Two-Round Man-in-the-Middle Security from LPN

David Cash[1], Eike Kitz[2], and Stefano Tessaro[3]

[1] Rutgers University
[2] Ruhr University Bochum
[3] University of California, Santa Barbara

Abstract. Secret-key authentication protocols have recently received a considerable amount of attention, and a long line of research has been devoted to devising efficient protocols with security based on the hardness of the learning-parity with noise (LPN) problem, with the goal of achieving low communication and round complexities, as well as highest possible security guarantees.
In this paper, we construct 2-round authentication protocols that are secure against sequential man-in-the-middle (MIM) attacks with tight reductions to LPN, Field-LPN, or other problems. The best prior protocols had either loose reductions and required 3 rounds (Lyubashevsky and Masny, CRYPTO'13) or had a much larger key (Kiltz et al., EUROCRYPT'11 and Dodis et al., EUROCRYPT'12). Our constructions follow from a new generic deterministic and round-preserving transformation enhancing actively-secure protocols of a special form to be sequentially MIM-secure while only adding a limited amount of key material and computation.

Keywords. Secret-key authentication, Man-in-the-Middle security, LPN, Field LPN.

1 Introduction

This paper constructs efficient provably-secure protocols for *secret-key* authentication, i.e., for the basic cryptographic task where one party, called the *prover*, proves to another – the *verifier* – that they share the same key. Theoretical constructions of such protocols (with strong security, to be defined below) exist from any one-way function. Moreover, practical two-round protocols can be built from any message-authentication code (MAC) by having one party authenticate a random challenge, and can be instantiated efficiently for example assuming AES-128 is unpredictable.

In contrast, this paper contributes to a line of work [16, 18, 13, 19, 20, 10, 15, 22] on building provably-secure authentication protocols with security reductions to the *learning parity with noise (LPN)* and related problems that are as efficient as possible, meaning that key-size, communication, and rounds are minimized. LPN problem provides confidence in security due to the failure to find polynomial-time algorithms for it and its variants, despite wide interest [7,

21, 5], and finding constructions of cryptographic primitives based on LPN has given rise to a substantial body of works [6, 14, 2, 17].

The motivation behind LPN-based authentication protocols is their potential to be implemented with different efficiency characteristics from protocols with security reductions to blockcipher security or to problems from number theory and related fields. For instance, the parallel nature of LPN-based protocols seems difficult to achieve with factoring or discrete-log type assumption. The potential efficiency benefits of LPN-based implementations are a subject of ongoing research, which has identified some advantageous scenarios [15] but also invented faster attacks [21]. We thus focus on developing techniques for protocol design and theoretical analysis that beat previous asymptotic runtimes, key sizes, and round complexity of protocols with similar security reductions. We make no specific claims of more efficient protocols in specific deployment scenarios.

Concurrently to the above, the recent interest on secret-key authentication has also motivated attempts to develop a better understanding of its foundations, providing theoretical constructions based on concrete number-theoretic assumptions like the *Decisional Diffie-Hellman* (DDH) assumption, or general assumptions like weak pseudorandom functions [10, 22]. We will also contribute to these lines of work with new constructions.

But before we turn to describing our contributions in detail, we first give an overview of different security notions for secret-key authentication, as well as of previous works.

Security notions. Several security notions for secret-key authentication protocols have been considered, inspired by corresponding notions for the task of public-key authentication [12]. The weakest, *passive security*, says that an attacker should not be able to fool the verifier after observing several sessions between an honest prover and an honest verifier. This seems unreasonably weak for most settings, so the stronger *man-in-the-middle (MIM) security* notion says that no attacker should be able to cause the verifier to accept in any session where a message has been changed. Realizing that MIM security from LPN seems difficult to achieve efficiently, several works instead targeted an intermediate notion called *active security* which says that the attacker cannot fool the verifier after interacting with the prover arbitrarily and observing sessions passively.

The LPN Assumption (and its variants). Recall that for parameters $\ell \in \mathbb{N}$ and $0 \leq \gamma \leq \frac{1}{2}$, the (decisional) *Learning Parity with Noise (LPN)* problem $\mathsf{LPN}_{\ell,\gamma}$ is the problem of distinguishing a polynomial number of samples of the form $(\mathbf{r}_i, \mathbf{r}_i^T\mathbf{s} + e_i)$, for a common random secret $\mathbf{s} \in \{0,1\}^\ell$, random vector $\mathbf{r}_i \in \{0,1\}^\ell$, and random bit e_i (taking value one with probability γ), from samples of the form $(\mathbf{r}_i, b_i)$, where b_i is a random bit. The corresponding $\mathsf{LPN}_{\ell,\gamma}$ assumption is that no efficient (i.e., polynomial-time) attacker can distinguish between the two distributions, except with negligible advantage. Ignoring the obvious differences in the error distributions, this is the modulo 2 variant of the learning with error problem introduced in [27].

We are also going to consider a variant of the LPN problem, introduced and studied in [15], called *Field LPN*. The $\mathsf{Field\text{-}LPN}_{\ell,\gamma}$ problem is very similar,

however samples have the form $(\mathbf{r}_i, \mathbf{r}_i \circ \mathbf{s} + \mathbf{e}_i)$ or $(\mathbf{r}_i, \mathbf{r}'_i)$, where $\circ$ denotes multiplication of ℓ-bit vectors interpreted as elements of the extension field $\mathbb{F}_{2^\ell}$, $\mathbf{e}_i$ is a random vector where each component is 1 independently with probability γ, and $\mathbf{r}'_i$ is uniform.

Prior constructions. Let us briefly outline the landscape of earlier works on secret-key authentication. Table 1 summarizes some of these results.

Juels and Weis [18] first pointed out that a very simple two-round secret-key authentication protocol by Hopper and Blum [16], called the *HB protocol*, enjoys very low hardware complexity, and is hence amenable to implementations on RFID tags. Moreover, they proved that it is passively secure under the LPN assumption. Also in [18], they proposed a further three-round protocol, called HB^+, which was proven actively secure in its sequential version under the LPN assumption, and later the proof was extended to its parallel version by Katz *et al.* [19]. The round complexity was then reduced to *two* rounds by a new protocol of Kiltz *et al.* [20], and in contrast to HB^+, this latter protocol enjoys a tight reduction to the hardness of LPN. Heyse *et al.* [15] then proposed an even more efficient two-round protocol, called *Lapin*, based on the hardness of the *field* LPN problem. We stress that three-round protocols are less attractive than two-round ones since the prover needs to keep a *state* (beyond the secret key), which is problematic on lightweight devices like RFID tags.

In contrast, progress has been significantly harder in the context of MIM security. On the one hand, researchers have attempted to design multiple HB-like protocols with MIM security [8, 11, 25, 13] without or only partial security proofs. Otherwise, provably MIM-secure constructions all in fact provide a full message-authentication code (MAC) secure under LPN or Field LPN [20, 10, 15]. Unfortunately, these constructions are significantly less efficient than the existing actively secure protocols mentioned above.

