Post-Quantum Provably-Secure Authentication and MAC from Mersenne Primes

Anonymous

Submission to CT-RSA 2020

Abstract. This paper presents a novel, yet efficient secret-key authentication and MAC, which provide post-quantum security promise, whose security is reduced to the quantum-safe conjectured hardness of Mersenne Low Hamming Combination (MERS) assumption recently introduced by Aggarwal, Joux, Prakash & Santha (CRYPTO 2018). Our protocols are very suitable to weak devices like smart card and RFID tags.

Keywords: secret-key authentication, MAC, MERS assumption, man-in-the-middle security.

1 Introduction

1.1 Motivation

SECRET KEY AUTHENTICATION AND HB FAMILY. *Secret-key unilateral authentication* protocol is a process by which a *prover* authenticates itself to a *verifier*, where they share a secret. The current best way to construct such a protocol is a challenge-response protocol by a strong pseudo-random function, e.g., AES. A verifier sends a random challenge m and a prover answers its ciphertext $c = \mathsf{AES}_K(m)$.

In recent years such protocols have become an important mechanism for low-cost device authentication with small computational power such as smart cards or radio-frequency identification (RFID) tags. Unfortunately, it is hard to implement the blockcipher-based authentication protocol in such constrained devices. Hopper and Blum [HB01] introduced a two-round secret-key authentication protocol, denoted by HB. The advantages of HB are that implementation requires only bit-wise operations and that the security is based on the hardness of the Learning Parity with Noise (LPN) problem [BFKL94]. Therefore, HB is attractive for low-cost devices. Juels and Weis [JW05] pointed out that HB is insecure against active adversary and proposed HB^+ built upon the HB protocol, a three-round secret-key authentication protocol. [1] Soon after, HB^+ was shown vulnerable to a *man-in-the-middle* (MIM) attack proposed by Gilbert, Robshaw, and Silbert [GRS05]. The line of researches [BCD06,DK07,GRS08b,KPV$^+$17,HKL$^+$12,CKT16] proposed variants of $\mathsf{HB}/\mathsf{HB}^+$ and some of them are secure against MIM attacks.

[1] Later, Katz, Shin, and Smith gave simplified security proofs of them [KSS10]

Their underlying problems are the LPN problem and its variants. Several attacks on the LPN problem have been proposed over the last years [LF06,EKM17]. Most of them are variants of the BKW algorithm [BKW03] whose running time is $2^{\mathcal{O}(\frac{k}{\log k})}$. In addition, [EKM17] introduced an algorithm solving the LPN problem running in the quantum setting. They make the HB family very inefficient in practice either in classical or quantum setting. Moreover, Bernstein and Lange [BL12] discussed the comparison of Lapin [HKL^{+}12] and (light-weight) block-ciphers on RFID tags and smart cards. Armknecht, Hamann, and Mikhalev [AHM14] also discussed the hardware limits of low-cost RFID tags in the range of \$0.05–\$0.10. They concluded that all LPN-based authentication protocols cannot be implemented in the low-cost RFID tags in this range.

Hence, it is desirable to come up with a new proposal for secret-key authentication and MAC that provides provable security with better efficiency in terms of key-size, communication, and rounds, while providing post-quantum security promise.

THE MERSENNE LOW-HAMMING COMBINATION (MERS) PROBLEM AND ITS APPLICATION. In 2017, Aggarwal, Joux, Prakash, and Santha proposed the *Mersenne Low Hamming Combination* (MERS) problem [AJPS18,AJPS17]: Given a Mersenne prime in the form $p = 2^n - 1$ (where n is prime), samples of the $\mathsf{MERS}_{n,h}$ distribution are constructed as $(a, b = as + e)$, where $a \in \mathbb{Z}_p$ is chosen uniformly at random, the secret s and the error e are chosen uniformly at random from the elements in $\mathbb{Z}_p$ of the Hamming weight h. The decisional version of the MERS assumption states that any efficient adversary cannot distinguish the $\mathsf{MERS}_{n,h}$ distribution from the uniform distribution over $\mathbb{Z}_p^2$. Aggarwal et al. proposed a public-key encryption scheme based on the $\mathsf{MERS}_{n,h}$ problem [AJPS18,AJPS17].

Regarding the practical aspect, MERS assumption provides efficiency due to its reliance on Mersenne primes [BKLM11]. The potential benefit of MERS-based scheme is a subject of several ongoing research [AJPS18,AJPS17,Sze17,FN17]. Unfortunately, because of their constraint that $n = \Theta(h^2)$ from the correctness of the key-encapsulation mechanisms, the mechanisms in [AJPS18,AJPS17,Sze17,FN17] set $n = 216091$ or 756839. This impacts the sizes of public key and ciphertext, which are approximately n bits, 26.41 KiB – 100.39 KiB. Thus, the main motivation behind MERS-based authentication scheme and MAC is their potential suitability for lightweight devices such as Radio Frequency Identification (RFID) tags and smart card.

1.2 Our contribution

There are three main contributions in this paper:

– New version of MERS problem: The first contribution of this work is MERS-U, which is the MERS problem assuming that the secret is *uniform*. We formally prove that the MERS-U problem is as hard as the MERS problem is hard as in the case of the LWE problem [ACPS09].

– Two-round authentication with S-MIM security: The second contribution is a two-round authentication protocol secure against sequential man-in-the-middle (S-MIM) attacks with tight reductions to the MERS problem. Our construction need not require $n = \Theta(h^2)$ as in KEMs/PKEs in [AJPS18,AJPS17,Sze17,FN17] and we can set $n = \Theta(h)$, say, $n = 4h$. Thus, we can set $n = 521$ and $h = 128$, and this makes our protocol efficient and compact, say, the communication complexity is at most $3n = 1563$ bits.
– Message Authentication Code (MAC): The third contribution is to construct a MAC scheme that is existentially unforgeable under chosen message attacks (UF-CMA) assuming that the MERS problem is hard. Our MAC improves upon the key size, communication and computation complexity with respect to prior works [KPV+17,DKPW12]. Again, we can set $n = \Theta(h)$ as in the authentication.

Protocol	$\#r$	Assumption	Security	Key Size	Comm.
$\mathsf{Auth}_{\mathsf{wprf}}$ [DKPW12]	3	weak PRF	active	$\lvert\mathbb{K}\rvert + \lvert\mathbb{H}\rvert$	$2\lvert\mathbb{D}\rvert + \lvert\mathbb{F}\rvert$
$\mathsf{Auth}_{\mathsf{Fig2}}$ [LM13]	3	weak PRF	S-MIM	$\lvert\mathbb{K}\rvert + \lvert\mathbb{H}\rvert$	$\lvert\mathbb{D}\rvert + 2\lvert\mathbb{F}\rvert$
$\mathsf{Auth}_{\mathsf{wprf}}$ [CKT16]	2	weak PRF	S-MIM	$2\ell\lvert\mathbb{K}\rvert + \lvert\mathbb{H}\rvert$	$\lvert\mathbb{D}\rvert + \lvert\mathbb{F}\rvert$
Auth [KPV+17]	2	$\mathsf{LPN}_{\ell,\gamma}$	active	2ℓ	$2\ell + (\ell+1)\eta$
Lapin [HKL+12]	2	$\mathsf{Ring\text{-}LPN}_{\ell,\gamma}$	active	2ℓ	3ℓ
$\mathsf{Auth}_{\mathsf{LPN}}$ [CKT16]	2	$\mathsf{LPN}_{\ell,\gamma}$	S-MIM	5ℓ	$(\eta+2)\ell$
$\mathsf{AuthT}_{\mathsf{LPN}}$ [CKT16]	2	$\mathsf{LPN}_{\ell,\gamma}$	S-MIM	$(2\eta+2)\ell$	$2\ell+\eta$
$\mathsf{Auth}_{\mathsf{Field\text{-}LPN}}$ [CKT16]	2	$\mathsf{Field\text{-}LPN}_{\ell,\gamma}$	S-MIM	4ℓ	3ℓ
$\mathsf{Auth}_{\mathsf{s\text{-}mim}}$ [Sect. 6]	2	$\mathsf{MERS}_{\mathsf{n,h}}$	S-MIM	$4n$	$3n$

Table 1. Authentication Protocols based on Weak-PRFs, the LPN-related assumptions, and the MERS assumption. A family of weak PRFs is denoted by $\mathcal{F} := \{F : \mathbb{K} \times \mathbb{D} \to \mathbb{F}\}$. A family of pairwise independent hash functions is denoted by $\mathcal{H} := \{H : \mathbb{H} \times \mathbb{D} \to \mathbb{F}\}$. ℓ and γ defines the dimension and the error rate of the LPN problem. $\eta = O(\ell)$ defines the number of parallel repetitions. n and h are parameters for $\mathsf{MERS}_{n,h}$.

1.3 Related Works

Security notions. Bellare and Rogaway [BR94] gave the formal security definition of *mutual* authentication schemes. Their security model captures MIM attack and more. Vaudeney [Vau07] gave the formal security and privacy definitions of RFID authentications. In this paper, we only consider *unilateral* authentication scheme and do not consider any corruption. Mol and Tessaro [MT12] gave the security definitions for *unilateral* authentication scheme that captures from passive attacks to MIM attacks. Lyubashevky and Masny [LM13] introduced an interesting notion of security against Man-In-the-Middle (MIM) attacks, which slightly weakens MIM to only allow the attacker to interfere with *non-overlapping*

Protocol	Assumption	Security	Key Size	Comm.
MAC_1 [KPV+17]	$\mathsf{LPN}_{\ell,\gamma}$	UF-CMA	$2\ell + \|H\| + \|\pi\|$	$\ell\eta + \eta + \nu$
MAC_2 [KPV+17]	$\mathsf{LPN}_{\ell,\gamma}$	UF-CMA	$(\mu+1)\ell + \eta + \|H\| + \|\pi\|$	$\ell\eta + \eta + \nu$
$\mathsf{MAC}_{\mathsf{MERS}}$ [Sect. 7]	$\mathsf{MERS}_{\mathsf{n,h}}$	UF-CMA	$(\mu+2)n + \|H\| + \|\pi\|$	$2n + \nu$

Table 2. MACs based on the LPN-related assumptions and the MERS assumption. ℓ and γ defines the dimension and the error rate of the LPN problem. $\eta = O(\ell)$ defines the number of parallel repetitions. n and h are parameters for $\mathsf{MERS}_{n,h}$. A family of pairwise independent hash functions is denoted by $\mathcal{H} := \{H : \mathbb{M} \times \{0,1\}^\nu \to \{0,1\}^\mu\}$. A family of pairwise independent permutations is denoted by $\mathcal{P} := \{\pi : \{0,1\}^z \to \{0,1\}^z\}$, where $z = \ell\eta + \eta + \nu$ for LPN case and $z = 2n + \nu$ for MERS case.

sequential sessions. This seems sufficient for real-world application in which the keys do not allow parallel sessions. Cash, Kiltz, and Tessaro [CKT16] also defined Sequential MIM (S-MIM) security. We adopt the following definition of S-MIM security.

Authentication from LPN/LWE. Hopper and Blum [HB01] introduced a secret-key authentication protocol that is proven secure against passive adversaries from the hardness of the LPN problem. Since then, a family of LPN-based authentication protocols has been developed. Juels and Weis [JW05] proposed an efficient three-round variant of HB, called HB^+, which they proved to be secure against active attacks. Later, Gilbert et al. [GRS05] show that HB^+ is not secure against a MIM attack, resulting in several variants [MP07,DK07]. However, most of these variants lack security proofs [GRS08a]. Recent proposals [GRS08b,KPV+17,HKL+12,LM13,CKT16] have proofs for active security or variants of MIM security.

LPN-based protocols have gained some popularity since they require only small number of primitive bit-wise operations (e.g. "XOR" and "AND") for their implementation. However, all LPN-based protocols require huge security parameters. [EKM17] estimates the hardness of $\mathsf{LPN}_{\ell,\tau}$. According to their estimation, for $\tau = 1/8$, $\ell = 670, 1060, 1410$ corresponds to 128, 192, and 256 bit security assuming that the memory is constrained to 2^{80} bits. If we set $\tau = 1/20$ as in [KPV+17], then ℓ should be larger than 1280 for 128-bit security.

