinguish between two types of forgers. A type1 forger returns (M^*,σ^*,s^*) such that $s^* = s_k$ for some $k \in [q]$. A type2 forger returns (M^*,σ^*,s^*) such that $s^* \neq s_k$ for all $k \in [q]$.

The following lemma proves security against Type1 forger.

Lemma 4. *Let* $\mathcal{F}$ *be a Type1 forger that* (t,q,ϵ)*-breaks the existential unforgeability of our scheme. Then there exists an adversary* $\mathcal{A}$ *that* (t',ϵ')*-breaks the RSA assumption with* $t \approx t'$ *and* $\epsilon' \geq \frac{1}{4r^2}\big(\frac{1}{qd}(\epsilon - \frac{q^{m+1}}{2^{m\eta}}) - 3\epsilon'' - \frac{q(\eta+d)(2r+1+rq(\eta+d))}{2^r}\big)$.

Proof. In the following let X_i denote the probability that $\mathcal{F}$ is successful in Game i and the challenger does not abort.

Game 0. We define Game 0 as the EUF-CMA experiment between the challenger and the forger $\mathcal{F}$. By definition we have $\Pr[X_0] = \epsilon$.

Game 1. In this game, the challenger aborts if there exist at least $m+1$ indices $k_1,\ldots,k_{m+1} \in [q]$ such that $s_k = s_{k'}$ for all $k, k' \in \{k_1,\ldots,k_{m+1}\}$. We denote this event by $\mathsf{Abort}_{\mathsf{mColl}}$. We know $\Pr[\mathsf{Abort}_{\mathsf{mColl}}] \leq \frac{q^{m+1}}{2^{m\eta}}$ from Lemma 3. Thus we have $\Pr[X_1] \geq \Pr[X_0] - \frac{q^{m+1}}{2^{m\eta}}$.

Game 2. In this game, the challenger chooses randomness $s_1,\ldots,s_q$ and guesses $k^* \xleftarrow{\$} [q]$ such that $s_{k^*} = s^*$ at the beginning of the game. The challenger aborts if $\mathcal{F}$ outputs a forgery (M^*,σ^*,s^*) with $s_{k^*} \neq s^*$. Since $s^* \in \{s_i\}_{i=1}^q$, we have $\Pr[X_2] \geq \Pr[X_1]/q$.

Game 3. In this game, the challenger chooses $j^* \xleftarrow{\$} [d]$ before setting the public key and aborts if $j^* \notin S(M^*)$ or $j^* \in S(M_k)$ for some $k \in \{k \mid s_k = s_{k^*}\}$. Recall that $\{k \mid s_k = s_{k^*}\} \leq m$, so $S(M^*) \not\subseteq \cup_{k \in \{k | s_k = s_{k^*}\}} S(M_k)$ from the property of S. Thus there exists at least one $j' \in [d]$ such that $j' \in S(M^*)$ and $j' \notin S(M_k)$ for all $k \in \{k \mid s_k = s_{k^*}\}$. We have $\Pr[X_3] \geq \Pr[X_2]/d$.

Game 4. Let E_{all} be $E_{all} = (\cup_{(i,j) \in [q] \times [\eta]} \{s_i|_j\}) \cup (\cup_{(i,j) \in [q] \times [d]} \{s_i||j\})$ in the following. The challenger in this game proceeds just like the challenger in the previous game, except that we add an abort condition. The challenger aborts if for some $t \in E_{all}$, the resolving index μ is greater than r^2. We denote this event with Abort_μ. Let us assume PRF_K is replaced with a truly random function, and let us consider the probability of not finding a prime by evaluating the random function r^2 times and computing the exclusive or with c. This is equivalent to sampling r^2 uniform r-bit strings. Lemma 1 tells us that the probability of finding a prime by sampling r random bits is at least $1/r$, thus the probability of not finding a prime in r^2 trials is at most $(1 - 1/r)^{r^2}$. Since the challenger has to compute F at most $q(\eta + d)$ times, so we can therefore construct an adversary distinguishing PRF_K from a random function with probability at least $\epsilon_\mu \geq \mathsf{Abort}_\mu - q(\eta + d)(1 - 1/r)^{r^2} \geq \mathsf{Abort}_\mu - q(\eta + d)/2^r$, where the latter inequality uses that $(1 - 1/r)^r \leq 1/2$ for all $r \in \mathbb{N}$. Since we must have $\epsilon_\mu \leq \epsilon''$, this implies $\Pr[X_4] \geq \Pr[X_3] - \epsilon'' - q(\eta + d)/2^r$.

Game 5. In this game, the challenger aborts if there exists $t \in E_{all}$ such that $\mathsf{F}(t)$ divides $\phi(N)$. We denote this event by $\mathsf{Abort}_{\mathsf{div}}$. Recall that $\phi(N) = 4p'q'$ and that F returns only odd primes. Again replacing PRF_K with a truly random function, the probability that one out of at most $q(\eta + d)$ randomly chosen odd r-bit primes equals one of the two odd primes dividing $\phi(N)$ is at most $(q(\eta + d)2r)/2^r$ by Lemma 1. Now consider the case where the truly random function is instantiated with PRF_K, and suppose that a collision occurs with probability $\Pr[\mathsf{Abort}_{\mathsf{div}}]$. Then this would allow an attack distinguishing PRF_K from a random function with probability at least $\epsilon_{\mathsf{div}} \geq \Pr[\mathsf{Abort}_{\mathsf{div}}] - (q(\eta + d)2r)/2^r$. Since we have $\epsilon_{\mathsf{div}} \leq \epsilon''$, this implies $\Pr[X_5] \geq \Pr[X_4] - \Pr[\mathsf{Abort}_{\mathsf{div}}] \geq \Pr[X_4] - \epsilon'' - (q(\eta + d)2r)/2^r$.