While following [3] the notion of MIM security traditionally allows an attacker to interact with arbitrarily many instances of the prover and the verifier concurrently, Lyubashevky and Masny [22] recently considered the notion of *sequential MIM* (sMIM) security, which slightly weakens MIM to only allow the attacker to interfere with non-overlapping sequential sessions. They argue this notion is sufficient for situations in which keys are managed to never allow parallel session, and the sMIM notion is an interesting technical step towards improving authentication protocol security beyond active security while maintaining efficiency. Moreover, existing MIM attacks against actively secure protocols are often sequential (e.g., [26]). They give new protocols based on LPN and field-LPN that nearly match the complexities of actively-secure ones, but all require three rounds and suffer from a non-tight reduction to the underlying problem.

With respect to other assumptions, we also note that efficient three-round constructions from DDH and weak pseudorandom functions have also been given, achieving both active security [10] and sMIM security [22]. Two-round MIM secure protocols from PRGs have recently been proposed [9].

All of our constructions come with reductions running in polynomial time and succeeding with probability polynomially proportional to that of a given

attack. One may consider looser reductions via so-called *complexity leveraging* where the reduction loses an exponential factor, with the view that one can enlarge the security parameter to compensate for the loss. Indeed one can prove (say) the AUTH_2 protocol from [20] as a fully-secure MAC with an exponential loss of security. A concrete instantiation of the result (assuming the BKW attack complexity is optimal [7]) will be more efficient than the other approaches we have outlined.

Polynomial reductions, however, are preferred as they are more robust to algorithmic advances against the underlying problems. Achieving them is, in our view, an interesting theoretical challenge that requires new techniques. In an implementation it is not clear to the authors if either approach (leveraging or polynomial reductions) is necessarily more secure given the many factors one must consider.

Our contributions. We provide the to-date most efficient constructions of sMIM-secure authentication protocols based on the hardness of LPN, as well as on other assumptions. Our constructions are *two* rounds and the first message consists of a truly random challenge, and enjoy tight security reduction to the underlying assumption.

We improve upon the round complexity of existing sMIM-secure protocols without increasing key length and communication complexity, and without resorting to complexity leveraging. See Table 1 for a comparison of two of our new protocols to prior work. Note that our protocols are only a small constant factor less efficient than the best known actively (or even passively) secure protocols.

At the high level, our constructions follow from a generic transformation that upgrades a two round protocol of a special form to be sMIM-secure without introducing significant overhead. The required form is not especially contrived, but requires some care in its formalization and we present examples of such protocols to obtain our instantiations. We note that our reduction does not employ rewinding or forking lemmas like [22], and is tighter and (arguably, to our taste) simpler as a result.

Our first construction achieves sMiM security with a tight reduction to LPN, two rounds of communication, and only a modest increase in either key size or communication over [22]. Our second construction, from *Field LPN*, matches the key size and communication of prior work in two rounds instead of three and has a tight reduction. In fact, for an appropriate choice of components, the second construction can be understood as a two-round version of the three-round protocol from [22], though their proof does not cover the two round version.

We also provide a simple construction of a two-round sMIM secure authentication protocol based on the DDH assumption, where the prover response consists of *two* group elements. Interestingly, the same construction was proven MIM secure under the (less standard) Gap-CDH assumption in [10].

Our last construction is based on an arbitrary weak PRF. The complexity of the construction is comparable to the one building a MAC from a weak PRF, using for example the constructions in [23, 24, 1]. However, our new protocols enjoy much better parallelism when compared to the naive approach, and is

hence interesting on its own right. It is also fair to point out that [22] accomplishes in three rounds the harder task of finding a generic construction from a (randomized) *weak* PRF. We observe however that the only known concrete instantiations of weak PRFs are based on LPN/LWE-type assumptions as well as on DDH, and for all these concrete instantiations our constructions are more efficient.

We remark that it is not hard to see that our proofs do not show (full) MIM security, but we are not aware of an explicit MIM attack against the protocols.

Organization. Section 2 contains basic definitions used below. In Section 3 we describe our transformation from weaker protocols of a special form, and in Section 4 we give several instantiations of the transformation.

Protocol	Rounds	Security			Complexity		Compl. trade-off	
		Assumption	Active$^{(*)}$	sMiM	key size	com.	key size	com.
HB [16]	2	$\mathsf{LPN}_{\ell,\gamma}$	–	–	ℓ	ℓ^2	ℓ^2	2ℓ
HB$^+$ [18]	3	$\mathsf{LPN}_{\ell,\gamma}$	$q\sqrt{\epsilon}$	–	2ℓ	$2\ell^2$	$2\ell^2$	3ℓ
AUTH_2 [20]	2	$\mathsf{LPN}_{\ell,\gamma}$	$q\epsilon$	–	2ℓ	ℓ^2	$2\ell^2$	2ℓ
Lapin [15]		$\mathsf{Field\text{-}LPN}_{\ell,\gamma}$			2ℓ	2ℓ	–	–
MAC_2 [20]	2	$\mathsf{LPN}_{\ell,\gamma}$	$q\epsilon$	$q\epsilon$	$3\ell^2$	ℓ^2	ℓ^3	4ℓ
Lapin+MAC_2 [15]		$\mathsf{Field\text{-}LPN}_{\ell,\gamma}$			ℓ^2	4ℓ	–	–
LM [22]	3	$\mathsf{LPN}_{\ell,\gamma}$	$q\sqrt{\epsilon}$		$\ell^{2\ (**)}$	ℓ^2	ℓ^2	3ℓ
		$\mathsf{Field\text{-}LPN}_{\ell,\gamma}$			4ℓ	3ℓ	–	–
This work	2	$\mathsf{LPN}_{\ell,\gamma}$	$q\epsilon$		5ℓ	ℓ^2	$2\ell^2$	3ℓ
		$\mathsf{Field\text{-}LPN}_{\ell,\gamma}$			4ℓ	3ℓ	–	–

Table 1: Authentication protocols based on LPN-related Assumptions. The security column lists the best possible security reduction from the given assumption, where q is the number of tag and verification queries. (The two MAC_2 protocols are even secure in the full MiM model.) The complexity column lists the key sizes and communication complexity of the protocol (with lower-order terms dropped), where ℓ parameterizes the hardness of the assumption. All LPN-based protocols offer a trade-off between key size and communication, which is listed in the last two columns. $^{(*)}$: Reductions to active security only considered one challenge session, and thus did not have the factor q. We state the bound for q challenge sessions for a fair comparison to MiM security. $^{(**)}$: We remark that the key size of the LPN-based protocol in [22] is ℓ^2 but one may be able to reduce it to $O(\ell)$ by using an *almost* pairwise independent hash function.