Authentication from number-theoretic problems. Concurrently to above, there is another type of protocols based on number-theoretic assumptions, which are DDH-based protocols introduced in [DKPW12,LM13,CKT16]. Unfortunately, same for RSA, the DDH implementation is not suitable for low-cost device. Besides that, factoring and the DDH assumption are known to be threatened by Shor's algorithm that runs by quantum computer [Sho97].

Authentication from weak PRFs. Dodis et al. [DKPW12] show how to construct a three-round authentication from any weak PRFs, which is secure against active attacks. Later, Lyubashevaky and Masny [LM13] constructs a three-round authentication from any weak PRFs with MIM security in sequential sessions.

MAC. Message Authentication Code (MAC) is one of the most fundamental primitive in cryptography, used to authenticate a message. Similarly to secret-key authentication, most of MAC schemes have been based on PRFs. This is achieved either by using secure block ciphers [Pre97] or number-theoretic constructions as shown in [DKPW12,KPV$^+$17]; the latter provides provably (weakly) MIM-secure[2] authentication scheme and MAC based on LPN/LWE and their ring/field variants.

1.4 Organization of the Paper

In Section 2, we review the basic notion and notations, secret-key authentication, and MAC. In Section 3, we review the MERS problem and assumption. In Section 4, we construct a two-round secret-key authentication scheme that is secure against passive adversaries. Next, we build an efficient two-round authentication protocol that has special properties (ROR-CMA security) in Section 5. We then build an efficient two-round authentication protocol secure against S-MIM attacks upon it in Section 6, by applying the transformation of [CKT16]. Finally, we obtain a MAC scheme from the MERS problem in Section 7.

2 Preliminaries

2.1 Notation

We denote by $\|x\|$ the Hamming weight of an n-bit string x, which is the total number of 1's in x. Let $\mathfrak{H}_{n,h}$ be the set of all n-bit strings of Hamming weight h.

Let n be a positive integer and let $p = 2^n - 1$. We call p a Mersenne number if n is prime. If p is itself a prime number then p is called a Mersenne prime.[3]

Let $\mathbb{Z}_p$ be the integer ring modulo p, where p is a Mersenne prime. We have the following properties [AJPS18]: For any $x, y \in \mathbb{Z}_p$, we have

Lemma 2.1. *Let $x, y \in \mathbb{Z}_p$, then the following properties hold:*

Property 1: $\|x + y \pmod p\| \leq \|x\| + \|y\|$

Property 2: $\|x \cdot y \pmod p\| \leq \|x\| \cdot \|y\|$

Property 3: $x \neq 0^n \Rightarrow \|-x \pmod p\| = n - \|x\|$

The proof of this lemma is in [AJPS18].

---

[2] "MIM security" in [DKPW12] is defined by two-phase games. This is $(\{P, V\}, \{V\})$-auth security, while the MIM security is $(\{\}, \{P, V\})$-auth security using [MT12]'s terminology.

[3] For example, n can be 2, 3, 5, 7, 13, 17, 19, 31, 61, 89, 107, 127, 521, 607, 1279, 2203, 2281, 3217, 4253, 4423, 9689, 9941, 11213, 19937, 21701, 23209, 44497, 86243, 110503, 132049, 216091, 756839, 859433, and so on. Mersenne-756839 employed $n = 756839$ and Ramstake employed $n = 216091$ and 756839

2.2 Secret-Key Authentication Syntax

Secret-key authentication protocol $\mathsf{Auth} = (\mathsf{KeyGen}, \mathsf{P}, \mathsf{V})$ is an interactive protocol in which P and V share the same secret key SK (in the context of RFID, we consider P as a *tag* and V as a *reader*). More formally, a secret-key authentication protocol proceeds in two phases:

– **Key-generation algorithm**: The key-generation algorithm $\mathsf{KeyGen}(1^\kappa)$ is executed on the security parameter κ and outputs a secret key SK.
– **Authentication Protocol**: The interactive algorithm between P and V takes as input the shared secret key SK and is executed r rounds. And finally, V outputs either `Accept` or `Reject`.

In this paper, we only consider *two-round random-challenge* secret-key authentication protocols, in which the protocol is run as follows; the verifier chooses a challenge c from the challenge space $\mathcal{C}$ uniformly at random and sends it as the first message; the prover receives c, computes a response $\tau \leftarrow \mathsf{P}_{\mathsf{SK}}(c)$, and sends it as the second message; the verifier receives τ and outputs its decision $d \leftarrow \mathsf{V}_{\mathsf{SK}}(c, \tau)$.

We say that the authentication protocol has *completeness error* α if for all secret keys SK generated by $\mathsf{KeyGen}(1^\kappa)$ the honestly executed protocol returns reject with probability at most α. More formally, for all $1^\kappa \in \mathbb{N}, \mathsf{SK} \leftarrow \mathsf{KeyGen}(1^\kappa)$:

$$\Pr[c \leftarrow_\$ \mathcal{C}; \tau \leftarrow \mathsf{P}_{\mathsf{SK}}(c); d \leftarrow \mathsf{V}_{\mathsf{SK}}(c, \tau) : d = \texttt{Reject}] \le \alpha.$$

2.3 Security Models

As for public-key authentication [FS87], several security notions have been introduced for secret-key authentication. There are three main security models against impersonation attacks that are: *passive*, *active*, and *man-in-the-middle*. All three models proceed in two steps: In the first step, the adversary interacts with P and V and then in the second step, it starts interacting only with V in order to get accepted. The weakest notion, which is the passive security, is when the adversary should not be able to interact with V after eavesdropping several sessions in the authentication protocol between P and V. A stronger notion, which is the active security, is when the adversary should not be able to interact with V after interacting *arbitrarily* with P and eavesdropping passively several sessions in the authentication protocol between P and V.

Finally, the strongest and most realistic security model of adversary is a *man-in-the-middle attack* (MIM), where the adversary, in the first phase, can *arbitrarily* interact with P and V before making verification queries to the reader.

Passive Security. As the basic security notion, we review the definition of passive security for *two-round random-challenge* secret-key authentication protocols.

Definition 2.1 (Passive security). *Let* $\mathsf{Auth} = (\mathsf{KeyGen}, \mathsf{P}, \mathsf{V})$ *be a* two-round random-challenge *secret-key authentication protocol. Define the security game* $\mathsf{Exp}^{\mathsf{pa}}_{\mathsf{Auth},\mathcal{A}}(\kappa)$ *between a challenger and an adversary* $\mathcal{A}$ *as in Figure 1. For any adversary* $\mathcal{A}$*, we define its advantage against* Auth *as the quantity*

$$\mathsf{Adv}^{\mathsf{pa}}_{\mathsf{Auth},\mathcal{A}}(\kappa) := \Pr[\mathsf{Exp}^{\mathsf{pa}}_{\mathsf{Auth},\mathcal{A}}(\kappa) \Rightarrow \mathtt{True}].$$

We say Auth *is* (t, q, ϵ)*-passively-secure if for all* t*-time adversary* $\mathcal{A}$ *querying to* T *at most* q *times, we have* $\mathsf{Adv}^{\mathsf{pa}}_{\mathsf{Auth},\mathcal{A}}(\kappa) \leq \epsilon$.

$\underline{\mathsf{Exp}^{\mathsf{pa}}_{\mathsf{Auth},\mathcal{A}}(\kappa)}$

$\mathsf{SK} \leftarrow_{\$} \mathsf{KeyGen}(1^\kappa)$
$st \leftarrow \mathcal{A}^{T(\cdot)}(1^\kappa)$
$c^* \leftarrow_{\$} \mathcal{C}$
$\tau^* \leftarrow \mathcal{A}(st, c^*)$
return $(\mathsf{V}_{\mathsf{SK}}(c^*, \tau^*) = \mathtt{Accept})$

$\underline{\text{Oracle } T()}$

$c \leftarrow_{\$} \mathcal{C}$
$\tau \leftarrow \mathsf{P}_{\mathsf{SK}}(c)$
return (c, τ)

Fig. 1. Definition of $\mathsf{Exp}^{\mathsf{pa}}_{\mathsf{Auth},\mathcal{A}}(\kappa)$

2.4 Tag Sparsity Definition and Security

In this section we define an important tool that our construction relies on, which is *tag sparsity* [CKT16].

This is the property of an authentication protocol $\mathsf{Auth} = (\mathsf{KeyGen}, \mathsf{P}, \mathsf{V})$ for which the tag τ is composed into two distinct components, which are $\tau_1 \in \mathcal{T}_1$ and $\tau_2 \in \mathcal{T}_2$.

Informally speaking, this notion says that for any challenge c, a secret SK, and a left tag τ_1, the number of right tags τ_2 that makes $\tau = (\tau_1, \tau_2)$ accepted is negligible.

Definition 2.2 (Right Tag-Sparsity [CKT16, Definition 4]). *Let* $\mathsf{Auth} = (\mathsf{KeyGen}, \mathsf{P}, \mathsf{V})$ *be a two-round random-challenge secret-key authentication protocol with tags in* $\mathcal{T}_1 \times \mathcal{T}_2$ *and challenge space* $\mathcal{C}$*. For* $\epsilon = \epsilon(1^\kappa)$*, we say that* Auth *has* ϵ*-sparse right tags (or* Auth *has* ϵ-right tag sparsity*) if*

$$\Pr[\tau_2 \leftarrow_{\$} \mathcal{T}_2; d \leftarrow \mathsf{V}_{\mathsf{SK}}(c, (\tau_1, \tau_2)) : d = \mathtt{Accept}] \leq \epsilon$$

for all $c \in \mathcal{C}, \mathsf{SK}$*, and* $\tau_1 \in \mathcal{T}_1$.

ROR-CMA security. In our construction we are also considering a new property introduced in [CKT16], called *real-or-random right-tag chosen-message security* (ROR-CMA) suitable to tag-sparsity notion. Roughly speaking, the scheme is ROR-CMA-secure if, given a random challenge c^*, any efficient adversary cannot distinguish a real prover from the fake prover that returns the random right tag τ_2 on all challenge except c^* even if it can finally access to the verification oracle on the challenge c^* and τ^* of its choice. The formal statement follows:

Definition 2.3 (ROR-CMA security). *Let* $\mathsf{Auth} = (\mathsf{KeyGen}, \mathsf{P}, \mathsf{V})$ *be a two-round random-challenge secret-key authentication protocol. For* $b \in \{0,1\}$*, we define the security game* $\mathsf{Exp}^{\mathsf{ror\text{-}cma},b}_{\mathsf{Auth},\mathcal{A}}(\kappa)$ *between a challenger and an adversary* $\mathcal{A}$ *as in Figure 2. For any adversary* $\mathcal{A}$*, we define its* ROR-CMA *advantage against* Auth *as the quantity*

$$\mathsf{Adv}^{\mathsf{ror\text{-}cma}}_{\mathsf{Auth},\mathcal{A}}(\kappa) := \left| \Pr[\mathsf{Exp}^{\mathsf{ror\text{-}cma},0}_{\mathsf{Auth},\mathcal{A}}(\kappa) \Rightarrow 1] - \Pr[\mathsf{Exp}^{\mathsf{ror\text{-}cma},1}_{\mathsf{Auth},\mathcal{A}}(\kappa) \Rightarrow 1] \right|.$$

We say Auth *is* (t, q, ϵ)-ROR-CMA*-secure if for all* t*-time adversary* $\mathcal{A}$ *issuing at most* q *queries to the oracle* $T_b(\cdot)$*, we have* $\mathsf{Adv}^{\mathsf{ror\text{-}cma}}_{\mathsf{Auth},\mathcal{A}}(\kappa) \leq \epsilon$.