Game 6. In the following, let $E = \cup_{i=1}^{q} \{s_i\}$, $E^* = E \backslash \{s_{k^*}\}$. In this game the challenger picks $y \xleftarrow{\$} \mathbb{Z}_N^*$ and sets h by

$$h = y^{\mathsf{P}(s_{k^*})\mathsf{Q}(s_{k^*}, [d] \backslash \{j^*\}) \cdot \prod_{t \in E^*} \mathsf{P}(t)\mathsf{Q}(t, [d])}.$$

The distribution of the public key is unchanged from the previous game. This change is only conceptual, so we have $\Pr[X_6] = \Pr[X_5]$.

Game 7. Now the challenger computes a signature σ_k on some chosen-message M_k as

$$\sigma_k = \begin{cases} y^{\mathsf{Q}(s_{k^*}, [d] \backslash \{S(M_k) \cup \{j^*\}\}) \cdot \prod_{t \in E^*} \mathsf{P}(t)\mathsf{Q}(t, [d])} & s_k = s_{k^*} \\ y^{\mathsf{P}(s_{k^*})\mathsf{Q}(s_{k^*}, [d] \backslash \{j^*\}) \cdot \mathsf{Q}(s_k, [d] \backslash S(M_k)) \cdot \prod_{t \in E_k^*} \mathsf{P}(t)\mathsf{Q}(t, [d])} & s_k \neq s_{k^*} \end{cases}$$

where $E_k^* = E^* \backslash \{s_k\}$. It is easy to check that $\sigma_k = h^{1/\mathsf{P}(s_k)\mathsf{Q}(s_k,S(M_k))}$. In the above, we used the fact that $j^* \notin S(M_k)$ if $s_k = s_{k^*}$. This change is only conceptual, so we have $\Pr[X_7] = \Pr[X_6]$.

Game **8.** The challenger in this game aborts if there exists $t, t' \in E_{all}$ such that $\mathsf{F}(t) = \mathsf{F}(t')$ and $t \neq t'$. This event is denoted with $\mathsf{Abort}_{\mathsf{col}}$. Recall that $\mathsf{F}(z) = \mathsf{PRF}_K(\mu, z) \oplus c$, where μ is incremented until $\mathsf{PRF}_K(\mu, z) \oplus c$ is prime. Let us again assume PRF_K is replaced with a truly random function. Then evaluating F is equivalent to sampling a uniformly random r-bit prime. There are at least $2^r/r$ such primes by Lemma 1, and at most $q(\eta + d)$ primes are sampled. Applying Lemma 3, we conclude that the collision probability for a truly random function is at most $rq^2(\eta + d)^2 \cdot 2^{-r}$. Now consider the case where the truly random function is instantiated with PRF_K, and suppose that a collision occurs with probability $\Pr[\mathsf{Abort}_{\mathsf{col}}]$. Then this would allow an attack distinguishing PRF_K from a random function with probability at least $\epsilon_{\mathsf{col}} \geq \Pr[\mathsf{Abort}_{\mathsf{col}}] - rq^2(\eta+d)^2/2^r$. Since we have $\epsilon_{\mathsf{col}} \leq \epsilon''$, this implies $\Pr[X_8] = \Pr[X_7] - \Pr[\mathsf{Abort}_{\mathsf{col}}] \geq \Pr[X_7] - \epsilon'' - rq^2(\eta + d)^2/2^r$.

Game **9.** In this game, the challenger chooses $\mu^* \xleftarrow{\$} [r^2]$ in advance and aborts if μ^* is not resolving index of $s_{k^*}||j^*$. Due to the changes introduced in the Game 4 we know that the resolving index of $s_{k^*}||j^*$ lies in the interval $[1, r^2]$. Thus we have $\Pr[X_9] \geq \Pr[X_8]/r^2$.

Game **10.** Recall that c is uniformly distributed, and we abort if μ^* is not the resolving index of $s_{k^*}||j^*$. The latter implies that $\mathsf{PRF}(\mu^*, s_{k^*}||j^*)$ is prime, thus e has the distribution of uniformly random prime. In this game, the challenger determines c differently. Instead of sampling c at random, the challenger sets $c = \mathsf{PRF}(\mu^*, s_{k^*}||j^*) \oplus e$, where e is the random r-bit prime the challenger chooses. Observe that this defines $\mathsf{F}(s_{k^*}||j^*) = e$. The distribution of μ^*, c, and e is not altered. Thus We have $\Pr[X_{10}] = \Pr[X_9]$.

The RSA adversary. We replace the challenger in Game 10 with RSA adversary $\mathcal{A}$. $\mathcal{A}$ receives a RSA challenge (N', e', y') as input and aborts if e' is not an odd prime or $e' > 2^r$. Otherwise $\mathcal{A}$ sets $N = N', e = e'$ and proceeds like the challenger in Game 10. Recall that $s_{k^*} = s^*$, $\mathsf{F}(s^*||j^*) = e$ and $j^* \in S(M^*)$. Otherwise $\mathcal{A}$ aborts as the challenger does in Game 10. Since we have set $r = \lfloor \log_2 N \rfloor - 1$, the probability that $e \geq 2^r$ is at most $1/4$. Thus, the success probability of $\mathcal{A}$ is at least $\Pr[X_{10} \wedge e < 2^r] \geq \frac{1}{4}\Pr[X_{10}]$.

Answering the Signing Queries. Due to the changes introduced in the Games 4 to 7, $\mathcal{A}$ can answer signing queries without the knowledge of the factorization of N.