2 Preliminaries

For a set $\mathcal{X}$, $x \xleftarrow{\$} \mathcal{X}$ denotes sampling x from $\mathcal{X}$ according to the uniform distribution. We use bold lowercase letters for vectors and bold uppercase letters for matrices, e.g., $\mathbf{x} \in \mathbb{F}_2^\ell$, $\mathbf{X} \in \mathbb{F}_2^{\ell \times n}$. For $\mathbf{c} \in \mathbb{F}_2^\ell$, let $\mathbf{M_c}$ denote the matrix of the linear map $l_{\mathbf{c}}$ implementing the finite field multiplication with $\mathbf{c}$ when interpreted as an element in $\mathbb{F}_{2^\ell}$.[4]

[4] This representation is unique once the irreducible polynomial f defining $\mathbb{F}_{2^n} = \mathbb{F}_2[x]/(f)$ is fixed.

Symmetric authentication syntax. We are going to consider secret-key *authentication* protocols, where a prover proves to a verifier that they hold the same secret key over two or more rounds.

More formally, an *r-round symmetric authentication protocol* with associated key space $\mathcal{K}$ is a triple of algorithms $\mathsf{Auth} = (\mathsf{Gen}, \mathsf{P}, \mathsf{V})$ with the following properties:

- Key Generation. The probabilistic key-generation algorithm $K \leftarrow \mathsf{Gen}(1^k)$ takes as input a security parameter $k \in \mathbb{N}$ (in unary) and outputs a secret key $K \in \mathcal{K}$.
- Interactive Execution. The probabilistic interactive algorithms P and V, which we refer to as the *prover* and the *verifier*, take both as input a secret key $K \in \mathcal{K}$, synchronously interact with each other over r rounds, and finally V always receives the last message and outputs a decision $\mathsf{out}(\mathsf{P}_K, \mathsf{V}_K) \in \{\mathsf{accept}, \mathsf{reject}\}$.

We say that Auth has completeness error α if for all $k \in \mathbb{N}$, $\Pr[\mathsf{out}(\mathsf{P}_K, \mathsf{V}_K) = \mathsf{reject} \,;\, K \leftarrow \mathsf{Gen}(1^k)] \leq \alpha$. In this paper, we will focus on the simpler case of *two-round* protocols, where additionally the first message is a random challenge $c \in \mathcal{C}$ for some set $\mathcal{C}$. We call such protocols *two-round random-challenge* secret-key authentication protocols. In particular, in such protocols the prover simplifies to a probabilistic algorithm P_K, taking the challenge and the secret key K, and producing the message t to be sent back to the adversary. Moreover, for a challenge $c \in \mathcal{C}$ and response t from the prover, the verifier is fully specified by an algorithm $\mathsf{V}_K(c, t) \in \{\mathsf{accept}, \mathsf{reject}\}$.

Security. Several security notions for symmetric-key authentication protocols have been considered in the literature. The weakest one, *passive security*, says that an attacker should not be able to fool the verifier after observing several sessions between a honest prover and a honest verifier. The stronger notion called *active security* says that the attacker cannot fool the verifier after interacting with the prover arbitrarily and observing sessions passively.

This paper targets the security notion of *(sequential) security against man-in-the-middle attacks* (or s-mim security, for short). Here, the adversary acts as a man-in-the middle in a sequence of independent sessions between the prover and the verifier, all with the same secret key. The adversary wins whenever it manages to let the verifier accept in some session *and* has changed at least one of the messages sent by the prover or the verifier. We are going to formalize this notion for the relevant case of two-round protocols with random challenge.

Concretely, we describe this security notion via the following game S-MIM for an attacker A and a two-round random-challenge authentication protocol $\mathsf{Auth} = (\mathsf{Gen}, \mathsf{P}, \mathsf{V})$ with challenge set $\mathcal{C}$.

main S-MIM:	**Procedure** $\mathrm{P}(c')$:
$\mathsf{sid} \leftarrow 0$	If $c'[\mathsf{sid}] = \perp$ then
$K \xleftarrow{\$} \mathsf{Gen}(1^k)$	$\quad c'[\mathsf{sid}] \leftarrow c'$, $t[\mathsf{sid}] \xleftarrow{\$} \mathsf{P}_K(c')$
Run $A^{\mathrm{C}(),\mathrm{P}(),\mathrm{V}()}(1^k)$	Ret $t[\mathsf{sid}]$
Ret $\exists i$: $(c[i], t[i]) \neq (c'[i], t'[i]) \wedge d[i] = \mathsf{accept}$	
	Procedure $\mathrm{V}(t')$:
Procedure $\mathrm{C}()$:	$t'[\mathsf{sid}] \leftarrow t'$, $c \xleftarrow{\$} \mathrm{C}()$
If $c[\mathsf{sid}] = \perp$ then	$d[\mathsf{sid}] \leftarrow \mathsf{V}_K(c, t'[\mathsf{sid}])$
$\quad c[\mathsf{sid}] \xleftarrow{\$} \mathcal{C}$	$\mathsf{sid} \leftarrow \mathsf{sid} + 1$
Ret $c[\mathsf{sid}]$	Ret $d[\mathsf{sid}]$

In the game, the attacker makes calls to three oracles, $\mathrm{C}(\cdot), \mathrm{P}(\cdot)$ and $\mathrm{V}(\cdot)$. All oracles use a global variable sid to "synchronize" the sessions being simulated. The first oracle returns, for every session, a new random challenge. The oracle $\mathrm{P}(c')$ runs the prover on input c' and returns the response t. Oracle $\mathrm{V}(t')$ checks that t' is a valid response for the current session challenge $c[\mathsf{sid}]$ (obtained by calling $\mathrm{C}()$), and increases the session number. Note that there is a unique value $c[\mathsf{sid}]$ defined in every session, and P only provides (at most) one valid challenge-tag pair (c', t) per session. The $\mathsf{s\text{-}mim}$ advantage is $\mathbf{Adv}^{\mathsf{s\text{-}mim}}_{\mathsf{Auth}}(A) = \Pr\left[\mathsf{S\text{-}MIM}^{A}_{\mathsf{Auth}} \Rightarrow \mathsf{true}\right]$, and we say that Auth is (t, r, ϵ)-$\mathsf{s\text{-}mim}$-secure if for all attackers A with time complexity t and running at most r sessions, we have $\mathbf{Adv}^{\mathsf{s\text{-}mim}}_{\mathsf{Auth}}(A) \leq \epsilon$.

Hash functions. Our constructions rely on almost pairwise-independent hash functions.