$\mathsf{Exp}^{\mathsf{ror\text{-}cma},b}_{\mathsf{Auth},\mathcal{A}}(\kappa)$	Oracle $T_b(c)$
$\mathsf{SK} \leftarrow_\$ \mathsf{KeyGen}(1^\kappa)$	$(\tau_1, \tau_2^1) \leftarrow_\$ \mathsf{P}_{\mathsf{SK}}(c); \tau_2^0 \leftarrow_\$ \mathcal{T}_2$
$c^* \leftarrow_\$ \mathcal{C}$	**if** $c = c^*$ **then**
$(\tau^*, state) \leftarrow_\$ \mathcal{A}^{T_b(\cdot)}(1^\kappa, c^*)$	**return** $\tau := (\tau_1, \tau_2^1)$
$d \leftarrow_\$ \mathsf{V}_{\mathsf{SK}}(c^*, \tau^*)$	**else**
return $\mathcal{A}(state, d)$	**return** $\tau := (\tau_1, \tau_2^b)$

Fig. 2. Definition of $\mathsf{Exp}^{\mathsf{ror\text{-}cma},b}_{\mathsf{Auth},\mathcal{A}}(\kappa)$

2.5 Security against Sequential Man-in-the-Middle Adversary

In this paper, we target a weaker notion of the man-in-the-middle security, which is Sequential MIM (S-MIM) security, of [LM13,CKT16]; in which the adversary can first interact *sequentially* with P and V in independent sessions and then makes verification queries to V in order to make the latter accept.

Cash, Kiltz, and Tessaro [CKT16] defined S-MIM security notion for two-round random-challenge secret-key authentication protocols. We invoke the adversary $\mathcal{A}$ who access to three oracles: C, P, and V. To synchronize the sessions, each of these oracles use a variable sid associated to a given session. For every session, $\mathcal{A}$ invokes $C()$ to get a new random challenge c, and then invokes the oracle $P()$

on input c' that runs $\mathsf{P}_{\mathsf{SK}}(c')$ and returns a response τ. Finally, given τ' from $\mathcal{A}$, $V()$ checks whether τ' is a valid response on a session challenge $c[\mathsf{sid}]$ or not, and then increases the session number sid. $\mathcal{A}$ wins if it makes V accepts in some session and has changed at least one of messages in the session sent by P and V.

Definition 2.4 (S-MIM **security [CKT16, Section 2]).** *Let* $\mathsf{Auth} = (\mathsf{KeyGen}, \mathsf{P}, \mathsf{V})$ *be a two-round random-challenge secret-key authentication protocol. Define the security game* $\mathsf{Exp}^{\mathsf{s\text{-}mim}}_{\mathsf{Auth},\mathcal{A}}(\kappa)$ *between a challenger and an adversary* $\mathcal{A}$ *as in Figure 3. For any adversary* $\mathcal{A}$*, we define its* S-MIM *advantage against* Auth *as the quantity*

$$\mathsf{Adv}^{\mathsf{s\text{-}mim}}_{\mathsf{Auth},\mathcal{A}}(\kappa) := \Pr[\mathsf{Exp}^{\mathsf{s\text{-}mim}}_{\mathsf{Auth},\mathcal{A}}(\kappa) \Rightarrow \mathtt{True}].$$

We say Auth *is* (t, q, ϵ)-S-MIM*-secure if for all t-time adversary $\mathcal{A}$ invoking at most q sessions, we have* $\mathsf{Adv}^{\mathsf{s\text{-}mim}}_{\mathsf{Auth},\mathcal{A}}(\kappa) \leq \epsilon$.

$\mathsf{Exp}^{\mathsf{s\text{-}mim}}_{\mathsf{Auth},\mathcal{A}}(\kappa)$

$\mathsf{sid} \leftarrow 0$
$\mathsf{SK} \leftarrow_\$ \mathsf{KeyGen}(1^\kappa)$
run $\mathcal{A}^{C(\cdot),P(\cdot),V(\cdot)}(1^\kappa)$
return $\begin{pmatrix} \exists i \colon (c[i], \tau[i]) \neq (c'[i], \tau'[i]) \\ \wedge d[i] = \mathtt{Accept} \end{pmatrix}$

Oracle $C()$

if $c[\mathsf{sid}] = \perp$ **then**
  $c[\mathsf{sid}] \leftarrow_\$ \mathcal{C}$
return $c[\mathsf{sid}]$

Oracle $P(c')$

if $c'[\mathsf{sid}] = \perp$ **then**
  $c'[\mathsf{sid}] \leftarrow c'$
  $\tau[\mathsf{sid}] \leftarrow_\$ \mathsf{P}_{\mathsf{SK}}(c')$
return $\tau[\mathsf{sid}]$

Oracle $V(\tau')$

$\tau'[\mathsf{sid}] \leftarrow \tau', c \leftarrow_\$ C()$
$d[\mathsf{sid}] \leftarrow_\$ \mathsf{V}_{\mathsf{SK}}(c, \tau'[\mathsf{sid}])$
$\mathsf{sid} \leftarrow \mathsf{sid} + 1$
return $d[\mathsf{sid}]$

Fig. 3. Definition of $\mathsf{Exp}^{\mathsf{s\text{-}mim}}_{\mathsf{Auth},\mathcal{A}}(\kappa)$

Let $\mathsf{Auth}' = (\mathsf{KeyGen}', \mathsf{P}', \mathsf{V}')$ be two-round random-challenge authentication protocol with challenge space $\mathcal{C}$ and split tag space $\mathcal{T} = \mathcal{T}_1 \times \mathcal{T}_2$. We assume that $\mathcal{T}_2 = \mathbb{F}$ is a finite field with addition $+$ and multiplication $\circ$. Let $H := \{H_{K_H} \colon \mathcal{T}_1 \to \mathbb{F}\}$ be a family of pairwise independent hash functions. Cash et al. [CKT16] turn Auth' satisfying ROR-CMA security into $\mathsf{Auth} = (\mathsf{KeyGen}, \mathsf{P}, \mathsf{V})$ as follows:

- **Public parameters**: The same as Auth'.
- **Key generation**: The key-generation algorithm KeyGen picks $K_H \leftarrow_\$ \mathcal{K}_H$, $K_F \leftarrow_\$ \mathbb{F} \setminus \{0\}$, and $K' \leftarrow_\$ \mathsf{KeyGen}'(1^\kappa)$. The key is $K := (K_H, K_F, K')$.

- **Challenge**: The challenge is $c \leftarrow_\$ \mathcal{C}$.
- **Response**: The response is $\sigma = (\sigma_1, \sigma_2)$; the prover first computes $\tau = (\tau_1, \tau_2) \leftarrow_\$ \mathsf{P}'_{K'}(c)$ and
$$\sigma = (\sigma_1, \sigma_2) := \Big(\tau_1, \tau_2 \circ K_F + H_{K_H}(\tau_1)\Big) \in \mathcal{T}_1 \times \mathbb{F}.$$
- **Verification**: Given a challenge c and response $\sigma = (\sigma_1, \sigma_2)$, the verifier first computes
$$\tau = (\tau_1, \tau_2) := \Big(\sigma_1, (\sigma_2 - H_{K_H}(\sigma_1)) \circ K_F^{-1}\Big)$$
and returns the decision $d \leftarrow_\$ \mathsf{V}'_{K'}(c, \tau)$.

Theorem 2.1 ([CKT16, Theorem 5]). *Suppose that H is δ-almost universal and that* Auth' *is* (t, r, ϵ)-ROR-CMA-*secure, satisfies β-right tag sparsity, and has completeness error α. then* Auth *is* $\big(t', r, r \cdot (\epsilon + r/|\mathcal{C}| + \beta\delta|\mathbb{F}|/(|\mathbb{F}| - 1) + r\alpha\big)$-S-MIM-*secure, where $t' \approx t$.*

2.6 Message Authentication Codes

A MAC scheme is a tuple of three probabilistic polynomial-time algorithms $\mathsf{MAC} = (\mathsf{KeyGen}, \mathsf{Tag}, \mathsf{Verify})$ over $(\mathcal{K}, \mathcal{M}, \mathcal{T})$ where $\mathcal{K}$, $\mathcal{M}$, and $\mathcal{T}$ are key space, message space, and tag space, respectively:

- **Key-generation algorithm**: The probabilistic key-generation algorithm KeyGen gives secret key SK on input a security parameter κ.
- **Tag-generation algorithm**: The probabilistic authentication algorithm Tag takes as inputs the secret key SK, the message m and then outputs a tag σ.
- **Verification algorithm**: The deterministic verification algorithm Verify takes as inputs a secret key SK, a message m and a tag σ and outputs either `Accept` or `Reject`.

Completeness We say that MAC has a completeness error α, if for all $m \in \mathcal{M}$ and $1^\kappa \in \mathbb{N}$:

$\Pr[\mathsf{SK} \leftarrow_\$ \mathsf{KeyGen}(1^\kappa); \sigma \leftarrow_\$ \mathsf{Tag}(\mathsf{SK}, m); d \leftarrow \mathsf{Verify}(\mathsf{SK}, m, \sigma) : d = \texttt{Reject}] \leq \alpha$.

We often say that MAC is *perfectly correct* if $\alpha = 0$.

UF-CMA security The standard security notion for MAC scheme is *unforgeability under chosen-message attacks* (UF-CMA), captured by the experiment described in Figure 4.

Definition 2.5. *Let* $\mathsf{MAC} = (\mathsf{KeyGen}, \mathsf{Tag}, \mathsf{Verify})$ *be a MAC scheme. We define the security game* $\mathsf{Exp}^{\text{uf-cma}}_{\mathsf{MAC},\mathcal{A}}(\kappa)$ *between a challenger and an adversary $\mathcal{A}$ as in Figure 4. For any adversary $\mathcal{A}$, we define* UF-CMA *advantage against* MAC *as the quantity*
$$\mathsf{Adv}^{\text{uf-cma}}_{\mathsf{MAC},\mathcal{A}}(\kappa) := \Pr[\mathsf{Exp}^{\text{uf-cma}}_{\mathsf{MAC},\mathcal{A}}(\kappa) \Rightarrow \texttt{True}].$$
We say that a MAC *is* (t, q, ϵ)-UF-CMA-*secure if for all t-time adversary* $\mathsf{Adv}^{\text{uf-cma}}_{\mathsf{MAC},\mathcal{A}}(\kappa)$ *issuing at most q queries to the oracles $T(\cdot)$ and $V(\cdot,\cdot)$, we have* $\mathsf{Adv}^{\text{uf-cma}}_{\mathsf{MAC},\mathcal{A}}(\kappa) \leq \epsilon$.

$\mathsf{Exp}^{\mathsf{uf\text{-}cma},b}_{\mathsf{MAC},\mathcal{A}}(\kappa)$	Oracle $T(m)$
$\mathcal{Q}_T, \mathcal{Q}_V \leftarrow \emptyset$	$\mathcal{Q}_T \leftarrow \mathcal{Q}_T \cup \{m\}$
$\mathsf{SK} \leftarrow_\$ \mathsf{KeyGen}(1^\kappa)$	$\sigma \leftarrow_\$ \mathsf{Tag}(\mathsf{SK}, m)$
run $\mathcal{A}^{T(\cdot),V(\cdot,\cdot)}(1^\kappa)$	**return** σ
return $\begin{pmatrix} \exists (m,\sigma) \in \mathcal{Q}_V \text{ s.t. } m \notin \mathcal{Q}_T \\ \wedge \mathsf{Verify}(\mathsf{SK}, m, \sigma) = \mathtt{Accept} \end{pmatrix}$	Oracle $V(m, \sigma)$
	$\mathcal{Q}_V \leftarrow \mathcal{Q}_V \cup \{(m,\sigma)\}$
	return $\mathsf{Verify}(\mathsf{SK}, m, \sigma)$

Fig. 4. Definition of $\mathsf{Exp}^{\mathsf{uf\text{-}cma}}_{\mathsf{MAC},\mathcal{A}}(\kappa)$

2.7 Hash Functions

Our construction relies on pairwise-independent hash functions and is defined as following:

Definition 2.6 (Pairwise-independent hash functions). *A function $h\colon \mathcal{K} \times \mathcal{N} \to \mathcal{M}$ is called pairwise-independent hash function if for $x_1 \neq x_2 \in \mathcal{N}$, $y_1, y_2 \in \mathcal{M}$,*

$$\Pr_{\mathsf{SK} \leftarrow \mathcal{K}}[h_{\mathsf{SK}}(x_1) = y_1 \wedge h_{\mathsf{SK}}(x_2) = y_2] \leq \frac{1}{|\mathcal{M}|^2}.$$

Concrete Construction. We now consider the following construction of pairwise independent function based on ring of integers modulo prime ($\mathbb{Z}_p$):

Lemma 2.2. *For every $n \in \mathbb{N}$, define: $h : \mathbb{Z}_p^2 \times \mathbb{Z}_p \to \mathbb{Z}_p$ by $h_{a,b}(x) = a \cdot x + b$. Then the function h is pairwise-independent. That is, for all $x_1 \neq x_2$ and $y_1, y_2 \in \mathbb{Z}_p$,*

$$\Pr_{(a,b) \leftarrow \mathbb{Z}_p^2}[h_{a,b}(x_1) = y_1 \wedge h_{a,b}(x_2) = y_2] \leq 1/p^2.$$

The proof can be found in [Rub12]

3 The MERS Problem

Aggarwal et al. introduced new assumptions [AJPS18] mimicking NTRU/Ring-LWE with short secret over integers, relying on the properties of Mersenne primes in the ring $\mathbb{Z}_p$ instead of polynomial ring $\mathbb{Z}_q[x]/(x^n - 1)$. We here employ their latter assumption mimicking Ring-LWE with short secret and extend it to that mimicking Ring-LWE with uniform secret.