Extracting the Solution to the RSA Challenge. Eventually, $\mathcal{F}$ returns a forgery (M^*, σ^*, s^*), from which $\mathcal{A}$ extracts the solution to the RSA challenge as follows. First observe that

$$\begin{aligned}
\sigma^* &= h^{1/\mathsf{P}(s^*)\mathsf{Q}(s^*,S(M^*))} = y^{\left(\prod_{t\in E^*}\mathsf{P}(t)\mathsf{Q}(t,[d])\right)\cdot\mathsf{Q}(s^*,[d]\backslash\{j^*\})/\mathsf{Q}(s^*,S(M^*))} \\
&= y^{\left(\prod_{t\in E^*}\mathsf{P}(t)\mathsf{Q}(t,[d])\right)\cdot\mathsf{Q}(s^*,[d]\backslash S(M^*))/\mathsf{F}(s^*||j^*)} \\
&= y^{\left(\prod_{t\in E^*}\mathsf{P}(t)\mathsf{Q}(t,[d])\right)\cdot\mathsf{Q}(s^*,[d]\backslash S(M^*))/e} = y^{f/e}
\end{aligned}$$

where $f = \mathsf{Q}(s^*, [d]\backslash S(M^*)) \cdot \prod_{t \in E^*} \mathsf{P}(t)\mathsf{Q}(t, [d])$. Then we can see that $(\sigma^*)^e = y^f$ holds from the above equation. Furthermore, $\gcd(e, f) = 1$ by Game 8. Thus we can apply Lemma 2 and extract $y^{1/e}$ which is answer to the RSA challenge from σ^*.

The following lemma proves security against Type2 forger.

Lemma 5. *Let $\mathcal{F}$ be a Type2 forger that (t, q, ϵ)-breaks the existential unforgeability of our scheme. Then there exists an adversary $\mathcal{A}$ that (t', ϵ')-breaks the RSA assumption with $t \approx t'$ and $\epsilon' \geq \frac{1}{4r^2}\Big(\frac{1}{(q+1)\eta}\big(\epsilon - 2\epsilon'' - \frac{q(\eta+d)(2r+1)}{2^r}\big) - 2\epsilon'' - \frac{r(q+1)^2(\eta+d)^2+1}{2^r}\Big)$.*

Proof. Let X_i denote the probability that $\mathcal{F}$ is successful in Game i and the challenger does not abort.

Game 0. We define Game 0 as the EUF-CMA experiment between the challenger and the forger $\mathcal{F}$. By definition we have $\Pr[X_0] = \epsilon$.

Game 1. Let E_{all} be $E_{all} = (\cup_{(i,j)\in[q]\times[\eta]}\{s_i|_j\}) \cup (\cup_{(i,j)\in[q]\times[d]}\{s_i||j\})$ in the following. In this game, the challenger aborts if for some $t \in E_{all}$, the resolving index μ is greater than r^2. As the proof of Lemma 4, we have $\Pr[X_1] \geq \Pr[X_0] - \epsilon'' - q(\eta + d)/2^r$.

Game 2. In this game, the challenger aborts if there exists $t \in E_{all}$ such that $\mathsf{F}(t)$ divides $\phi(N)$. As the proof of Lemma 4, we have $\Pr[X_2] \geq \Pr[X_1] - \epsilon'' - (q(\eta + d)2r)/2^r$.

Game 3. In this game, the challenger chooses the randomness $s_1, \ldots, s_q$ in advance. Let $E = \cup_{i=1}^{q}\{s_i\}$. The challenger picks $y \xleftarrow{\$} \mathbb{Z}_N^*$ and sets public key h by

$$h = y^{\prod_{t\in E} \mathsf{P}(t)\mathsf{Q}(t,[d])}.$$

The distribution of the public key is unchanged from the previous game. This change is only conceptual, so we have $\Pr[X_3] = \Pr[X_2]$.

Game 4. Now the challenger computes a signature σ_k on some chosen-message M_k as

$$\sigma_k = y^{\mathsf{Q}(s_k,[d]\backslash S(M_k))\prod_{t\in E_k} \mathsf{P}(t)\mathsf{Q}(t,[d])}$$

where $E_k = E\backslash\{s_k\}$. It is easy to check that $\sigma_k = h^{1/\mathsf{P}(s_k)\mathsf{Q}(s_k,S(M_k))}$. This change is only conceptual, so we have $\Pr[X_4] = \Pr[X_3]$.

Game 5. In this game the challenger guesses the shortest prefix of s^* that differs from all prefixes of $s_1, \ldots, s_q$. Note that this prefix must exist, because the Type2-forger will return a forgery (M^*, σ^*, s^*) with $s^* \notin \{s_1, \ldots, s_q\}$. To this end, the challenger proceeds as follows. If $q = 0$, it samples a bit $\psi \xleftarrow{\$} \{0, 1\}$ at random, and aborts if the forger returns s^* with $s^*|_1 \neq \psi$. If $q \geq 0$, the challenger picks $i \in [q]$ and $j \in [\eta]$ and sets $\psi = s_i|_{j-1}||b$, where b is the complement of the j-th bit of s_i. (Recall that we defined the 0-th prefix as the empty string $\emptyset$, thus $s_i|_0 = \emptyset$.) The challenger aborts if either ψ is a prefix of some $s_i \in \{s_1, \ldots, s_q\}$, that is, there exists (i', j') such that $\psi = s_{i'}|_{j'}$, or if the forger returns (M^*, σ^*, s^*) such that ψ is not a prefix of s^*. If $q = 0$,

then the challenger aborts with probability 1/2. Otherwise there are at most $q\eta$ possible choices of ψ. Thus we have $\Pr[X_5] = \Pr[X_4]/(q+1)\eta$.

Game **6.** We add an abort condition. If $\mathsf{F}(\psi)|\prod_{t\in E}\mathsf{P}(t)\mathsf{Q}(t,[d])$, or (equivalently) $\mathsf{F}(\psi) = \mathsf{F}(t)$ for some $t \in E_{all}$, then the challenger aborts. Note that $\psi \neq t$ for all $t \in E_{all}$. As the proof of Lemma 4, we have $\Pr[X_6] \geq \Pr[X_5] - \epsilon'' - r(q+1)^2(\eta+d)^2/2^r$.

Game **7.** We introduce a number of changes to the challenger.

- The challenger aborts if the resolving index of ψ is greater than r^2.
- The challenger guesses resolving index of ψ as $\mu^* \xleftarrow{\$} [r^2]$ and aborts if μ^* is not the resolving index of ψ.
- Instead of sampling c at random, we set $c = \mathsf{PRF}(\mu^*, \psi) \oplus e$, where e is the uniformly random r-bit prime that the challenger chooses.