Definition 1 (Almost pairwise-independent hash functions). *For $\delta \geq 1$, a function $\mathsf{H} : \mathcal{K}_{\mathsf{H}} \times \mathcal{X} \to \mathcal{Y}$ is* δ-almost pairwise-independent *if*

$$\Pr\left[\, \mathsf{H}_{K_{\mathsf{H}}}(x) = y \wedge \mathsf{H}_{K_{\mathsf{H}}}(x') = y' \,\right] \leq \frac{\delta}{|\mathcal{Y}|^2}$$

for all distinct $x, x' \in \mathcal{X}$ and all $y, y' \in \mathcal{Y}$, and where $K_{\mathsf{H}} \xleftarrow{\$} \mathcal{K}_{\mathsf{H}}$. Moreover, by itself, $\mathsf{H}_{K_{\mathsf{H}}}(x)$ is uniformly distributed over $\mathcal{Y}$.

The requirement that a single input has uniformly distributed output is not common, but will be useful in applications and satisfied by the construction given below. Moreover, Definition 1 implies *adaptive* security, i.e., when given x, $\mathsf{H}_{K_{\mathsf{H}}}(x) = y$, for any x' and y' chosen *adaptively* depending on y, the probability that $\mathsf{H}_{K_{\mathsf{H}}}(x') = y'$ is at most $\delta/|\mathcal{Y}|$.

Lemma 2. *If H is δ-almost pairwise-independent, then for every (unbounded) adversary A and every $x \in \mathcal{X}$, we have*

$$\Pr[\, \mathsf{H}_{K_{\mathsf{H}}}(x') = y' \wedge x' \neq x \;:\; K_{\mathsf{H}} \xleftarrow{\$} \mathcal{K}_{\mathsf{H}},\ (x', y') \xleftarrow{\$} A(\mathsf{H}_{K_{\mathsf{H}}}(x), x) \,] \leq \frac{\delta}{|\mathcal{Y}|}\ .$$

Proof. Assume wlog that A is deterministic, and let $x'(x,y)$ and $y'(x,y)$ be the values of x' and y' output by A on inputs y, x, where $x'(x,y) \neq x$ by assumption. Then,

$$\begin{aligned}&\Pr\left[\,\mathsf{H}_{K_\mathsf{H}}(x') = y' \;:\; K_\mathsf{H} \xleftarrow{\$} \mathcal{K}_\mathsf{H},\ (x',y') \xleftarrow{\$} A(\mathsf{H}_{K_\mathsf{H}}(x), x)\,\right]\\ &= \sum_y \Pr\left[\,\mathsf{H}_{K_\mathsf{H}}(x) = y \wedge \mathsf{H}_{K_\mathsf{H}}(x'(x,y)) = y'(x,y)\,\right],\end{aligned}$$

which is smaller than $|\mathcal{Y}| \cdot \frac{\delta}{|\mathcal{Y}|^2} = \frac{\delta}{|\mathcal{Y}|}$. □

A CONSTRUCTION. We will make use of the following key-length efficient construction of a δ-almost-pairwise independent function, where $\mathcal{K}_\mathsf{H} = \mathbb{F}^2$, $\mathcal{Y} = \mathbb{F}$ and $\mathcal{X} = \mathbb{F}^\ell$ for some finite field $\mathbb{F}$. The function, given $K_\mathsf{H} = (a,b) \in \mathbb{F}^2$ and input $x = (x_0, \ldots, x_{\ell-1}) \in \mathbb{F}^\ell$, outputs $\mathsf{H}_{a,b}(x) = \sum_{i=0}^{\ell-1} x_i \circ a^i + b$.

Lemma 3. *The function* H *above is* δ*-almost pairwise independent for* $\delta = \ell - 1$.

The folklore proof is given for completeness.

Proof. Fix $x = (x_0, x_1, \ldots, x_{\ell-1})$ and $x' = (x'_0, x'_1, \ldots, x'_{\ell-1})$. Also, we define the polynomial $p_x(a) = \sum_{i=0}^{\ell-1} x_i \circ a^i$, and analogously, define $p_{x'}(a)$. Given two $y, y' \in \mathbb{F}$, we look at the number of keys (a,b) such that $p_x(a) + b = y$ and $p_{x'}(a) + b = y'$. This in particular implies that a needs to satisfy

$$p_x(a) - p_{x'}(a) = \sum_{i=0}^{\ell-1} (x_i - x'_i) \circ a^i = y - y' \;,$$

and since there exists i with $x_i \neq x'_i$, note that by the Schwartz-Zippel lemma there are at most $\ell - 1$ solutions a with the above property, since $p_x(a) - p_{x'}(a)$ is a polynomial of degree at most $\ell - 1$. Each such a defines a unique b, and thus there are overall at most $\ell - 1$ solutions, and each one of them is taken with probability $|\mathbb{F}|^2$.

Finally, note that the distribution of $\mathsf{H}_{a,b}(x)$ is, by itself, uniform, because the term b is uniform, and thus completely blinds the output. □

3 Generic Construction

This section presents our main result, a generic construction of a two-round sequential MIM-secure authentication protocol Auth. Our construction relies on a simpler two-round symmetric authentication protocol Auth' used as a component and which satisfies a particular form of security, in addition to having a structured tag space, as we discuss next. Later below, we will provide several instantiations of this generic construction in Section 4 via constructions of Auth' based on a set of different assumptions.

3.1 Tools

Our construction is going to rely on an authentication protocol $\mathsf{Auth} = (\mathsf{Gen}, \mathsf{P}, \mathsf{V})$ whose responses given by the prover (which we call *tags*, following existing conventions in the literature) $\tau \xleftarrow{\$} \mathsf{P}_K(c)$ are composed of two distinct components $\tau = (\tau_1, \tau_2) \in \mathcal{T}_1 \times \mathcal{T}_2$. We refer to τ_1 and τ_2 as the *left* and *right* tag, respectively. In addition to this, we are going to require that the protocol satisfies two new properties which we now introduce and discuss.

Tag sparsity. The first property is a *combinatorial* property on the tag space of Auth. We are going to require that given any challenge c, any secret key K, and any *left* component of the tag τ_1, there are only few right components τ_2 such that $\tau = (\tau_1, \tau_2)$ is a valid tag for challenge c and key K. This is captured formally by the following definition.

Definition 4 (Right tag-sparsity). *For an $\epsilon = \epsilon(k)$, we say that $\mathsf{Auth} = (\mathsf{Gen}, \mathsf{P}, \mathsf{V})$ with tags in $\mathcal{T}_1 \times \mathcal{T}_2$, challenge space $\mathcal{C}$, and key space $\mathcal{K}$ has* ϵ-sparse right tags *(or alternatively, Auth has* ϵ-right tag sparsity*) if*

$$\Pr\left[\mathsf{V}_K(c, (\tau_1, \tau_2)) = \mathsf{accept};\ \tau_2 \xleftarrow{\$} \mathcal{T}_2\right] \leq \epsilon$$

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

Note that one equivalent formulation is that for all K, c, and τ_1, there are at most $\epsilon \cdot |\mathcal{T}_2|$ valid τ_2.