For two integers $n > h$ and for n-bit Mersenne prime $p = 2^n - 1$, and for integer $s \in \mathbb{Z}_p$, we define an oracle $\mathcal{O}_{s,n,h}$ as follows: choose $a \leftarrow_\$ \mathbb{Z}_p$ and $e \leftarrow_\$ \mathfrak{H}_{n,h}$ and return $(a, a \cdot s + e \bmod p)$. We also define a uniform oracle $\mathcal{U}$ as follows: choose $(a, b) \leftarrow_{\$} \mathbb{Z}_p^2$ and return it. [4]

Let us define the *Mersenne Low-Hamming Combination Assumption* (the MERS assumption).

Definition 3.1 (MERS problem). *For two positive integers $n > h$ and for an adversary $\mathcal{A}$, we introduce the $\mathsf{MERS}_{n,h}$ advantage as the quantity:*

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{MERS}_{n,h}}(\kappa) := \left|\Pr[\mathcal{A}^{\mathcal{O}_{s,n,h}()} \Rightarrow \mathtt{True}] - Pr[\mathcal{A}^{\mathcal{U}()} \Rightarrow \mathtt{True}]\right|,$$

where $s \leftarrow_{\$} \mathfrak{H}_{n,h}$. We say that the $\mathsf{MERS}_{n,h}$ problem is (t, q, ϵ)-hard if all t-time attacker $\mathcal{A}$ with time complexity t, making at most q queries, we have $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{MERS}_{n,h}}(\kappa) \leq \epsilon$.

The original definition [AJPS18, Definition 5] allows an adversary to query at most twice. We generalize the assumption by allowing polynomially-many queries.

3.1 MERS Problem with Uniform Secret

We next define the $\mathsf{MERS\text{-}U}_{n,h}$ problem with an n-bit Mersenne prime $p = 2^n - 1$ and integer $h \in \{0, \ldots, n\}$.

Definition 3.2 (MERS problem with uniform secret). *For two positive integers $n > h$ and for an adversary $\mathcal{A}$, we define the $\mathsf{MERS\text{-}U}_{n,h}$ advantage as the quantity:*

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{MERS\text{-}U}_{n,h}}(\kappa) := \left|\Pr[\mathcal{A}^{\mathcal{O}_{s,n,h}()} \Rightarrow \mathtt{True}] - \Pr[\mathcal{A}^{\mathcal{U}()} \Rightarrow \mathtt{True}]\right|, \tag{1}$$

where $s \leftarrow_{\$} \mathbb{Z}_p$. We say that the $\mathsf{MERS\text{-}U}_{n,h}$ problem is (t, q, ϵ)-hard if all attacker $\mathcal{A}$ with time complexity t, making at most q queries, we have $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{MERS\text{-}U}_{n,h}}(\kappa) \leq \epsilon$.

It is easy to show that if $\mathsf{MERS}_{n,h}$ is (t', q, ϵ')-hard, then $\mathsf{MERS\text{-}U}_{n,h}$ is also (t, q, ϵ)-hard with $t' \approx t$ and $\epsilon' \approx \epsilon$ (by a simple randomization of the secret s). We note that the converse is also true.

Proposition 3.1. *If the $\mathsf{MERS\text{-}U}_{n,h}$ problem is $(t', q + 1, \epsilon')$-hard, then the $\mathsf{MERS}_{n,h}$ problem is (t, q, ϵ')-hard, where $t' \approx t$ and $\epsilon' \approx \epsilon$.*

Proof. We show a reduction algorithm by following the reduction in [ACPS09, Lemma 2]. Consider the following conversion, which will map $\mathcal{O}_{s,n,h}$ (and $\mathcal{U}$) into $\mathcal{O}_{\bar{e},n,h}$ where $\bar{e} \leftarrow_{\$} \mathfrak{H}_{n,h}$ (and $\mathcal{U}$), respectively: It takes a sample $(\bar{a}, \bar{b})$ with $\bar{a} \neq 0$ from the oracle of $\mathsf{MERS\text{-}U}_{n,h}$. It then converts a sample (a, b) into (a', b'), where $a' := -\bar{a}^{-1} \cdot a$ and $b' := b + a' \cdot \bar{b}$.

[4] In the original definition, a is chosen from $\{0, 1\}^n$. This change introduces only negligible distance

- Suppose that $\bar{b} = \bar{a} \cdot s + \bar{e}$, where $s \leftarrow \mathbb{Z}_p$ and $\bar{e} \leftarrow \mathfrak{H}_{n,h}$. In this case, a' is uniformly distributed since a is uniformly distributed and the map $a \mapsto -\bar{a}^{-1}a$ is one-to-one. Moreover, if $b = as + e$ with $e \in \mathfrak{H}_{n,h}$, then $b' = b + a' \cdot \bar{b} = as + e + a'(\bar{a}s + \bar{e}) = as + e + a'\bar{a}s + a'\bar{e} = a'\bar{e} + e$ since $a'\bar{a} \equiv -a \pmod{p}$. Thus, the converted samples are identified with the samples from $\mathcal{O}_{\bar{e},n,h}$.
- On the other hand, if the oracle is $\mathcal{U}$, then the converted samples are also distributed according to the uniform distribution.

Therefore, the conversion algorithm converts the oracle $\mathcal{O}_{s,n,h}$ (and $\mathcal{U}$) into $\mathcal{O}_{\bar{e},n,h}$ where $\bar{e} \leftarrow_{\$} \mathfrak{H}_{n,h}$ (and $\mathcal{U}$), respectively. This completes the proof. □

3.2 Hardness and Concrete Parameters

Meet-in-the-Middle attack. de Boer et al. [dBDJdW18] presented a meet-in-the-middle attack for solving the MERS problem.Their classical attack runs in the time $\tilde{O}\left(\binom{n-1}{h-1}^{1/2}\right)$. The quantum version runs in the time $\tilde{O}\left(\binom{n-1}{h-1}^{1/3}\right)$. They correspond to roughly $\frac{1}{4}h \lg n$ and $\frac{1}{6}h \lg n$ bits security, respectively.
LLL-attack. The authors of [BCGN17,dBDJdW18] presented an LLL-based algorithm for solving the ratio version of MERS assumption[5] and the MERS problem used in the present paper. For small $h = O(\sqrt{n})$, the running time of the LLL attack is $O(2^{2h})$ on Turing machine and $O(2^h)$ on quantum machine.

Coron and Gini [CG19] also gave an LLL-based attack to solve the MERS problem. The (expected) running time of their attack is $O(2^{1.75h})$.

Tiepelt and Szepieniec [TS19] analyzed an quantum-LLL algorithm and applied it to the MERS problem.

As claimed in [AJPS18], attacks against MERS cannot exceed the complexity of the order 2^h where h is the hamming weight parameter.

Assuming that, when considering the security and implementation of our protocols, one should choose the parameter h at least half of the desired security level κ.

Primality of n in Mersenne primes. Agrawall discussed that $p = 2^n - 1$ and n should be primes to avoid an attack on composite n. For the details, see Aggrawal et al. [AJPS18].

Parameters. Assuming the attacks and constraints above, we choose parameter values as $(\kappa, h, n) = (256, 128, 521)$. It will serve classical 256-bit sec. and quantum 192-bit sec.

4 Passively-Secure Authentication Based on MERS

In this section we introduce our new two-round authentication protocol based on $\mathsf{MERS}_{n,h}$ problem with passive security. Our $\mathsf{Auth}_{\mathsf{pa}}$ is defined as follows:

[5] The Mersenne Low Hamming Ratio Assumption states that, given an n-bit Mersenne prime $p = 2^n - 1$ and an integer h, any PPT adversary cannot distinguish between $F/G \bmod p$ with $F, G \leftarrow_{\$} \mathfrak{H}_{n,h}$, and $R \leftarrow \mathbb{Z}_p$ with non-negligible advantage.

$\mathsf{Auth} : \mathsf{SK} = S \leftarrow_\$ \mathbb{Z}_p$

Prover | **Verifier**

Verifier: $A \leftarrow_\$ \mathbb{Z}_p$

$\xleftarrow{\quad A \quad}$

Prover: $E \leftarrow_\$ \mathfrak{H}_{n,h}$

$B \leftarrow AS + E$

$\xrightarrow{\quad B \quad}$

Verifier: if $\|B - AS\| = h$, then `Accept`

Fig. 5. Passively-secure authentication protocol $\mathsf{Auth}_{\mathsf{pa}}$

- **Public parameters**: The authentication protocol has the following public parameters that depend on the security parameter κ.
  - $n \in \mathbb{N}$: the length of A, S, and E
  - $h \in \mathbb{N}$: the Hamming weight of E
- **Key generation**: The key-generation algorithm $\mathsf{KeyGen}(1^\kappa)$ outputs $\mathsf{SK} = S \leftarrow_\$ \mathbb{Z}_p$.
- **Authentication protocol**: To be authenticated by verifier, a prover follows the two-round authentication protocol shown on Figure 5.

Theorem 4.1. *If the* $\mathsf{MERS\text{-}U}_{n,h}$ *problem is* (t, q, ϵ)*-hard and* $\frac{1}{p}\sum_{i=0}^{2h}\binom{n}{i}$ *is negligible in* κ*, then* $\mathsf{Auth}_{\mathsf{pa}}$ *is passively-secure authentication.*

The proof results straightforwardly from the $\mathsf{MERS\text{-}U}_{n,h}$ assumption:
We have

$$\Pr[\mathsf{Exp}^{\mathsf{pa}}_{\mathsf{Auth},\mathcal{A}}(\kappa) \Rightarrow \mathtt{True}] \le \epsilon + \frac{1}{p}\sum_{i=0}^{2h}\binom{n}{i}.$$

The security proof is obtained by following the proof of [KSS10, Theorem 2].

Proof. Let $\mathcal{A}$ be an adversary against passive security of $\mathsf{Auth}_{\mathsf{pa}}$. Let us consider the following reduction algorithm $\mathcal{B}$ solving $\mathsf{MERS\text{-}U}_{n,h}$ by using $\mathcal{A}$: In the learning phase, $\mathcal{B}$ sends a sample (a, b) from its oracle as a transcript (A, B). In the impersonating phase, $\mathcal{B}$ gets a sample $(\bar{a}, \bar{b})$ from its oracle, sends $A := -\bar{a}$ to $\mathcal{A}$, and receives B from $\mathcal{A}$. It outputs 1 if $\|\bar{b} + B\| \le 2h$ and 0 otherwise.

If $\mathcal{B}$'s oracle is $\mathcal{U}$, then $\mathcal{B}$ outputs 1 with probability exactly $\frac{1}{p} \cdot \sum_{i=0}^{2h}\binom{n}{i}$, since $\bar{b}$ is uniformly distributed and independent of everything else.