With the same arguments as the proof of Lemma 4 we have $\Pr[X_7] \geq 1/r^2(\Pr[X_6] - \epsilon'' - 1/2^r)$.

The RSA adversary. We replace the challenger in Game 7 with the RSA adversary $\mathcal{A}$. $\mathcal{A}$ receives an RSA challenge (N', e', y') as input and aborts if e' is not an odd prime or $e' > 2^r$. Otherwise $\mathcal{A}$ sets $N = N', e = e'$ and proceeds like the challenger in Game 7. Recall that we have $\mathsf{F}(\psi) = e$ now. As the proof of Lemma 4, the success probability of $\mathcal{A}$ is at least $\frac{1}{4}\Pr[X_7]$.

Answering the Signing Queries. Due to the changes introduced in the Game 1 to 4, $\mathcal{A}$ can answer signing queries without the knowledge of the factorization of N.

Extracting the Solution to the RSA Challenge. Eventually, $\mathcal{F}$ returns a forgery (M^*, σ^*, s^*), from which $\mathcal{A}$ extracts the solution to the RSA challenge as follows. In the case where resolving index of t is more than r^2 or $\mathsf{F}(t)|\phi(N)$ for some $t \in (\cup_{i\in[\eta]}\{s^*|_i\}) \cup (\cup_{i\in S(M^*)}\{s^*||i\})$, $\sigma = (p, q)$ if $\mathcal{F}$ is successful. Thus $\mathcal{A}$ can efficiently compute $y^{1/e}$ from the output of $\mathcal{F}$ in this case. Otherwise,

$$\begin{aligned}\sigma^* &= h^{1/\mathsf{P}(s^*)\mathsf{Q}(s^*,S(M^*))} = y^{\left(\prod_{t\in E}\mathsf{P}(t)\mathsf{Q}(t,[d])\right)/\mathsf{P}(s^*)\mathsf{Q}(s^*,S(M^*))} \\ &= y^{\left(\prod_{t\in E}\mathsf{P}(t)\mathsf{Q}(t,[d])\right)/\left(z\cdot\mathsf{F}(\psi)\right)} = y^{\left(\prod_{t\in E}\mathsf{P}(t)\mathsf{Q}(t,[d])\right)/ez}\end{aligned}$$

where $z = \mathsf{Q}(s^*, S(M^*)) \cdot \prod_{\{i\in[\eta]|s^*|_i\neq\psi\}}\mathsf{F}(s^*|_i)$ holds. Thus we have $\left((\sigma^*)^z\right)^e = y^{\prod_{t\in E}\mathsf{P}(t)\mathsf{Q}(t,[d])}$. Since $\gcd(e, \prod_{t\in E}\mathsf{P}(t)\mathsf{Q}(t,[d])) = 1$ by Game 6, we can apply Lemma 2 and extract $y^{1/e}$.

5 Our Second Scheme

Our first scheme suffers from its heavy signing and verification algorithms while providing very short public key size. This is because the signing and verification algorithms need the generation of a large number of primes. On the other hand, $\mathsf{Sig}_{\mathsf{RSA}}[\mathsf{H}_{\mathsf{cfs}}]$ in [10] has an opposite property. That is, the signing and verification algorithms are more efficient than our scheme, but the public key size is huge. In this section, we propose a generalized version of these schemes. The scheme has parameters u_1 and u_2 with condition $u_1 \cdot u_2 = d$ where d is some constant

depending on m and λ. If we take u_1 smaller, the public key size becomes smaller and the signing and verification algorithms become heavier. On the other hand, if we take u_1 larger, the public key size becomes larger and the signing and verification algorithm becomes more efficient. Especially, if we set $(u_1, u_2) = (1, d)$, then the scheme can be seen as a slight variant of our first scheme. Similarly, if we set $(u_1, u_2) = (d, 1)$, then we can obtain a slight variant of $\mathsf{Sig}_{\mathsf{RSA}}[\mathsf{H}_{\mathsf{cfs}}]$ in [10]. Furthermore, we can also construct a stateful version of the scheme. The scheme is more efficient than the above scheme except for its slightly larger signature size.

CONSTRUCTION. Let S be a map $S : \mathcal{M} \to 2^{[u_1]\times[u_2]}$ where $\mathcal{M} = \{0,1\}^l$ is the message space. We assume that for all $M^*, M_1, \ldots, M_m \in \mathcal{M}$ it holds that $S(M^*) \not\subseteq \cup_{i=1}^m S(M_i)$ if $M^* \notin \{M_1, \ldots, M_m\}$. Such S can be constructed using cover free family [6, 15] as in [20] if $u_1 u_2 \geq 16m^2 l$. See Appendix A for the details. We define the scheme as follows.

$\mathsf{Gen}(1^\lambda)$**:** It picks two large safe r'-bit primes p and q, and sets $N = pq$. Then it chooses a random key K for the pseudorandom function $\mathsf{PRF} : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^r$ and picks $c \xleftarrow{\$} \{0,1\}^r$, where $r = \lfloor \log_2 N \rfloor - 1$. These values define functions $\mathsf{F} : \{0,1\}^* \to \mathbb{N}$, $\mathsf{P} : \{0,1\}^\eta \to \mathbb{N}$, and $\mathsf{Q} : \{0,1\}^\eta \times 2^{[u_2]} \to \mathbb{N}$ as in section 4. Finally, it picks $h', h_1, \ldots, h_{u_1} \xleftarrow{\$} \mathbb{QR}_N$. The public key is $PK = (N, h', h_1, \ldots, h_{u_1}, K, c)$, the secret key is $sk = (PK, p, q)$.