ROR-CMA security. We also consider a new property called *real-or-random right-tag chosen-message security* (or $\mathsf{ror\text{-}cma}$ security, for short), which is specific to protocols as above with tag space $\mathcal{T}_1 \times \mathcal{T}_2$. It considers a game where an attacker first receives a challenge c^*, then can obtain prover tags for arbitrary challenges of its choice, and at the end can issue *exactly one* verification query for the challenge c^*. The notion demands that the attacker cannot distinguish this game from another game where queries for challenges $c \neq c^*$ have the right tag τ_2 replaced by a *random* element from the same set. Formally, we introduce the following two games – denoted $\mathsf{ROR\text{-}CMA}(0), \mathsf{ROR\text{-}CMA}(1)$ – involving Auth as well as an adversary A which outputs a decision value in $\{\mathsf{true}, \mathsf{false}\}$ at the end of the game:

main $\mathsf{ROR\text{-}CMA}(b)$:	**Procedure** $\mathrm{T}(c)$:
$K \xleftarrow{\$} \mathsf{Gen}(1^k)$	$(\tau_1, \tau_2^1) \xleftarrow{\$} \mathsf{P}_K(c)$, $\tau_2^0 \xleftarrow{\$} \mathcal{T}_2$
$c^* \xleftarrow{\$} \mathcal{C}$	If $c = c^*$ then
$(\tau^*, state) \xleftarrow{\$} A^{\mathrm{T}(\cdot)}(c^*)$	Ret $\tau = (\tau_1, \tau_2^1)$
$d \leftarrow \mathsf{V}_K(c^*, \tau^*)$	Else ret $\tau = (\tau_1, \tau_2^b)$
Ret $A(state, d)$	

Then, for an attacker A and a two-round protocol Auth, we define the $\mathsf{ror\text{-}cma}$ *advantage* as

$$\mathbf{Adv}^{\mathsf{ror\text{-}cma}}_{\mathsf{Auth}}(A) = \Pr[\mathsf{ROR\text{-}CMA}^A_{\mathsf{Auth}}(0) \Rightarrow \mathsf{true}] - \Pr[\mathsf{ROR\text{-}CMA}^A_{\mathsf{Auth}}(1) \Rightarrow \mathsf{true}]\ .$$

Accordingly, we say that Auth is (t, q, ϵ)-$\mathsf{ror\text{-}cma}$-secure if for all t-time attackers A issuing at most q queries to oracle $\mathrm{T}(\cdot)$, we have $\mathbf{Adv}^{\mathsf{ror\text{-}cma}}_{\mathsf{Auth}}(A) \leq \epsilon$.

Relation to active security. We stress that $\mathsf{ror\text{-}cma}$ security and negligible right-tag sparsity, when achieved simultaneously, do not even imply passive security. Indeed, it is easy to modify any protocol with these two properties into one accepting tags of the form $(\tau_1, 0)$ for every K and c (and hence becoming completely insecure) without invalidating these two properties. However, any such protocol can easily be enhanced to be secure against *active* adversaries by blinding τ_2 with a secret field element K, either via addition or multiplication. (Note that negligible right-tag sparsity implies that the set of right tags has overwhelming size.)

Nonetheless, in order to better understand our construction below, it is important to observe why the resulting protocol is *not* necessarily $\mathsf{s\text{-}mim}$ secure. Consider e.g. the protocol such that $\mathsf{P}_K(c) = (\tau_1, \tau_2 = \mathsf{PRF}_K(\tau_1 \| c))$ for a random τ_1 and pseudorandom function PRF with key K and n-bit output, and for which V_K accepts (τ_1, τ_2) on input c if and only if $\mathsf{PRF}_K(\tau_1 \| c)$ has Hamming distance at most 1 from τ_2. One can verify that this protocol is $\mathsf{ror\text{-}cma}$ secure and has negligible right-tag sparsity. But when the above tranformation is applied, resulting in tags $(\tau_1, \tau_2 = \mathsf{PRF}_K(\tau_1 \| c) + K')$, an attacker can easily derive a new valid tag for c as $(\tau_1, \tau_2 + \Delta)$ for any weight-one Δ – hence breaking $\mathsf{s\text{-}mim}$ security. (Similar counterexamples can be built when blinding via multiplication.)

3.2 The Generic Construction

We now turn to describing our generic construction transforming a $\mathsf{ror\text{-}cma}$-secure two-round random challenge authentication protocol Auth' with ϵ-right tag sparsity (for a small ϵ) into a sequential MIM secure two-round authentication protocol.

Description. Let $\mathsf{Auth}' = (\mathsf{Gen}', \mathsf{P}', \mathsf{V}')$ be two-round authentication protocol with associated key space $\mathcal{K}$, challenge space $\mathcal{C}$, and split tag space $\mathcal{T} = \mathcal{T}_1 \times \mathcal{T}_2$, where we assume that $\mathcal{T}_2 = \mathbb{F}$ is a finite field.[5] We will use $+$ and $\circ$ to denote addition and multiplication of field elements, respectively. Let $\mathsf{H} : \mathcal{K}_{\mathsf{H}} \times \mathcal{T}_1 \to \mathbb{F}$ be a hash function. We build a 2-round symmetric authentication protocol $\mathsf{Auth} = (\mathsf{Gen}, \mathsf{P}, \mathsf{V})$ as follows. (The protocol Auth inherits the completeness error of Auth'.)

- Key Generation. The key-generation algorithm $\mathsf{Gen}(1^k)$ picks a key $K_{\mathsf{H}} \xleftarrow{\$} \mathcal{K}_{\mathsf{H}}$ for H, an element $K_{\mathbb{F}} \xleftarrow{\$} \mathbb{F} \setminus \{0\}$, and generates a key $K' \xleftarrow{\$} \mathsf{Gen}'(1^k)$ for Auth'. The key is $K = (K', K_{\mathsf{H}}, K_{\mathbb{F}})$.
- Challenge. The challenge is generated by the verifier V as $c \xleftarrow{\$} \mathcal{C}$.

[5] This is w.l.o.g., as we can always represent $\mathcal{T}_2$ as a bit-string $\{0,1\}^t$ for some $t \in \mathbb{N}$ which we associate with $\mathbb{F}_{2^t}$.

– Response. The response $\sigma = (\sigma_1, \sigma_2)$ to challenge $c \in \mathcal{C}$ is computed by the prover P by first running $\tau = (\tau_1, \tau_2) \xleftarrow{\$} \mathsf{P}'_{K'}(c)$ and

$$\sigma = (\sigma_1, \sigma_2) = (\tau_1, \tau_2 \circ K_{\mathbb{F}} + \mathsf{H}_{K_{\mathsf{H}}}(\tau_1)) \in \mathcal{T}_1 \times \mathbb{F}.$$

– Verify. Given challenge c and response $\sigma = (\sigma_1, \sigma_2)$, the verifier V reconstructs

$$\tau = (\tau_1, \tau_2) = (\sigma_1,\ (\sigma_2 - \mathsf{H}_{K_{\mathsf{H}}}(\sigma_1)) \circ {K_{\mathbb{F}}}^{-1})$$

and returns the decision $\{\mathsf{accept}, \mathsf{reject}\} \leftarrow \mathsf{V}'_{K'}(c, \tau)$.