Next, suppose that $\mathcal{B}$'s oracle is $\mathcal{O}_{s,n,h}$. In this case, the simulation of the learning phase is perfect, where the secret key is $S = s$. Therefore, the event that $\|B - A \cdot S\| = h$ holds with probability is exactly $\Pr[\mathsf{Exp}^{\mathsf{pa}}_{\mathsf{Auth},\mathcal{A}}(\kappa) \Rightarrow \mathtt{True}]$.

We note that if $\|B - A \cdot S\| = h$ holds, then $\|B + \bar{a} \cdot s\| = h$ also holds. Meanwhile, $\|\bar{b} - \bar{a}s\| = h$ since the oracle is $\mathcal{O}_{s,n,h}$. Thus, with probability at least $\Pr[\mathsf{Exp}^{\mathsf{pa}}_{\mathsf{Auth},\mathcal{A}}(\kappa) \Rightarrow \mathtt{True}]$, $\|\bar{b}+B\| = \|\bar{b}-\bar{a}s+\bar{a}s+B\| \leq \|\bar{b}-\bar{a}s\|+\|\bar{a}s+B\| = 2h$ holds.

Therefore, we have

$$\Pr[s \leftarrow \mathbb{Z}_p : \mathcal{D}^{\mathcal{O}_{s,n,h}}() = 1] - \Pr[\mathcal{D}^{\mathcal{U}}() = 1] \geq \Pr[\mathsf{Exp}^{\mathsf{pa}}_{\mathsf{Auth},\mathcal{A}}(\kappa) \Rightarrow \mathtt{True}] - \frac{1}{p}\sum_{i=0}^{2h}\binom{n}{i}$$

and this yields the theorem as we wanted. □

Active attack against $\mathsf{Auth}_{\mathsf{pa}}$. The active attack against $\mathsf{Auth}_{\mathsf{pa}}$ based on $\mathsf{MERS}_{n,h}$ is quite similar to the active attack against HB^+ [GRS05]. It consists for an arbitrary fixed A, the adversarial verifier can send fixed A repeatedly and obtain

$$B_1 \equiv AS + E_1 \pmod{p}, \ldots, B_k \equiv AS + E_k \pmod{p},$$

where E_i's Hamming weight is at most h. If $h < n/2$, the adversary can determine AS's bits from LSB to MSB as follows: (1) taking the majority of LSB of B_i, which is AS's LSB, (2) taking the majority of 2-th bits of $B_i -$ LSB of AS, which is AS's 2-th bit, and so on. It then learns $AS \bmod p$ and obtains S by computing A^{-1}.

5 ROR-CMA-Secure Authentication Based on MERS

Our $\mathsf{Auth}_{\mathsf{ror}}$ is defined as follows:

- **Public parameters**: n and h as in section 4.
- **Key generation**: The key-generation algorithm $\mathsf{KeyGen}_{\mathsf{ror}}(1^\kappa)$ outputs $\mathsf{SK} = (S_1, S_2) \leftarrow_\$ \mathbb{Z}_p^2$.
- **Authentication protocol**: To be authenticated by V, P follows the 2-round authentication protocol shown on Figure 6.

Theorem 5.1. $\mathsf{Auth}_{\mathsf{ror}}$ *has* $\binom{n}{h}/p$*-sparse right tags.*

Proof. For any secret (S_1, S_2), challenge A, and left tag $R \neq 0$, we have $\Pr[\mathsf{V}_{(S_1,S_2)}(A, (R, B)) \Rightarrow \mathtt{Accept} : B \leftarrow_\$ \mathbb{Z}_p] = |\mathfrak{H}_{n,h}|/p = \binom{n}{h}/p$. □

Theorem 5.2. *If the* $\mathsf{MERS\text{-}U}_{n,h}$ *problem is* (t, q, ϵ)*-hard, then* $\mathsf{Auth}_{\mathsf{ror}}$ *is* (t', q, ϵ)*-* $\mathsf{ROR\text{-}CMA}$*-secure, where* $t' \approx t$*.*

Proof (Proof of Theorem 5.2). We follow the proof of the $\mathsf{ROR\text{-}CMA}$ security of the LPN-based authentication scheme in Cash, Kiltz, and Tessaro [CKT16, Theorem 7].

The security of the MERS-based $\mathsf{Auth}_{\mathsf{ror}}$ essentially builds on the ROR-CMA notion. Let us consider an adversary $\mathcal{A}$ who plays the security game $\mathsf{Exp}^{\mathsf{ror\text{-}cma},b}_{\mathsf{Auth}_{\mathsf{ror}},\mathcal{A}}(\kappa)$.

$\mathsf{Auth}_{\mathsf{ror}} : \mathsf{SK} = (S_1, S_2) \leftarrow_\$ \mathbb{Z}_p^2$

Prover		**Verifier**
		$A \leftarrow_\$ \mathbb{Z}_p$
	$\xleftarrow{A}$	
$R \leftarrow_\$ \mathbb{Z}_p, E \leftarrow_\$ \mathfrak{H}_{n,h}$ $B \leftarrow R(S_1 A + S_2) + E$		
	$\xrightarrow{R, B}$	
		if $R \neq 0$. `Accept` if$\|\|(B - R(S_1 A + S_2)\|\| = h$

Fig. 6. ROR-CMA-secure authentication protocol $\mathsf{Auth}_{\mathsf{ror}}$

$\mathcal{B}^{\mathsf{oracle}}()$	Procedure $\bar{T}(c)$

$S_2' \leftarrow_\$ \mathbb{Z}_p$
$A^* \leftarrow_\$ \mathbb{Z}_p$
$(\tau^*, state) \leftarrow \mathcal{A}^{\bar{T}(\cdot)}(1^\kappa, A^*)$
Parse $\tau^* = (R^*, B^*)$
$d \leftarrow (\|B^* - R^* \cdot S_2'\|?= h)$
return $\mathcal{A}(state, d)$
	if $A = A^*$ **then**
	  $R \leftarrow_\$ \mathbb{Z}_p$
	  $\tilde{B} \leftarrow_\$ \mathfrak{H}_{n,h}$
	else
	  $(\tilde{R}, \tilde{B}) \leftarrow \mathsf{oracle}$
	  $R \leftarrow \tilde{R} \cdot (A - A^*)^{-1}$
	$B \leftarrow \tilde{B} + R \cdot S_2'$
	return $\tau = (R, B)$

Fig. 7. Definition of $\mathcal{B}$

We build an adversary $\mathcal{B}$ who solves the $\mathsf{MERS}_{n,h}$ problem, where n and h are known, by using $\mathcal{A}$ as in Figure 7.

Assume that S_1 is the secret of the $\mathsf{MERS\text{-}U}_{n,h}$ problem. $\mathcal{B}$ chooses $S_2' \leftarrow_\$ \mathbb{Z}_p$ and $A^* \leftarrow_\$ \mathbb{Z}_p$. It implicitly defines $S_2 := -A^* \cdot S_1 + S_2' \bmod p$. Since S_2' is uniform over $\mathbb{Z}_p$, S_2 is also. In addition, we have

$$B^* - R^* \cdot (S_1 \cdot A^* + S_2) \equiv B^* - R^* \cdot S_2' \pmod{p}.$$

Thus, the decision by $\mathcal{B}$ is always correct.

We assume that oracle returns $(\tilde{R}, \tilde{B} = \tilde{R}S_1 + E)$, where $E \leftarrow_\$ \mathfrak{H}_{n,h}$ or $\mathbb{Z}_p$.

Let us consider $\bar{T}(\cdot)$, the simulation of $T(\cdot)$. If $A = A^*$, then the simulation is perfect, since $S_2' = S_1 A^* + S_2 \bmod p$ and $B = R \cdot S_2' + \tilde{B}$ where $\tilde{B} \leftarrow_\$ \mathfrak{H}_{n,h}$. Otherwise, that is, if $A \neq A^*$, we have

$$\begin{aligned} B &= \tilde{B} + R \cdot S_2' \\ &= \tilde{R}S_1 + E + R \cdot S_2' \\ &= R \cdot (A - A^*)S_1 + E + R \cdot S_2' \\ &= R \cdot (AS_1 - A^*S_1 + S_2') + E \\ &= R \cdot (AS_1 + S_2) + E, \end{aligned}$$

where E is chosen from $\mathfrak{H}_{n,h}$ or $\mathbb{Z}_p$ uniformly at random.

If E is chosen from $\mathfrak{H}_{n,h}$, then (R, B) is distributed as a response computed by the honest prover with secret key (S_1, S_2). On the other hand, if E is chosen from $\mathbb{Z}_p$, then (R, B) is uniformly distributed over $\mathbb{Z}_p^2$. Therefore, $\mathcal{B}$'s simulations are perfect in both cases. This completes the proof. □

S-MIM ATTACK AGAINST $\mathsf{Auth}_{\mathsf{ror}}$. Flip B's two bits. With probability $\approx 1/h(n-h)$, it will modify E while keeping its Hamming weight.

6 S-MIM-Secure Authentication Based on MERS

Now we turn our ROR-CMA-secure protocol into a S-MIM-secure protocol by using the transformation described in Section 2.5 by using the pairwise independent hash function in Section 2.7.

We set $\mathbb{F} := \mathbb{Z}_p$ and employ the family of pairwise independent hash functions $\{H_{K_1,K_2} \colon \mathbb{Z}_p \to \mathbb{Z}_p \mid K_1, K_2 \in \mathbb{Z}_p\}$, where $H_{K_1,K_2}(R) = K_1 \cdot R + K_2$. Applying the transformation, the key consists of $K = (S_1, S_2, K_F, K_1, K_2)$. The response to a challenge c is computed as $\sigma = (\sigma_1, \sigma_2)$, where

$$\sigma_1 = R \text{ and } \sigma_2 = \underbrace{(R \cdot (S_1 \cdot A + S_2) + E)}_{=\tau_2} \cdot K_F + \underbrace{K_1 \cdot R + K_2}_{=H_{K_H}(\tau_1)}.$$

We can apply the compression technique in [CKT16]. Prover sends $\sigma = (R, Z)$, where

$$\begin{aligned} Z &= (R \cdot (S_1 \cdot A + S_2) + E) \cdot K_F + (K_1 \cdot R + K_2) \\ &= R(S_1 K_F \cdot A + S_2 K_F + K_1) + K_F \cdot E + K_2 \\ &= R(X_1 \cdot A + X_2) + X_3 \cdot E + X_4, \end{aligned}$$

$\mathsf{Auth}_{\mathsf{s\text{-}mim}} : \mathsf{SK} = (X_1, X_2, X_3, X_4) \leftarrow_\$ \mathbb{Z}_p^* \times \mathbb{Z}_p \times \mathbb{Z}_p^* \times \mathbb{Z}_p$

Prover		**Verifier**
		$A \leftarrow_\$ \mathbb{Z}_p$
	$\xleftarrow{\quad A \quad}$	
$R \leftarrow_\$ \mathbb{Z}_p, E \leftarrow_\$ \mathfrak{H}_{n,h}$ $Z \leftarrow R(X_1A + X_2) + X_3E + X_4$		
	$\xrightarrow{\quad R, Z \quad}$	
		if $R \neq 0$. `Accept` if $\|\|(Z - R(X_1A + X_2) - X_4) \cdot X_3^{-1}\|\| = h$

Fig. 8. S-MIM-secure authentication protocol $\mathsf{Auth}_{\mathsf{s\text{-}mim}}$

by substituting $X_1 = S_1K_F$, $X_2 = S_2K_F + K_1$, $X_3 = K_F$, and $X_4 = K_2$. The verifier also checks if

$$R \neq 0 \wedge \|(Z - R(X_1A + X_2) - X_4) \cdot X_3^{-1}\| = h$$

or not. (We can choose them as $X_1 \leftarrow_\$ \mathbb{Z}_p^*$, $X_2 \leftarrow \mathbb{Z}_p$, $X_3 \leftarrow \mathbb{Z}_p^*$, and $X_4 \leftarrow \mathbb{Z}_p$.)