$\mathsf{Sign}(sk, M)$**:** It first picks random $s \xleftarrow{\$} \{0,1\}^\eta$ and computes $\mathsf{F}(t)$ for $t \in (\cup_{i\in[\eta]}\{s|_i\}) \cup (\cup_{j\in S'(M)}\{s||j\})$ where $S'(M) = \{j | \exists i, (i,j) \in S(M)\}$. Let $e_j = \mathsf{F}(s||j)$. If the resolving index of t is more than r^2 or $\mathsf{F}(t)$ divides $\phi(N)$ for some $t \in (\cup_{i\in[\eta]}\{s|_i\}) \cup (\cup_{j\in S'(M)}\{s||j\})$, then it outputs $((p,q), s)$. If $\gcd(\mathsf{P}(s), \mathsf{Q}(s, S'(M))) \neq 1$, then it outputs $((p,q), s)$. [4] Otherwise it computes

$$\sigma = (h')^{1/\mathsf{P}(s)} \cdot \prod_{(i,j)\in S(M)} h_i^{1/e_j}$$

where inverse of $\mathsf{P}(s)$ and e_j is computed modulo the order $\phi(N) = (p-1)(q-1)$ of the multiplicative group $\mathbb{Z}_N^*$. The signature is (σ, s).

$\mathsf{Verify}(M, (\sigma, s), PK)$**:** Given a signature (σ, s), it first checks whether resolving index of t is more than r^2 or $2\mathsf{F}(t)+1$ divides N for some $t \in (\cup_{i\in[\eta]}\{s|_i\}) \cup (\cup_{j\in S'(M)}\{s||j\})$, or $\gcd(\mathsf{P}(s), \mathsf{Q}(s, S'(M))) \neq 1$. If one of the above holds, it outputs $\top$ if $\sigma = (p,q)$ and otherwise $\bot$. Next, it returns if

$$\sigma^{\mathsf{P}(s)\mathsf{Q}(s,S'(M))} = (h')^{\mathsf{Q}(s,S'(M))} \cdot \prod_{(i,j)\in S(M)} h_i^{\mathsf{P}(s)\mathsf{Q}(s,S'(M)\setminus\{j\})}.$$

Otherwise it returns $\bot$.

[4] Similarly to our first scheme, this step can be ignored in practice.

Correctness. The correctness can be verified by the following equation:

$$\begin{aligned}\sigma^{\mathsf{P}(s)\mathsf{Q}(s,S'(M))} &= \Big((h')^{1/\mathsf{P}(s)} \cdot \prod_{(i,j)\in S(M)} h_i^{1/\mathsf{F}(s||j)}\Big)^{\mathsf{P}(s)\mathsf{Q}(s,S'(M))} \\ &= (h')^{\mathsf{Q}(s,S'(M))} \cdot \prod_{(i,j)\in S(M)} h_i^{\mathsf{P}(s)\mathsf{Q}(s,S'(M)\setminus\{j\})}\end{aligned}$$

The following theorem establishes the security of the scheme. The theorem can be proven by a similar argument to the proof of Theorem 1 and [10]. We omit the proof due to a lack of space.

Theorem 2. *Let* PRF *be a* (t'', ϵ'')*-secure pseudo-random function. Suppose there exists a forger* $\mathcal{F}$ *who* (t, q, ϵ)*-breaks the EUF-CMA security of the above scheme. Then there exists an adversary* $\mathcal{A}$ *that* (t', ϵ')*-breaks the RSA assumption with* $t \approx t'$ *and* $\epsilon \leq (q+1)\eta u_1 u_2\big(4r^2\epsilon' + 3\epsilon'' + \frac{r(q+1)^2(\eta+u_2)^2}{2^{r-1}}\big) + \frac{q^{m+1}}{2^{m\eta}}$.

Stateful Version of The Scheme. We can also construct a stateful version of the above scheme. The scheme is more efficient than the above scheme except for the signature size, which is slightly larger than the above scheme. There are two reasons why we can obtain a more efficient scheme in the stateful setting. The first reason is that we can remove the computation of $\mathsf{P}(s)$ from the above. Conceptually, it is equivalent to removing the weakly secure signature scheme in [13] from the above construction. Instead, we use a trick from [12], which leads to a more efficient scheme. The second reason is that we can use a one-time signature instead of the m-time signature ($m \geq 2$) in the stateful setting. This reduces the public key and signing and verification cost. See the full version of this paper for the details.

6 Our Third Scheme

While providing a short signature size, the signing and verification algorithms of our first and second schemes are heavy if we want the public key size to be small. In this section, we propose another signature scheme that provides a shorter public key and a more efficient signing and verification algorithm using a chameleon hash-like technique [7, 10]. A signature of the scheme is longer than that of our first and second schemes, but still much shorter than that of the scheme in [13]. The structure of the scheme can be seen as a hybrid of our first scheme and $\mathsf{Sig}_{\mathsf{RSA}}[\mathsf{H}_{\mathsf{rand}}]$ in [10]. Compared with $\mathsf{Sig}_{\mathsf{RSA}}[\mathsf{H}_{\mathsf{rand}}]$ in [10], the scheme has the same size of the signatures. The public key size of the scheme is smaller than that of the other scheme, while the signing and verification algorithms are slightly heavier than the other scheme. We can also construct a stateful version of the scheme. The scheme is more efficient than the above scheme except for its slightly larger signature size.

Construction. We define the scheme as follows. In the following, let $[X]_{2^l} \in \mathbb{Z}$ denote a canonical interpretation of a field element $X \in \mathbb{F}_{2^l}$ as an integer between

0 and $2^l - 1$. We assume that X and $[X]_{2^l}$ are efficiently computable from one another. The message space of the scheme is $\mathcal{M} = \{0,1\}^l$.

$\mathsf{Gen}(1^\lambda)$**:** It picks two large safe r'-bit primes p and q, and sets $N = pq$. Then it chooses a random key K for the pseudorandom function $\mathsf{PRF} : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^r$ and picks $c \xleftarrow{\$} \{0,1\}^r$, where $r = \lfloor \log_2 N \rfloor - 1$. These values define functions $\mathsf{F} : \{0,1\}^* \to \mathbb{N}$, $\mathsf{P} : \{0,1\}^\eta \to \mathbb{N}$, and $\mathsf{Q} : \{0,1\}^\eta \times 2^{[2m]} \to \mathbb{N}$ as in section 4. Finally, it picks $h', h_1, \ldots, h_m \xleftarrow{\$} \mathbb{QR}_N$. The public key is $PK = (N, h', h_1, \ldots, h_m, K, c)$, the secret key is $sk = (PK, p, q)$.