Overhead. We note that our transformation does *not* increase the tag size of the underlying protocol Auth', and thus retains its communication complexity. Moreover, the key length increases by adding $K_{\mathbb{F}}$ and K_{H}. Below, we will show that H can be instantiated with the hash-function construction given in Section 2, and thus these two additional keys consist overall of *three* field elements.

3.3 Security

The following theorem establishes the concrete security of our generic construction. In particular, it says that as long as for sufficiently small δ and ϵ, H is δ-almost pairwise independent and Auth' has both ϵ-right-tag sparsity and is ror-cma-secure, then the construction is s-mim-secure.

Theorem 5 (Security of the generic construction). *Assume that* H *is* δ*-almost universal and that* Auth' *satisfies* ϵ*-right tag sparsity and has completeness error* α*. Then, for all* s-mim*-attackers* A *invoking at most* r *sessions, there exists a* ror-cma*-attack* B *such that*

$$\mathbf{Adv}^{\mathsf{s\text{-}mim}}_{\mathsf{Auth}}(A) \leq r \cdot \left(\mathbf{Adv}^{\mathsf{ror\text{-}cma}}_{\mathsf{Auth}'}(B) + \frac{r}{|\mathcal{C}|} + \epsilon\delta \frac{|\mathbb{F}|}{|\mathbb{F}| - 1} + r \cdot \alpha \right) ,$$

where B *has running time approximately equal to that of* A*, and makes at most* r *queries to its oracle. In other words, if* Auth' *is* (t, r, ϵ)-ror-cma*-secure, then* Auth *is* $(t', r, r \cdot (\epsilon + r/|\mathcal{C}| + \epsilon\delta|\mathbb{F}|/(|\mathbb{F}| - 1)))$-s-mim*-secure, where* $t' \approx t$*.*

Proof. Let A be an attacker for game S-MIM which calls its oracles for at most r sessions. In the following, we are going to upper bound $\mathbf{Adv}^{\mathsf{s\text{-}mim}}_{\mathsf{Auth}}(A) = \Pr\left[\mathsf{S\text{-}MIM}^{A}_{\mathsf{Auth}} \Rightarrow \mathsf{true}\right]$. The proof proceeds via a sequence of games.

As our first step, we prove that it is sufficient to consider the *first* round where the attacker alters the communication between prover and verifier, and the latter still accepts. Formally, for all $\mathsf{sid}^* \in \{1, \ldots, r\}$, let $\textsc{Win}_{\mathsf{sid}^*}$ be the event that in the experiment $\mathsf{S\text{-}MIM}^{A}_{\mathsf{Auth}}$ session sid^* is the first session where the attacker makes the verifier non-trivially accept (and thus $d[\mathsf{sid}^*] = \mathsf{accept}$) with $(c'[\mathsf{sid}^*], t'[\mathsf{sid}^*]) \neq (c[\mathsf{sid}^*], t[\mathsf{sid}^*])$. In particular, for all $\mathsf{sid} < \mathsf{sid}^*$ we either have $(c[\mathsf{sid}], t[\mathsf{sid}]) = (c'[\mathsf{sid}], t'[\mathsf{sid}])$ or $d[\mathsf{sid}] = \mathsf{reject}$. Moreover, let $\textsc{Win} =$

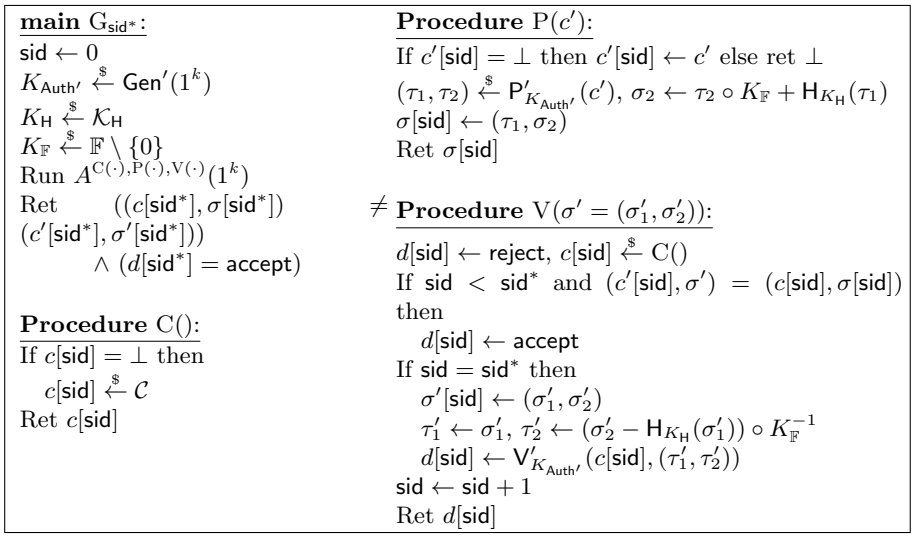

Fig. 1: Game $\mathrm{G}_{\mathsf{sid}^*}$ for $\mathsf{sid}^* \in \{1, \ldots, r\}$ in the proof of Theorem 5. All oracles return $\perp$ if $\mathsf{sid} > \mathsf{sid}^*$.

$\bigcup_{\mathsf{sid}^*=1}^{r} \mathrm{WIN}_{\mathsf{sid}^*}$ be the event that $\mathsf{S\text{-}MIM}^A_{\mathsf{Auth}}$ outputs true in the first place. Clearly, the r events $\mathrm{WIN}_1, \ldots, \mathrm{WIN}_r$ are disjoint, and therefore

$$\Pr[\,\mathrm{WIN}\,] = \Pr\left[\bigcup_{\mathsf{sid}^*=1}^{r} \mathrm{WIN}^*_{\mathsf{sid}}\right] = \sum_{\mathsf{sid}^*=1}^{r} \Pr[\,\mathrm{WIN}^*_{\mathsf{sid}}\,]\,.$$