The compressed authentication systems, denoted by $\mathsf{Auth}_{\mathsf{s\text{-}mim}}$, is summarized as follows:

- **Public parameters**: n and h as in section 4.
- **Key generation**: The key-generation algorithm KeyGen outputs $\mathsf{SK} = (X_1, X_2, X_3, X_4) \leftarrow_\$ \mathbb{Z}_p^* \times \mathbb{Z}_p \times \mathbb{Z}_p^* \times \mathbb{Z}_p$.
- **Challenge**: The challenge is $A \leftarrow_\$ \mathbb{Z}_p$.
- **Response**: The response is $\sigma = (R, Z)$ with $Z := R \cdot (X_1A + X_2) + X_3E + X_4$, where $R \leftarrow_\$ \mathbb{Z}_p$ and $E \leftarrow_\$ \mathfrak{H}_{n,h}$.
- **Verification**: Given a challenge A and response $\sigma = (R, Z)$, the verifier accepts if and only if $R \neq 0$ and $\|(Z - R(X_1A + X_2) - X_4) \cdot X_3^{-1}\| = h$.

Combining Theorem 5.1, Theorem 5.2, and Theorem 2.1, we get the following corollary.

Corollary 6.1. *If* $\mathsf{MERS\text{-}U}_{n,h}$ *is* (t, q, ϵ)*-hard, then* $\mathsf{Auth}_{\mathsf{s\text{-}mim}}$ *is* (t', q, ϵ')*-S-MIM-secure, where* $t' \approx t$ *and* $\epsilon' = q \cdot \left(\epsilon + q/p + \binom{n}{h}/(p-1)\right)$.

7 MAC from MERS

In this section, we introduce MAC based on MERS-U. Our construction is an analogue to that in [KPV+17]. The scheme MAC = (KeyGen, Tag, Verify) is summarized as follows:

– **Public parameters**: The public parameters $\mathsf{p}(1^\kappa)$ on the security parameter κ, outputs the public parameters n and h as in section 4.
– **Key generation**: The algorithm KeyGen, given public parameters p, samples $s_0', s_0, s_1, \ldots, s_\mu \leftarrow_\$ \mathbb{Z}_p$, $\mathsf{h}\colon \{0,1\}^* \times \{0,1\}^\nu \to \{0,1\}^\mu$, and pairwise-independent permutation π over $\mathbb{Z}_p \times \mathbb{Z}_p \times \{0,1\}^\nu$, and outputs $\mathsf{SK} := (s_0', s_0, s_1, \ldots, s_\mu, \mathsf{h}, \pi)$.
– **Tagging**: The algorithm Tag, given a secret key SK and a message $m \in \mathcal{M}$. This probabilistic authentication algorithm proceeds as follows:
  - Samples $R \leftarrow_\$ \mathbb{Z}_p$, $E \leftarrow_\$ \mathfrak{H}_{n,h}$ and $\beta \leftarrow_\$ \{0,1\}^\nu$.
  - Compute $A := \mathsf{h}(m, \beta)$.
  - Compute $S_A = s_0 + \sum_{i=1}^{\mu} A[i] \cdot s_i$.
  - Compute $B := R \cdot S_A + E + s_0'$.
  - Output $\sigma = \pi(R, B, \beta)$.
– **Verification**: The algorithm Verify proceeds as follows:
  - Parse $\pi^{-1}(\sigma)$ as (R, B, β). If $R = 0$, then `Reject`.
  - Compute $A := \mathsf{h}(m, \beta)$ and $S_A := s_0 + \sum_{i=1}^{\mu} A[i] \cdot s_i$.
  - If $\|B - (R \cdot S_A + s_0')\| = h$ then return `Accept`, otherwise `Reject`.

Our scheme is perfectly correct.

In what follows, we let $\alpha_{n,h} := \binom{n}{h}/p$.

Theorem 7.1. *If the* $\mathsf{MERS\text{-}U}_{n,h}$ *problem is* (t, Q, ϵ)*-hard, then* MAC *is* (t', Q, ϵ')*-* $\mathsf{UF\text{-}CMA}$*-secure, where* $t \approx t'$ *and*

$$\epsilon = \min\left\{\epsilon'/2 - Q^2/2^\mu, \epsilon'/(8\mu Q_{\mathsf{Verify}}) - Q_{\mathsf{Verify}}\alpha_{n,h}\right\},$$

where $Q_{\mathsf{Verify}} \leq Q$ *is the number of verification queries.*

We obtain our main theorem by combining two lemmas Lemma A.3 and Lemma A.2.

References

ACPS09. Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 595–618. Springer, Heidelberg, August 2009. 1.2, 3.1

AHM14. Frederik Armknecht, Matthias Hamann, and Vasily Mikhalev. Lightweight authentication protocols on ultra-constrained RFIDs – myths and facts. In Nitesh Saxena and Ahmad-Reza Sadeghi, editors, *Radio Frequency Identification: Security and Privacy Issues - 10th International Workshop, RFIDSec 2014*, volume 8651 of *LNCS*, pages 1–18, Oxford, UK, July 21–23 2014. Springer, Heidelberg. 1.1

AJPS17. Divesh Aggarwal, Antoine Joux, Anupam Prakash, and Mikos Santha. Mersenne-756839. Technical report, National Institute of Standards and Technology, 2017. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions`. 1.1, 1.2

AJPS18. Divesh Aggarwal, Antoine Joux, Anupam Prakash, and Miklos Santha. A new public-key cryptosystem via mersenne numbers. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part III*, volume 10993 of *LNCS*, pages 459–482. Springer, Heidelberg, August 2018. 1.1, 1.2, 2.1, 2.1, 3, 3, 3.2

BCD06. Julien Bringer, Hervé Chabanne, and Emmanuelle Dottax. HB^+: A lightweight authentication protocol secure against some attacks. In *Proceedings of the Second International Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing*, SECPERU '06, pages 28–33, Washington, DC, USA, 2006. IEEE Computer Society. 1.1

BCGN17. Marc Beunardeau, Aisling Connolly, Rémi Géraud, and David Naccache. On the hardness of the mersenne low hamming ratio assumption. In Tanja Lange and Orr Dunkelman, editors, *LATINCRYPT 2017*, volume 11368 of *LNCS*, pages 166–174. Springer, Heidelberg, September 2017. 3.2

BFKL94. Avrim Blum, Merrick L. Furst, Michael J. Kearns, and Richard J. Lipton. Cryptographic primitives based on hard learning problems. In Douglas R. Stinson, editor, *CRYPTO'93*, volume 773 of *LNCS*, pages 278–291. Springer, Heidelberg, August 1994. 1.1

BKLM11. Joppe W. Bos, Thorsten Kleinjung, Arjen K. Lenstra, and Peter L. Montgomery. Efficient SIMD arithmetic modulo a Mersenne number. In *Proceedings of the 2011 IEEE 20th Symposium on Computer Arithmetic*, ARITH '11, pages 213–221, Washington, DC, USA, 2011. IEEE Computer Society. 1.1

BKW03. Avrim Blum, Adam Kalai, and Hal Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *J. ACM*, 50(4):506–519, July 2003. 1.1

BL12. Daniel J. Bernstein and Tanja Lange. Never trust a bunny. In Jaap-Henk Hoepman and Ingrid Verbauwhede, editors, *Radio Frequency Identification. Security and Privacy Issues - 8th International Workshop, RFIDSec 2012*, volume 7739 of *LNCS*, pages 137–148, Nijmegen, The Netherlands, July 2–3 2012. Springer, Heidelberg. 1.1

BR94. Mihir Bellare and Phillip Rogaway. Entity authentication and key distribution. In Douglas R. Stinson, editor, *CRYPTO'93*, volume 773 of *LNCS*, pages 232–249. Springer, Heidelberg, August 1994. 1.3

CG19. Jean-Sebastien Coron and Agnese Gini. Improved cryptanalysis of the AJPS Mersenne based cryptosystem. In *Number-Theoretic Methods in Cryptology 2019 – NutMiC 2019*, 2019. Available at `https://eprint.iacr.org/2019/610`. 3.2

CKT16. David Cash, Eike Kiltz, and Stefano Tessaro. Two-round man-in-the-middle security from LPN. In Eyal Kushilevitz and Tal Malkin, editors, *TCC 2016-A, Part I*, volume 9562 of *LNCS*, pages 225–248. Springer, Heidelberg, January 2016. 1.1, 1.2, 1.3, 1.4, 2.4, 2.2, 2.4, 2.5, 2.4, 2.5, 2.1, 5, 6

dBDJdW18. Koen de Boer, Léo Ducas, Stacey Jeffery, and Ronald de Wolf. Attacks on the AJPS mersenne-based cryptosystem. In Tanja Lange and Rainer Steinwandt, editors, *Post-Quantum Cryptography - 9th International Conference, PQCrypto 2018*, pages 101–120. Springer, Heidelberg, 2018. 3.2

DK07. Dang Nguyen Duc and Kwangjo Kim. Securing HB^+ against GRS man-in-the-middle attack. In *SCIS 2007, The 2007 Symposium on Cryptography and Information Security*, pages 2B3–4, Sasebo, Japan, January 23–26 2007. IEICE. 1.1, 1.3

DKPW12. Yevgeniy Dodis, Eike Kiltz, Krzysztof Pietrzak, and Daniel Wichs. Message authentication, revisited. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 355–374. Springer, Heidelberg, April 2012. 1.2, 1.3, 2

EKM17. Andre Esser, Robert Kübler, and Alexander May. LPN decoded. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017, Part II*, volume 10402 of *LNCS*, pages 486–514. Springer, Heidelberg, August 2017. 1.1, 1.3

FN17. Houda Ferradi and David Naccache. Integer reconstruction public-key encryption. Cryptology ePrint Archive, Report 2017/1231, 2017. `https://eprint.iacr.org/2017/1231`. 1.1, 1.2

FS87. Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In Andrew M. Odlyzko, editor, *CRYPTO'86*, volume 263 of *LNCS*, pages 186–194. Springer, Heidelberg, August 1987. 2.3

GRS05. Henri Gilbert, Matthew J. B. Robshaw, and Hervé Sibert. Active attack against HB+: a provably secure lightweight authentication protocol. *Electronics Letters*, 41(21):1169–1170, Oct 2005. See also `https://eprint.iacr.org/2005/237`. 1.1, 1.3, 4

GRS08a. Henri Gilbert, Matthew J. B. Robshaw, and Yannick Seurin. Good variants of HB+ are hard to find. In Gene Tsudik, editor, *FC 2008*, volume 5143 of *LNCS*, pages 156–170. Springer, Heidelberg, January 2008. 1.3

GRS08b. Henri Gilbert, Matthew J. B. Robshaw, and Yannick Seurin. $HB^\sharp$: Increasing the security and efficiency of HB^+. In Nigel P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 361–378. Springer, Heidelberg, April 2008. 1.1, 1.3

HB01. Nicholas J. Hopper and Manuel Blum. Secure human identification protocols. In Colin Boyd, editor, *ASIACRYPT 2001*, volume 2248 of *LNCS*, pages 52–66. Springer, Heidelberg, December 2001. 1.1, 1.3

HKL+12. Stefan Heyse, Eike Kiltz, Vadim Lyubashevsky, Christof Paar, and Krzysztof Pietrzak. Lapin: An efficient authentication protocol based on ring-LPN. In Anne Canteaut, editor, *FSE 2012*, volume 7549 of *LNCS*, pages 346–365. Springer, Heidelberg, March 2012. 1.1, 1.2, 1.3

JW05. Ari Juels and Stephen A. Weis. Authenticating pervasive devices with human protocols. In Victor Shoup, editor, *CRYPTO 2005*, volume 3621 of *LNCS*, pages 293–308. Springer, Heidelberg, August 2005. 1.1, 1.3

KPC+11. Eike Kiltz, Krzysztof Pietrzak, David Cash, Abhishek Jain, and Daniele Venturi. Efficient authentication from hard learning problems. In Kenneth G. Paterson, editor, *EUROCRYPT 2011*, volume 6632 of *LNCS*, pages 7–26. Springer, Heidelberg, May 2011. A.2

KPV+17. Eike Kiltz, Krzysztof Pietrzak, Daniele Venturi, David Cash, and Abhishek Jain. Efficient authentication from hard learning problems. *Journal of Cryptology*, 30(4):1238–1275, October 2017. 1.1, 1.2, 1.2, 1.3, 7, A.1, A.2, B, B