$\mathsf{Sign}(sk, M)$**:** It first picks random $s \xleftarrow{\$} \{0,1\}^\eta$, $\rho \xleftarrow{\$} \{0,1\}^l$ and computes $\mathsf{F}(t)$ for $t \in (\cup_{i\in[\eta]}\{s|_i\}) \cup (\cup_{j\in[2m]}\{s||j\})$. Let $e_i = \mathsf{F}(s||i)$. If the resolving index of t is more than r^2 or $\mathsf{F}(t)$ divides $\phi(N)$ for some $t \in (\cup_{i\in[\eta]}\{s|_i\}) \cup (\cup_{j\in[2m]}\{s||j\})$, then it outputs $((p,q), s, \rho)$. If $\gcd(\mathsf{P}(s), \mathsf{Q}(s,[2m])) \neq 1$, then it outputs $((p,q), s, \rho)$. [5] Otherwise it computes

$$\sigma = (h')^{1/\mathsf{P}(s)} \cdot \prod_{i\in[2m]} \Big(h_0 \prod_{j\in[m]} h_j^{([iM+\rho]_{2^l})^j}\Big)^{1/e_i}$$

where inverse of $\mathsf{P}(s)$ and e_i is computed modulo the order $\phi(N) = (p-1)(q-1)$ of the multiplicative group $\mathbb{Z}_N^*$. The signature is (σ, s, ρ).

$\mathsf{Verify}(M, (\sigma, s), PK)$**:** Given a signature (σ, s, ρ), it first checks whether resolving index of t is more than r^2 or $2\mathsf{F}(t)+1$ divides N for some $t \in (\cup_{i\in[\eta]}\{s|_i\}) \cup (\cup_{j\in[2m]}\{s||j\})$, or $\gcd(\mathsf{P}(s), \mathsf{Q}(s,[2m])) \neq 1$. If one of the above holds, it returns $\top$ if $\sigma = (p,q)$ and otherwise $\bot$. Next, it returns $\top$ if

$$\sigma^{\mathsf{P}(s)\mathsf{Q}(s,[2m])} = (h')^{\mathsf{Q}(s,[2m])} \cdot \prod_{i\in[2m]} \Big(h_0 \prod_{j\in[m]} h_j^{([iM+\rho]_{2^l})^j}\Big)^{\mathsf{P}(s)\mathsf{Q}(s,[2m]\setminus\{i\})}.$$

Otherwise it returns $\bot$.

Correctness. The correctness can be verified by the following equation:

$$\begin{aligned}\sigma^{\mathsf{P}(s)\mathsf{Q}(s,[2m])} &= \Big((h')^{1/\mathsf{P}(s)} \cdot \prod_{i\in[2m]} \Big(h_0 \prod_{j\in[m]} h_j^{([iM+\rho]_{2^l})^j}\Big)^{1/\mathsf{F}(s||i)}\Big)^{\mathsf{P}(s)\mathsf{Q}(s,[2m])} \\ &= (h')^{\mathsf{Q}(s,[2m])} \cdot \prod_{i\in[2m]} \Big(h_0 \prod_{j\in[m]} h_j^{([iM+\rho]_{2^l})^j}\Big)^{\mathsf{P}(s)\mathsf{Q}(s,[2m]\setminus\{i\})}\end{aligned}$$

The following theorem establishes the security of the scheme. The theorem can be proven by a similar argument to the proof of Theorem 1 and [10]. We omit the proof due to a lack of space.

Theorem 3. *Let* PRF *be a* (t'', ϵ'')*-secure pseudo-random function. Suppose there exists a forger* $\mathcal{F}$ *who* (t, q, ϵ)*-breaks the EUF-CMA security of the above scheme. Then there exists an adversary* $\mathcal{A}$ *that* (t', ϵ')*-breaks the RSA assumption with* $t \approx t'$ *and* $\epsilon \leq 4r^2(q+1)\eta\big(\epsilon' + 3\epsilon'' + \frac{m}{2^{r/2}} + \frac{r(q+1)^2(\eta+u_2)^2}{2^{r-1}}\big) + \frac{q^{m+1}}{2^{m\eta}}$.

[5] Similarly to our first scheme, this step can be ignored in practice.

Stateful Version of The Scheme. As our second scheme, we can consider a stateful version of the above scheme. The scheme is more efficient than the above scheme except for slightly longer signature size. The structure of the scheme can be seen as a combination of (randomized) $(1, 1)$-programmable hash function [11] in [7] with RSA based stateful signature scheme in [12]. See the full version of this paper for the details.

7 Comparison

In the Table 1, we compare our schemes with other RSA based signature schemes under appropriately chosen parameters. We ignore the penalty imposed on the modulus size due to the non-tight reduction in the table. The signature size of our first scheme is the same as that of $\mathsf{Sig}_{RSA}[\mathsf{H}_{\mathsf{cfs}}]$, which is currently the shortest signature scheme. As for the public key size, our first scheme is about 2000 bits, which is about 1/20000 of that of $\mathsf{Sig}_{RSA}[\mathsf{H}_{\mathsf{cfs}}]$, and about half compared with Hohenberger-Waters scheme. However, as we can see, our first scheme requires generation of about 2600 primes, which is impractical. For our second scheme, the public key size is about 1/100 of that of $\mathsf{Sig}_{RSA}[\mathsf{H}_{\mathsf{cfs}}]$ while its signing and verification cost is about 3 times higher than for their scheme. The second scheme indicates that we can considerably reduce the public key size of $\mathsf{Sig}_{RSA}[\mathsf{H}_{\mathsf{cfs}}]$ at the cost of relatively small increase in computational efficiency. We remark that other choices of parameters are also possible for this scheme. For our third scheme, compared with $\mathsf{Sig}_{RSA}[\mathsf{H}_{\mathsf{rand}}]$, the public key is reduced to less than 1/5 whereas the increase in the computational cost is less than 20 percent. We remark that the reduction cost of our first and second scheme is essentially the same as that of $\mathsf{Sig}_{RSA}[\mathsf{H}_{\mathsf{cfs}}]$, and that of our third scheme is also essentially the same as $\mathsf{Sig}_{RSA}[\mathsf{H}_{\mathsf{rand}}]$. We also remark that we can obtain more efficient schemes in the stateful setting. Especially, stateful version of our third scheme is at least as efficient as Hohenberger-Waters scheme [12] in all aspects. See the full version for the details.