As our first step, we introduce r new games $\mathrm{G}_1, \ldots, \mathrm{G}_r$, where $\mathrm{G}_{\mathsf{sid}^*}$ only allows the adversary A to execute sid^* sessions, and the verifier returns reject for the first $\mathsf{sid}^* - 1$ sessions unless the adversary A has been simply forwarding honestly generated messages. A formal description of $\mathrm{G}_{\mathsf{sid}^*}$ is given in Figure 1. There, we implicitly assume that all oracles return $\perp$ whenever $\mathsf{sid} > \mathsf{sid}^*$. It is easy to see that by construction, $\Pr\left[\,\mathrm{G}^A_{\mathsf{sid}^*} \Rightarrow \mathsf{true}\,\right] \geq \Pr[\,\mathrm{WIN}_{\mathsf{sid}^*}\,] - (\mathsf{sid}^* - 1)\alpha$. The offset depending on the completeness error α is due to the fact that $\mathrm{G}^A_{\mathsf{sid}^*}$ always accepts honest executions in sessions $\mathsf{sid} < \mathsf{sid}^*$, whereas this is not necessarily true in $\mathsf{S\text{-}MIM}^A_{\mathsf{Auth}}$. Therefore,

$$\Pr\left[\,\mathsf{S\text{-}MIM}^A_{\mathsf{Auth}} \Rightarrow \mathsf{true}\,\right] = \Pr[\,\mathrm{WIN}\,] \leq r^2\alpha + \sum_{\mathsf{sid}^*=1}^{r} \Pr\left[\,\mathrm{G}^A_{\mathsf{sid}^*} \Rightarrow \mathsf{true}\,\right]. \quad (1)$$

In the remainder of this proof, for every $\mathsf{sid}^* \in \{1, \ldots, r\}$, we are going to prove an upper bound on $\Pr\left[\,\mathrm{G}^A_{\mathsf{sid}^*} \Rightarrow \mathsf{true}\,\right]$. In particular, we now fix an arbitrary $\mathsf{sid}^* \in \{1, \ldots, r\}$, and let $\mathrm{H}_0 = \mathrm{G}_{\mathsf{sid}^*}$.

The proof now continues by transitioning from Game H_0 in turn to games H_1, H_2 and H_3. With respect to H_0, these games will only differ in the way in which queries to P are answered, but all games will otherwise inherit the **main** procedure, as well as C and V, verbatim from $\mathrm{G}_{\mathsf{sid}^*} = \mathrm{H}_0$. A formal specification

Procedure P(c'): // H_1	**Procedure** P(c'): // H_2	**Procedure** P(c'): // H_3
If $c'[\mathsf{sid}] = \perp$ then	If $c'[\mathsf{sid}] = \perp$ then	If $\mathsf{sid} < \mathsf{sid}^*$ and $c' = c[\mathsf{sid}^*]$
$\quad c'[\mathsf{sid}] \leftarrow c'$	$\quad c'[\mathsf{sid}] \leftarrow c'$	$\quad$ then Ret $\perp$
Else Ret $\perp$	Else Ret $\perp$	If $c'[\mathsf{sid}] = \perp$ then $c'[\mathsf{sid}] \leftarrow c'$
$(\tau_1, \tau_2) \xleftarrow{\$} \mathsf{P}'_{K_{\mathsf{Auth}'}}(c')$	$(\tau_1, \tau_2) \xleftarrow{\$} \mathsf{P}'_{K_{\mathsf{Auth}'}}(c')$	Else Ret $\perp$
If $c' \neq c[\mathsf{sid}^*]$ then	If $c' = c[\mathsf{sid}^*]$ then	$(\tau_1, \tau_2) \xleftarrow{\$} \mathsf{P}'_{K_{\mathsf{Auth}'}}(c')$
$\quad \tau_2 \xleftarrow{\$} \mathcal{T}_2$	$\quad \sigma_2 \leftarrow \tau_2 \circ K_{\mathbb{F}} + \mathsf{H}_{K_{\mathsf{H}}}(\tau_1)$	If $c' = c[\mathsf{sid}^*]$ then
$\sigma_2 \leftarrow \tau_2 \circ K_{\mathbb{F}} + \mathsf{H}_{K_{\mathsf{H}}}(\tau_1)$	Else $\sigma_2 \xleftarrow{\$} \mathcal{T}_2$	$\quad \sigma_2 \leftarrow \tau_2 \circ K_{\mathbb{F}} + \mathsf{H}_{K_{\mathsf{H}}}(\tau_1)$
$\sigma[\mathsf{sid}] \leftarrow (\tau_1, \sigma_2)$	$\sigma[\mathsf{sid}] \leftarrow (\tau_1, \sigma_2)$	Else $\sigma_2 \xleftarrow{\$} \mathcal{T}_2$
Ret $\sigma[\mathsf{sid}]$	Ret $\sigma[\mathsf{sid}]$	$\sigma[\mathsf{sid}] \leftarrow (\tau_1, \sigma_2)$
		Ret $\sigma[\mathsf{sid}]$

Fig. 2: Modified prover oracles in the games H_1, H_2, and H_3.

of the respective procedures is given in Figure 2, and we now discuss them in detail.

We first transition to Game H_1, where we will use ror-cma security of Auth' to replace the right half of every tag computed by P to a random component whenever $c' \neq c[\mathsf{sid}^*]$, i.e., different from the random challenge used in the last round. The proof of the following lemma is given below.

Lemma 6. *There exists an attacker B such that*

$$\Pr\left[\,\mathrm{H}_0^A \Rightarrow \mathsf{true}\,\right] - \Pr\left[\,\mathrm{H}_1^A \Rightarrow \mathsf{true}\,\right] \leq \mathbf{Adv}^{\mathsf{ror\text{-}cma}}_{\mathsf{Auth}'}(B)\ ,$$

where B has running time approximately equal to that of A, and makes at most r queries to its oracle.

Subsequently, in Game H_2, whenever $c' \neq c[\mathsf{sid}^*]$, instead of generating τ_2 at random, we directly generate σ_2 uniformly at random from the same set. Note that because $K_{\mathbb{F}} \neq 0$, we have that $\tau_2 \cdot K_{\mathbb{F}}$ is a fresh random value, and thus the two games H_1 and H_2 are identical,

In the next game, Game H_3, the procedure P replies to a query $c' = c[\mathsf{sid}^*]$ only if it is made in session sid^*, and otherwise returns $\perp$. As $c[\mathsf{sid}^*]$ is chosen uniformly at random, and independent of the interaction between the adversary and the oracles in the first $\mathsf{sid}^* - 1$ sessions, the "fundamental lemma" of game playing [4] yields

$$\begin{aligned}\Pr\left[\,\mathrm{H}_2^A \Rightarrow \mathsf{true}\,\right] - \Pr\left[\,\mathrm{H}_3^A \Rightarrow \mathsf{true}\,\right] \\ \leq \Pr\left[\,c[\mathsf{sid}^*] \in \{c'[1], c'[2], \ldots, c'[\mathsf{sid}^* - 1]\}\,\right] \leq \frac{r}{|\mathcal{C}|}\ . \quad (2)\end{aligned}$$

Therefore, putting together Equation (1), Lemma 6, and Equation (2), we obtain that there exists an attacker B making at most r oracle queries and with time complexity close to the one of A such that

$$\Pr\left[\,\mathsf{S\text{-}MIM}^A_{\mathsf{Auth}} \Rightarrow \mathsf{true}\,\right] \leq r \cdot \left(\mathbf{Adv}^{\mathsf{ror\text{-}cma}}_{\mathsf{Auth}'}(B) + \frac{r}{|\mathcal{C}|} + \Pr\left[\,\mathrm{H}_3^A \Rightarrow \mathsf{true}\,\right]\right) + r^2\alpha\ .$$

In the rest of the proof, we give an upper bound on the probability that the game H_3 outputs true. The argument is going to rely on the almost pairwise-independence of H and the right-tag sparsity of Auth', and is from now on a purely information-theoretic argument. In particular, it does not rely on $K_{\mathsf{Auth}'}$ being hidden, but only on the fact that all the right tags in sessions prior to sid^* are random.