KSS10. Jonathan Katz, Ji Sun Shin, and Adam Smith. Parallel and concurrent security of the HB and HB+ protocols. *Journal of Cryptology*, 23(3):402–421, July 2010. 1, 4

LF06. Éric Levieil and Pierre-Alain Fouque. An improved LPN algorithm. In Roberto De Prisco and Moti Yung, editors, *SCN 06*, volume 4116 of *LNCS*, pages 348–359. Springer, Heidelberg, September 2006. 1.1

LM13. Vadim Lyubashevsky and Daniel Masny. Man-in-the-middle secure authentication schemes from LPN and weak PRFs. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 308–325. Springer, Heidelberg, August 2013. 1.2, 1.3, 2.5

MP07. J. Munilla and A. Peinado. HB-MP: A further step in the HB-family of lightweight authentication protocols. *Comput. Netw.*, 51(9):2262–2267, June 2007. 1.3

MT12. Petros Mol and Stefano Tessaro. Secret-key authentication beyond the challenge-response paradigm: Definitional issues and new protocols. unpublished manuscripts, 2012. Available at `https://homes.cs.washington.edu/~tessaro/`. 1.3, 2

Pre97. Bart Preneel. Hash functions and MAC algorithms based on block ciphers. In Michael Darnell, editor, *6th IMA International Conference on Cryptography and Coding*, volume 1355 of *LNCS*, pages 270–282. Springer, Heidelberg, December 1997. 1.3

Rub12. Ronitt Rubinfeld. Randomness and computation. Course, MIT, 2012. 2.7

Sho97. Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, 26(5):1484–1509, October 1997. 1.3

Sze17. Alan Szepieniec. Ramstake. Technical report, National Institute of Standards and Technology, 2017. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions`. 1.1, 1.2

TS19. Marcel Tiepelt and Alan Szepieniec. Quantum LLL with an application to mersenne number cryptosystems. In Peter Schwabe and Nicolas Thériault, editors, *LATINCRYPT 2019*, volume 11774 of *LNCS*, pages 3–23. Springer, Heidelberg, 2019. 3.2

Vau07. Serge Vaudenay. On privacy models for RFID. In Kaoru Kurosawa, editor, *ASIACRYPT 2007*, volume 4833 of *LNCS*, pages 68–87. Springer, Heidelberg, December 2007. 1.3

A Proof of Theorem 7.1

In what follows, we say a forgery (m, σ) is *fresh* if the A contained in (m, σ) is different from all A's contained in all the previous queries to V and T. For our proof, we are distinguishing two cases: the case where the probability that A is fresh is sufficiently low as $\Pr[\texttt{Fresh}] \leq \epsilon'/2$, or the complement case where $\Pr[\texttt{Fresh}] > \epsilon'/2$.

$\underline{\mathsf{Real}_{\mathcal{B}}(\kappa), \mathsf{Rand}_{\mathcal{B}}(\kappa)}$

$L := \emptyset$
$s_0', s_0, s_1, \ldots, s_\mu \leftarrow_\$ \mathbb{Z}_p$
$d \leftarrow \mathcal{B}^{\mathsf{Eval}(\cdot), \mathsf{Chal}(\cdot,\cdot)}(1^\kappa)$
return $d \wedge (A^* \notin L)$

$\underline{\text{Oracle } \mathsf{Chal}(R^*, A^*) \text{ // one query}}$

$S_{A^*} := s_0 + \sum_{j=1}^{\mu} A^*[j] \cdot s_j$
$B^* := s_0' + R^* \cdot S_{A^*}$
return B^*

$\underline{\text{Oracle } \mathsf{Eval}(A)}$

if $A \in L$ **then**
  return $\perp$
$L \leftarrow L \cup \{A\}$
$S_A := s_0 + \sum_{j=1}^{\mu} A[j] \cdot s_j$
$R \leftarrow_\$ \mathbb{Z}_p; E \leftarrow_\$ \mathfrak{H}_{n,h}$
if Real **then**
  $B := s_0' + R \cdot S_A + E$
if Rand **then**
  $B \leftarrow_\$ \mathbb{Z}_p$
return $\tau = (R, B)$

Fig. 9. Definition of Real and Rand

Before proving our main theorem, we review a useful lemma for fresh case.

Lemma A.1. *Consider the two games* Real *and* Rand *between a challenger and an adversary* $\mathcal{B}$ *defined in Figure 9. Assume that the* $\mathsf{MERS\text{-}U}_{n,h}$ *problem is* (t, Q, ϵ)*-hard. Then, for all* (t', Q)*-adversary* $\mathcal{B}$ *with* $t' \approx t$*, we have*

$$|\Pr[\mathsf{Real}_{\mathcal{B}}(\kappa) \Rightarrow 1] - \Pr[\mathsf{Rand}_{\mathcal{B}}(\kappa) \Rightarrow 1]| \leq 2\mu\epsilon.$$

The proof of Lemma A.1 is in section B.

A.1 Fresh Case

Lemma A.2. *Suppose that there exists an adversary* $\mathcal{A}$ *that breaks* (t', Q, ϵ')*-*UF-CMA*-security of* MAC*. If the probability that the first forgery found by the adversary is more likely to be fresh:* $\Pr[\texttt{Fresh}] > \epsilon'/2$*, then we have another* (t, Q, ϵ)*-adversary* $\mathcal{B}$ *that breaks* $\mathsf{MERS\text{-}U}_{n,h}$ *with*

$$t \approx t' \text{ and } \epsilon \geq \epsilon'/(4\mu Q_{\mathsf{Verify}}) - Q_{\mathsf{Verify}}\alpha_{n,h},$$

where $Q_{\mathsf{Verify}} \leq Q$ *is the number of verification queries.*

Proof (Proof of Lemma A.2). We define the following games:

- Let G_0 be the original security game $\mathsf{Exp}^{\mathsf{uf\text{-}cma}}$.
- Let G_j for $j = 1, \ldots, Q_{\mathsf{Verify}}$ denote the games where the adversary is allowed to ask only j verification queries.
- We also define G'_j as same as the game G_j except that the tag oracle will use random R, B, β to compute σ instead of the real computation.

As [KPV+17], we have

$$\epsilon'/2 < \Pr[\mathtt{Fresh}] = \Pr[G_0 = 1] \leq \sum_{j}^{Q_{\mathsf{Verify}}} \Pr[G_j = 1].$$

Thus, what we should do is bounding $\Pr[G_j = 1]$.

Claim. Assume that $\mathcal{A}$ is a (t, Q)-adversary. for all j, there exists a (t', Q)-adversary $\mathcal{B}$ such that $t' \approx t$ and

$$|\Pr[G_j = 1] - \Pr[G'_j = 1]| \leq |\Pr[\mathsf{Real}_{\mathcal{B}}(\kappa) \Rightarrow 1] - \Pr[\mathsf{Rand}_{\mathcal{B}}(\kappa) \Rightarrow 1]| \, .$$

Proof (Proof of Claim). We construct $\mathcal{B}$ as follows:

1. $\mathcal{B}$ samples h and π.
2. $\mathcal{B}$ runs $\mathcal{A}$ on input 1^κ and simulates the oracles as follows:
   - $T(m)$:
     (a) sample a random $\beta \leftarrow_{\$} \{0,1\}^\nu$ and compute $A = \mathsf{h}(m, \beta)$.
     (b) query A to oracle Eval and obtain a pair (R, B).
     (c) return $\sigma := \pi(R, B, \beta)$.
   - $V(m, \sigma)$:
     (a) if (m, σ) is previously returned to $\mathcal{A}$, then $\mathcal{B}$ returns `Accept`.
     (b) if (m, σ) is not j-th verification query, then $\mathcal{B}$ returns `Reject`.
     (c) if (m, σ) is the j-th verification query; we call it (m^*, σ^*). let $(R^*, B^*, \beta^*) := \pi^{-1}(\sigma^*)$; compute $A^* := \mathsf{h}(m^*, \beta^*)$; send (R^*, A^*) to oracle Chal and obtain B'. If $\|B^* - B'\| = h$, then return `Accept`. otherwise, return `Reject`.

The j-th verification query is fresh by the definition. In addition, since the oracle Chal returns $B' := s'_0 + R^* \cdot S_{A^*}$, this simulated verification procedure correctly checks the Hamming weight of $\|B^* - (s'_0 + R^* \cdot S_{A^*})\|$ as the correct verification. Therefore, the simulation is perfect if A^* is fresh as we wanted. □

Claim. for all j,
$$\Pr[G'_j = 1] \leq \alpha_{n,h}$$

Proof (Proof of Claim). Fix a value $j \in \{1, \ldots, Q_{\mathsf{Verify}}\}$. In game G'_j, the adversary obtains no information on $(s'_0, s_0, s_1, \ldots, s_\mu)$ from the tagging oracle $T(\cdot)$ because the oracle returns random values (R, B). Therefore, the value $X := B^* - B' = B^* - (R^* \cdot S_{A^*} + s'_0)$ should be uniformly at random over $\mathbb{Z}_p$, since s'_0 is kept secret. Thus, the probability that the verification $\|B^* - B'\| = h$ passes is at most

$$\Pr[X \leftarrow \mathbb{Z}_p : \|X\| = h] = \binom{n}{h}/p = \alpha_{n,h}.$$

Combining those two claims, we obtain the following result: If $\mathcal{A}$ is (t, Q)-adversary, then there is a (t', Q)-adversary $\mathcal{B}$ such that $t' \approx t$ and

$$\begin{aligned}\Pr[G_j = 1] &\leq \Pr[G'_j = 1] + |\Pr[G_j = 1] - \Pr[G'_j = 1]| \\ &\leq \alpha_{n,h} + |\Pr[\mathsf{Real}_{\mathcal{B}}(\kappa) \Rightarrow 1] - \Pr[\mathsf{Rand}_{\mathcal{B}}(\kappa) \Rightarrow 1]|\end{aligned}$$

as we wanted. Applying Lemma A.1, we have

$$\Pr[G_j = 1] \leq \alpha_{n,h} + 2\mu\epsilon$$

under the assumption that the $\mathsf{MERS\text{-}U}_{n,h}$ problem is (t, Q, ϵ)-hard. Therefore, we have

$$\epsilon'/2 \leq \sum_{j}^{Q_{\mathsf{Verify}}} \Pr[G_j = 1] \leq Q_{\mathsf{Verify}}\alpha_{n,h} + 2Q_{\mathsf{Verify}}\mu\epsilon.$$

This yields

$$\epsilon \geq \epsilon'/(4Q_{\mathsf{Verify}}\mu) - Q_{\mathsf{Verify}}\alpha_{n,h}$$

as we wanted. □

A.2 Non-Fresh Case

Lemma A.3. *Let $\mu = \nu$. Suppose that there exists an adversary $\mathcal{A}$ that breaks (t', Q, ϵ')-UF-CMA-security of MAC. If the probability that the first forgery found by the adversary is more likely to be non-fresh, that is,* $\Pr[\texttt{Fresh}] \leq \epsilon'/2$, *then we have $\mathcal{B}$ that breaks the (t, Q, ϵ)-hardness of the $\mathsf{MERS\text{-}U}_{n,h}$ problem, where*

$$t \approx t' \text{ and } \epsilon \geq \epsilon'/2 - Q^2/2^{\mu}.$$

Proof. This proof is similar to the proof of the ROR-CMA security in section 5.

Let us construct an adversary $\mathcal{B}^{\mathsf{oracle}}$ who will distinguish between two oracles $\mathcal{O}$ and $\mathcal{U}$.

$\mathcal{B}$ samples $\pi, \mathsf{h}, s'_0, s_1, \ldots, s_\mu$ except s_0 as defined in KeyGen. It then runs $\mathcal{A}$ and simulates the oracles as follows:

- $T(m)$: On a query m,
  1. Sample β and compute $A := \mathsf{h}(m, \beta)$
  2. Call the oracle and obtain $(\tilde{R}, \tilde{B})$
  3. Compute $B := \tilde{B} + \tilde{R} \cdot (\sum_{i=1}^{\mu} A[i] \cdot s_i) + s'_0$
  4. Return $\sigma := \pi(\tilde{R}, B, \beta)$
- $V(m, \sigma)$: On a query (m, σ), $\mathcal{B}$ always answers `Reject`.