References

1. Mihir Bellare and Sara K. Miner. A forward-secure digital signature scheme. In *CRYPTO*, pp. 431–448, 1999.
2. Dan Boneh and Xavier Boyen. Short signatures without random oracles. In *EUROCRYPT*, pp. 56–73, 2004.
3. Ronald Cramer and Victor Shoup. Signature schemes based on the strong rsa assumption. In *ACM Conference on Computer and Communications Security*, pp. 46–51, 1999.
4. Yevgeniy Dodis, Iftach Haitner, and Aris Tentes. On the (in)security of rsa signatures. Cryptology ePrint Archive, Report 2011/087, 2011. `http://eprint.iacr.org/`.
5. Danny Dolev, Cynthia Dwork, and Moni Naor. Non-malleable cryptography (extended abstract). In *STOC*, pp. 542–552, 1991.

Table 1. Comparison of signature schemes based on the RSA assumption.

Signature scheme	Signature size (bits)	Public key size (bits)	Efficiency
Hohenberger-Waters *	$2 \times \lvert\mathbb{Z}_N\rvert = 2048$	$\lvert\mathbb{Z}_N\rvert + \lvert c\rvert + \lvert pk_{ch}\rvert = 4$k	$160 \times \mathsf{P}_{1024}$
$\mathsf{Sig}_{RSA}[\mathsf{H}_{\mathsf{Wat}}]$, [10]	$\lvert\mathbb{Z}_N\rvert + \lvert s\rvert$=1094	$l \times \lvert\mathbb{Z}_N\rvert + \lvert c\rvert = 161$k	$70 \times \mathsf{P}_{1024}$
$\mathsf{Sig}_{RSA}[\mathsf{H}_{\mathsf{cfs}}]$, $(m = 4)$ [10]	$\lvert\mathbb{Z}_N\rvert + \lvert s\rvert$=1074	$16m^2l \times \lvert\mathbb{Z}_N\rvert + \lvert c\rvert = 40$m	$50 \times \mathsf{P}_{1024}$
Ours in Sec. 4, $(m = 4)$	$\lvert\mathbb{Z}_N\rvert + \lvert s\rvert$=1074	$\lvert\mathbb{Z}_N\rvert + \lvert c\rvert$=2k	$2610 \times \mathsf{P}_{1024}$
Ours in Sec. 5, $(m = 4)$ $(u_1, u_2) = (410, 100)$	$\lvert\mathbb{Z}_N\rvert + \lvert s\rvert$=1074	$u_1 \times \lvert\mathbb{Z}_N\rvert + \lvert c\rvert$ =411k	$150 \times \mathsf{P}_{1024}$
$\mathsf{Sig}_{RSA}[\mathsf{H}_{\mathsf{rand}}]$, $(m = 4)$ [10]	$\lvert\mathbb{Z}_N\rvert + \lvert s\rvert + \lvert\rho\rvert$=1234	$(2m^2 + 1) \times \lvert\mathbb{Z}_N\rvert + \lvert c\rvert = 34$k	$50 \times \mathsf{P}_{1024}$
Ours in Sec. 6, $(m = 4)$	$\lvert\mathbb{Z}_N\rvert + \lvert s\rvert + \lvert\rho\rvert$=1234	$(m + 1) \times \lvert\mathbb{Z}_N\rvert + \lvert c\rvert = 6$k	$58 \times \mathsf{P}_{1024}$

* The RSA-based chameleon hash function from [12] was used (adding $1 \times \mathbb{Z}_N$ and $2 \times \mathbb{Z}_N$ to signature size and public key size respectively).

The chosen parameters are $\lambda = 80$, $q = 2^{30}, l = 2\lambda = 160$. We also set $\eta = |s| = \log q + \lambda/m = 50$ so that the term $q^{m+1}/2^{ml}$ is at most $1/2^{\lambda}$ as in [10]. Signatures are instantiated with a modulus of $|N| = 1024$ bits. The description of modulus N and key for PRF are not counted in the public key. We assume l-bit messages with $l = 2\lambda = 160$ in order to provide λ bits of security (to sign longer messages, we can apply a collision resistant hash function first.) The efficiency column counts the dominant operations for signing. $k \times \mathsf{P}_{\mu}$ counts the number of random μ-bit primes that need to be generated in the signing and verification algorithms. (For $\mu >> 60$, $1 \times \mathsf{P}_{\mu}$ takes more time than one exponentiation over modulus N.)

6. Péter L. Erdös, Peter Frankl, and Zoltán Füredi. Families of finite sets in which no set is covered by the union of two others. *J. Comb. Theory, Ser. A*, Vol. 33, No. 2, pp. 158–166, 1982.
7. Marc Fischlin. The cramer-shoup strong-rsasignature scheme revisited. In *Public Key Cryptography*, pp. 116–129, 2003.
8. Rosario Gennaro, Shai Halevi, and Tal Rabin. Secure hash-and-sign signatures without the random oracle. In *EUROCRYPT*, pp. 123–139, 1999.
9. Shafi Goldwasser, Silvio Micali, and Ronald L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM J. Comput.*, Vol. 17, No. 2, pp. 281–308, 1988.
10. Dennis Hofheinz, Tibor Jager, and Eike Kiltz. Short signatures from weaker assumptions. In *ASIACRYPT*, pp. 647–666, 2011.
11. Dennis Hofheinz and Eike Kiltz. Programmable hash functions and their applications. In *CRYPTO*, pp. 21–38, 2008.
12. Susan Hohenberger and Brent Waters. Realizing hash-and-sign signatures under standard assumptions. In *EUROCRYPT*, pp. 333–350, 2009.
13. Susan Hohenberger and Brent Waters. Short and stateless signatures from the rsa assumption. In *CRYPTO*, pp. 654–670, 2009.
14. Hugo Krawczyk and Tal Rabin. Chameleon signatures. In *NDSS*, 2000.
15. Ravi Kumar, Sridhar Rajagopalan, and Amit Sahai. Coding constructions for blacklisting problems without computational assumptions. In *CRYPTO*, pp. 609–623, 1999.
16. John Rompel. One-way functions are necessary and sufficient for secure signatures. In *STOC*, pp. 387–394, 1990.
17. Adi Shamir. On the generation of cryptographically strong pseudorandom sequences. *ACM Trans. Comput. Syst.*, Vol. 1, No. 1, pp. 38–44, 1983.
18. Adi Shamir and Yael Tauman. Improved online/offline signature schemes. In *CRYPTO*, pp. 355–367, 2001.

19. Brent Waters. Efficient identity-based encryption without random oracles. In *EUROCRYPT*, pp. 114–127, 2005.
20. Shota Yamada, Goichiro Hanaoka, and Noboru Kunihiro. Two-dimensional representation of cover free families and its applications: Short signatures and more. In *CT-RSA*, pp. 260–277, 2012.

A Construction of Map S

In many of our schemes, we use a map S with a special property. We describe how we realize this map S from cover free family [6, 15]. Although the idea we describe in this section is not new, we include this section in this paper for completeness.

We begin by recalling the definition of cover-free families. Let S_1, S_2 be sets. We say that S_2 does not cover S_1 if $S_1 \not\subseteq S_2$. Let d, m, α be integers, and let $F = (F_\mu)_{\mu\in[\alpha]}$ be a family of α subsets of $[d]$. We say that F is m-cover free if for any set I containing (up to) m indices $I = \{\mu_1, \ldots, \mu_m\} \subseteq [\alpha]$, it holds that $F_\nu \not\subseteq \cup_{\mu\in I} F_\mu$ for any ν that is not contained in I. In other words, if $|I| \leq m$, then the union $\cup_{\mu\in I} F_\mu$ does not cover F_ν for all $\nu \in [\alpha]\backslash I$. We say that F is w-uniform if $|F_\mu| = w$ for all $\mu \in [\alpha]$. Throughout this paper, we use a parameter in the following lemma.

Lemma 6. *([6, 15]) There is a deterministic polynomial-time algorithm that, on input of integers $m, \alpha = 2^n$, returns $d \in \mathbb{N}$ and the set family $F = (F_\mu)_{\mu\in[\alpha]}$, such that F is m-cover free over $[d]$ and w-uniform, where $d \leq 16m^2 n$ and $w = d/4m$.*

Note that in the case of $m = 1$, we have a cover-free family with smaller parameters. That is, $\alpha = 2^n$, $d = 2n$, and $w = n$. F_μ is defined as $F_\mu = \{2i - 1 + b_i | i \in [n]\}$ where we regard μ as a concatenation of bit strings in a natural way as $\mu = b_1 || \cdots || b_n$ with $b_i \in \{0, 1\}$ for $i \in [n]$. This cover-free family is used in many cryptographic protocols explicitly or implicitly, for example [16, 5].

For Our First Scheme. In our first scheme, we associate a message $M \in \mathcal{M}$ with a subset of $[d]$ by a map $S : \mathcal{M} \to 2^{[d]}$. S should satisfy the following property: "For all $M^*, M_1, \ldots, M_m \in \mathcal{M}$, it holds that $S(M^*) \not\subseteq \cup_{i=1}^m S(M_i)$ if $M^* \notin \{M_1, \ldots, M_m\}$." We can construct a map S with this property by defining S as $S(M) \stackrel{def}{=} F_{H(M)} \subseteq [d]$ where $H : \mathcal{M} \to [\alpha]$ is an injective (or hash) function.

For Our Second Scheme. In our second scheme, we associate a message $M \in \mathcal{M}$ with a subset of $[u_1] \times [u_2]$ by a map $S : \mathcal{M} \to 2^{[u_1]\times[u_2]}$. S should satisfy the following property: "For all $M^*, M_1, \ldots, M_m \in \mathcal{M}$, it holds that $S(M^*) \not\subseteq \cup_{i=1}^m S(M_i)$ if $M^* \notin \{M_1, \ldots, M_m\}$." To construct such a map, we first regard $[d]$ as $[u_1] \times [u_2]$, where u_1 and u_2 are integers satisfying $u_1 \geq u_2$ and $u_1 u_2 \geq d$. (The case for $u_1 \leq u_2$ is analogous.) We regard $i \in [d]$ as an element of $[u_1] \times [u_2]$ by associating it with $(i - u_1(\lceil i/u_1 \rceil - 1), \lceil i/u_1 \rceil)$. Then, all F_μ can be seen as a subset of $[u_1] \times [u_2]$ in a natural way and $(F_\mu)_{\mu\in\alpha}$ can be seen as an m-cover free family over $[u_1] \times [u_2]$. Then we define S as

$S(M) \stackrel{def}{=} F_{H(M)} \subseteq [u_1] \times [u_2]$ where $H : \mathcal{M} \to [\alpha]$ is an injective (or hash) function.

In the constructions, we treat H (and S) as an injective function for simplicity, but it is enough to assume that H is a collision resistant hash for our schemes to be secure. To avoid a birthday attack, we typically set $n = 2\lambda$. Besides, if we require F to be w-uniform, then $|S(M)| = w$ for all $M \in \mathcal{M}$.