Finally, $\mathcal{B}^{\mathsf{oracle}}$ outputs 1 if any query to T or V contains β that has appeared in a previous query to T or V. It outputs 0 otherwise.

We note that if $\mathsf{oracle} = \mathcal{O}_{s,n,h}$, then $\tilde{B} = \tilde{R} \cdot s + e$, where $e \leftarrow_{\$} \mathfrak{H}_{n,h}$ and the simulation of T is perfect by letting $s_0 := s$.

Claim. If $\mathsf{oracle} = \mathcal{O}_{s,n,h}$, then the probability that $\mathcal{B}^{\mathsf{oracle}}$ outputs 1 is $\geq \epsilon'/2$

Proof (Proof of Claim). The proof is the same as that in [KPV+17, Proof of Claim 4.5]. The simulation of T is perfect. In addition, until $\mathcal{A}$ makes a valid forgery, the simulation of V is also perfect. The probability that $\mathcal{A}$ output his first forgery which is *not* fresh is simply lower bounded by $\epsilon' - \epsilon'/2 = \epsilon'/2$. Thus, we obtain the lower bound in the claim. □

Claim. If $\mathsf{oracle} = \mathcal{U}$, then the probability that $\mathcal{B}^{\mathsf{oracle}}$ outputs 1 is at most $\leq Q^2/2^{\mu}$.

Proof (Proof of Claim). The proof is the same as that in [KPV+17, Proof of Claim 4.6].

We have $A_i = A_j$ if and only if $\mathsf{h}(m_i, \beta_i) = \mathsf{h}(m_j, \beta_j)$. Now we will upper bound the probability that an adversary find such collision which imply the same probability that $\mathcal{B}^{\mathsf{oracle}}$ outputs 1, assuming that an adversary makes at most Q queries and fixing that up to the $(i-1)$-th query by which we assume that all the $\mathcal{A}$'s were distinct. Then we obtain two cases of collision:

- The probability of collision that the i-th query in which β_i will collide with a previous β_j is at most $(i-1)/2^{\nu}$.
- If the first collision does not happen then the probability of collision in $\mathsf{h}(m_i, \beta_i) = \mathsf{h}(m_j, \beta_j)$ will be $(i-1)/2^{\mu}$.

Then similarly to the proof in [KPC+11] we obtain $\sum_{n=1}^{Q}((i-1)/2^{\nu} + (i-1)/2^{\mu}) \leq Q^2/2^{\mu}$ where $\mu = \nu$. □

Combining two claims, we have

$$\epsilon \geq \epsilon'/2 - Q^2/2^{\mu}$$

as we wanted. □

B Proof of Lemma A.1

The proof is almost same as that in [KPV+17].

For $i = 0, \ldots, \mu$ and $A \in \{0,1\}^{\mu}$, we define $A[1..i]$ as the i-bit string $A_1 \ldots A_i \in \{0,1\}^i$. (We let $A[1..0] = \perp$.) For $i = 0, \ldots, \mu$, $\mathsf{RF}_i, \mathsf{RF}'_i \colon \{0,1\}^i \to \mathbb{Z}_p$ be two random functions. (If $i = 0$, then $\mathsf{RF}_0(\perp) = b'$ for some random $b' \leftarrow_{\$} \mathbb{Z}_p$.)

We define the line of games as follows:

- G_0: this game is the same as Real except that

- in the beginning, we sample 2μ elements $s_{1,0}, \ldots, s_{\mu,0}, s_{1,1}, \ldots, s_{\mu,1}$ from $\mathbb{Z}_p$ instead of $\mu+1$ elements $s_0, s_1, \ldots, s_\mu$ from $\mathbb{Z}_p$.
- in the computation of S_A, we compute $S_A := \sum_{j=1}^{\mu} s_{j,A[j]}$ instead of $S_A := s_0 + \sum_{j=1}^{\mu} A[j] \cdot s_j$. (We also replace the computation of S_{A^*}.)

– $G_{1,i}$ for $i = 0, \ldots, \mu$: this game is the same as G_0 except that
  - in the oracle Chal, we let $s_0' := \mathsf{RF}_i(A^*[1..i])$
  - in the oracle Eval, we compute $B := \mathsf{RF}_i(A[1..i]) + RS_A + E$ instead of $B := s_0' + RS_A + E$.

– G_2: this game is the same as $G_{1,\mu}$ except that
  - in the oracle Chal, we sample $B^* \leftarrow_\$ \mathbb{Z}_p$ instead of $B^* := s_0' + R^* \cdot S_{A^*}$
  - in the oracle Eval, we compute $B := \mathsf{RF}_\mu(A)$ instead of $B := \mathsf{RF}_\mu(A) + RS_A + E$.

Lemma B.1. $\Pr[G_0 = 1] = \Pr[\mathsf{Real} \Rightarrow 1]$

Proof. In G_0, we replace the computation of S_A. We note that if we set $s_0 := \sum_{j=1}^{\mu} s_{j,0}$ and $s_j := s_{j,1} - s_{j,0}$, we have $S_A = s_0 + \sum_{j=1}^{\mu} A[j] \cdot s_j = \sum_{j}^{\mu} s_{j,A[j]}$. In addition, if we choose $s_{j,k}$ uniformly at random, then $s_0, s_1, \ldots, s_\mu$ are also distributed according to the uniform distribution over $\mathbb{Z}_p$. Hence, the two games are equivalent. □

Lemma B.2. *We have* $\Pr[G_0 = 1] = \Pr[G_{1,0} = 1]$.

Proof. G_0 is the same as $G_{1,0}$, since s_0' can be interpreted as $\mathsf{RF}_0(\perp)$ [KPV+17]. □

Lemma B.3. *Let $\mathcal{B}$ be a (t, Q)-adversary. For all $i \in \{0, \ldots, \mu-1\}$, there exists a (t', Q)-adversary $\mathcal{D}$ such that*

$$t' \approx t \text{ and } |\Pr[G_{1,i} = 1] - \Pr[G_{1,i+1} = 1]| \leq 2 \cdot \mathsf{Adv}_{\mathcal{D}}^{\mathsf{MERS\text{-}U}_{n,h}}(\kappa).$$

Proof. Notice that for arbitrarily fixed $b \in \{0,1\}$ and two random functions RF_i and RF_i', we can define a new random function RF_{i+1} by

$$\mathsf{RF}_{i+1}(A[1...i+1]) := \begin{cases} \mathsf{RF}_i(A[1...i]) & \text{if } A[i+1] = b \\ \mathsf{RF}_i(A[1...i]) + \mathsf{RF}_i'(A[1...i]) & \text{o.w.} \end{cases}$$

Our adversary $\mathcal{D}$ guesses $E \leftarrow_\$ \{0,1\}$ as the prediction of $A^*[i+1]$ and simulate the oracles by using the above observation. We construct a distinguisher $\mathcal{D}$ as follows:

1. Given 1^κ, $\mathcal{D}$ prepares parameter values as follows:
   – Sample $b \leftarrow \{0,1\}$ and initialize $L := \emptyset$ and $L_i := \emptyset$.
   – Choose $s_{j,\beta} \leftarrow \mathbb{Z}_p$ for all $j \in [1,\mu]$ and $\beta \in \{0,1\}$ except for $s_{i+1,1-b}$.
   – Query to its oracle for Q times and obtain the answers (R_j, B_j') for $j \in [Q]$.
2. $\mathcal{D}$ runs $\mathcal{B}$ and simulates Eval and Chal as follows:
   – Simulation of Eval on input $A \in \{0,1\}^\mu$:

(a) Update $L := L \cup \{A\}$
(b) If $A[i{+}1] = b$, then $R \leftarrow_{\$} \mathbb{Z}_p$, $E \leftarrow_{\$} \mathfrak{H}_{n,h}$, compute $B := \mathsf{RF}_i(A[1...i]) + R \cdot (\sum_{j=1}^{\mu} S_{j,A[j]}) + E$ and return (R, B).
(c) Else, that is, if $A[i+1] = 1 - b$, then
   i. If L_i contains $(A[1...i], (R_j, B'_j))$ for some j, then let $(R, B') := (R_j, B'_j)$.
   ii. Else, use a next fresh pair, that is, $(R, B') := (R_j, B'_j)$ for the first j. Add $(A[1...i], (R_j, B'_j))$ to the list L_i.
   iii. Compute $B := \mathsf{RF}_i(A[1...i]) + R \cdot (\sum_{j=1, j \neq i+1}^{\mu} S_{j,A[j]}) + B'$ and return (R, B).

– Simulation of Chal on input R^* and A^*:
(a) If $A^*[i+1] \neq b$, abort.
(b) Else, define $S_{A^*} := \sum_{j}^{\mu} S_{j,A^*[j]}$.
(c) Return $B^* := R^* \cdot S_{A^*} + \mathsf{RF}_i(A^*[1...i])$.

3. Finally, $\mathcal{B}$ will outputs its decision d and stops. $\mathcal{D}$ outputs $d \wedge (A^* \notin L)$.

Suppose that the guess b is correct. This happens with probability 1/2. If so, $\mathcal{D}$ perfectly simulates Chal, since $\mathsf{RF}_{i+1}(A^*[1...i+1]) = \mathsf{RF}_i(A^*[1...i])$ if $A^*[i+1] = b$. We next analyze the simulation of Eval: If $A[i+1] = b$, then we have $\mathsf{RF}_{i+1}(A[1...i+1]) = \mathsf{RF}_i(A[1...i])$. Thus, the distributions of Z are equal each other. Otherwise, that is, if $A[i+1] = 1 - b$, then we consider two cases: If the oracle outputs $B' := Rs + E$ with $E \leftarrow_{\$} \mathfrak{H}_{n,h}$, then we have

$$\begin{aligned} B &:= \mathsf{RF}_i(A[1...i]) + R \cdot \left(\sum_{j=1, j \neq i+1}^{\mu} s_{j,A[j]} \right) + R \cdot s + E \\ &= \mathsf{RF}_i(A[1...i]) + R \cdot \left(\sum_{j=1}^{\mu} s_{j,A[j]} \right) + E \end{aligned}$$

by letting $s_{i+1,1-b} := s$. Therefore, if the oracle is $\mathcal{O}_{s,n,h}$, then $\mathcal{D}$ perfectly simulates G_i. On the other hand, if the oracle is $\mathcal{U}$, that is, $B' = Rs + E + U$ with $E \leftarrow_{\$} \mathfrak{H}_{n,h}$ and $U \leftarrow_{\$} \mathbb{Z}_p$, then we have

$$\begin{aligned} B &:= \mathsf{RF}_i(A[1...i]) + R \cdot \left(\sum_{j=1, j \neq i+1}^{\mu} s_{j,A[j]} \right) + R \cdot s + E + U \\ &= \mathsf{RF}_i(A[1...i]) + U + R \cdot \left(\sum_{j=1}^{\mu} s_{j,A[j]} \right) + E. \end{aligned}$$

By letting $U := \mathsf{RF}'_i(A[1...i])$, we observe that $\mathcal{D}$ perfectly simulates G_{i+1}.

Therefore, we have

$$t' \approx t \text{ and } |\Pr[G_{1,i} = 1] - \Pr[G_{1,i+1} = 1]| = 2 \cdot \mathsf{Adv}_{\mathcal{D}}^{\mathsf{MERS\text{-}U}_{n,h}}(\kappa)$$

as we wanted. □

Lemma B.4. *We have* $\Pr[G_{1,\mu} = 1] = \Pr[G_2 = 1]$.

Proof. This is almost obvious. Notice that every query A to Eval and Chal should be fresh. Thus, in both cases, $\mathsf{RF}_\mu(A)$ makes B (and B^*) random. □

Lemma B.5. *We have* $\Pr[G_2 = 1] = \Pr[\mathsf{Rand} \Rightarrow 1]$.

Proof. In G_2, all returned values (R, B) from Eval and B^* from Chal are fresh and random if $A^* \notin L$. We also know that in Rand, all values are fresh and random if $A^* \notin L$, because s'_0 is random and kept secret. Therefore, there are no difference between G_2 and Rand if $A^* \notin L$. This completes the proof. □