Analysis of winning probability in H_3. In the following, for notational convenience we let $\sigma[\mathsf{sid}^*] = (\sigma_1 = \tau_1, \sigma_2)$ and $\sigma'[\mathsf{sid}^*] = (\sigma'_1, \sigma'_2)$ be the original and modified values in the second-round of session sid^*. Similarly, we simply denote $c = c[\mathsf{sid}^*]$ and $c' = c'[\mathsf{sid}^*]$. Concretely, we are going to consider three different cases when analyzing the probability $\Pr\left[\,\mathrm{H}_3^A \Rightarrow \mathsf{true}\,\right]$: (1) $c' \neq c$, (2) $c' = c$ and $\sigma_1 = \sigma'_1$, and (3) $c' = c$ and $\sigma_1 \neq \sigma'_1$. We now analyze the three individual cases.

Case $c' \neq c$. Observe first that in session sid^*, the attacker obtains (τ_1, σ_2), where $(\tau_1, \tau_2) \xleftarrow{\$} \mathsf{P}'_K(c'_{\mathsf{sid}^*})$ and $\sigma_2 \xleftarrow{\$} \mathcal{T}_2$, and inputs (σ'_1, σ'_2) to V. It wins if (σ'_1, τ'_2) is a valid tag, where $\tau'_2 = (\sigma'_2 - \mathsf{H}_{K_\mathsf{H}}(\sigma'_1)) \circ K_\mathbb{F}^{-1}$.

The crucial point is that $K_\mathbb{F}$ and K_H have *never* been used prior to the computation of τ'_2, as the oracle P has only returned random right tags. So we can equivalently think of generating these uniformly at random for the first time at this point independent of the rest of the game, and consider the probability that $\mathsf{V}'_K(c, (\sigma'_1, \tau'_2))$ verifies. Moreover, the value $Y := \mathsf{H}_{K_\mathsf{H}}(\sigma'_1)$ is going to be uniform (as we don't evaluate the function on any other point) by the δ-almost pairwise independence of H. Therefore, for every value $t \in \mathbb{F}$,

$$\Pr\left[\,\tau'_2 = t\,\right] = \Pr\left[\,(\sigma'_2 - Y) \circ K_\mathbb{F}^{-1} = t\,\right] = \Pr\left[\,Y = K_\mathbb{F} \circ t + \sigma'_2\,\right] = \frac{1}{|\mathbb{F}|}\,.$$

However, by ϵ-right tag sparsity, we know that there are at most $\epsilon|\mathbb{F}|$ possible values t for which (σ'_1, t) is a valid tag, and thus by the union bound

$$\Pr\left[\,\mathrm{H}_3^A \Rightarrow \mathsf{true} \mid\ c' \neq c\,\right] = \Pr\left[\,\mathsf{V}'_K(c, (\sigma'_1, \tau'_2)) = \mathsf{accept}\,\right] \leq \epsilon \cdot |\mathbb{F}| \cdot \frac{1}{|\mathbb{F}|} = \epsilon\,. \quad (3)$$

Case $c' = c$, $\sigma_1 = \sigma'_1 = \tau_1$ and $\sigma_2 \neq \sigma'_2$. In this case, in session sid^*, the attacker obtains (τ_1, σ_2), where $(\tau_1, \tau_2) \xleftarrow{\$} \mathsf{P}'_K(c)$ and $\sigma_2 \xleftarrow{\$} \tau_2 \circ K_\mathbb{F} + \mathsf{H}_{K_\mathsf{H}}(\tau_1)$. Subsequently, it inputs (τ_1, σ'_2) to V. It wins if $\mathsf{V}'_K(c, (\tau_1, \tau'_2)) = \mathsf{accept}$, where $\tau'_2 = (\sigma'_2 - \mathsf{H}_{K_\mathsf{H}}(\tau_1)) \circ K_\mathbb{F}^{-1} \neq \tau_2$. Once again, we evaluate H only with one input, and as above $Y = \mathsf{H}_{K_\mathsf{H}}(\tau_1)$ is uniformly distributed.

Now, given σ_2, τ_2, and σ'_2, we want to upper bound the probability that $\tau'_2 = t \neq \tau_2$ for some value $t \in \mathbb{F}$, where the probability is over the choice of $K_\mathbb{F}$ and Y.

$$\Pr\left[\,\tau'_2 = t \mid\ \sigma_2 = \tau_2 \circ K_\mathbb{F} + Y\,\right] = \frac{\Pr\left[\,t \circ K_\mathbb{F} + Y = \sigma'_2 \wedge \tau_2 \circ K_\mathbb{F} + Y = \sigma_2\,\right]}{\Pr\left[\,\tau_2 \circ K_\mathbb{F} + Y = \sigma_2\,\right]}\,.$$

Since $\tau_2 \neq t$, there exists exactly one $K_\mathbb{F}$ such that $(\tau_2 - t) \cdot K_\mathbb{F} = \sigma_2 - \sigma'_2$, and moreover, this defines a unique value for Y, which is taken with probability

at most $1/|\mathbb{F}|$, and thus the probability in the numerator is upper bounded by $1/(|\mathbb{F}|(|\mathbb{F}|-1))$. Moreover, $\tau_2 \circ K_{\mathbb{F}} + Y$ is clearly uniform (because Y is uniform), and thus the denominator is $1/|\mathbb{F}|$. Putting these together gives us $\Pr[\,\tau_2' = t \mid \sigma_2 = \tau_2 \circ K_{\mathbb{F}} + Y\,] \leq 1/(|\mathbb{F}|-1)$. Now, due to ϵ-right tag sparsity, there are at most $\epsilon \cdot |\mathbb{F}|$ right tags τ_2' that verify, and thus

$$\Pr\left[\,\mathrm{H}_3^A \Rightarrow \mathsf{true} \mid\ c' = c \wedge \sigma_1 = \sigma_1'\,\right] \leq \epsilon \cdot \frac{|\mathbb{F}|}{|\mathbb{F}|-1}\ . \tag{4}$$

Case $c' = c$ and $\sigma_1 \neq \sigma_1'$. For the final case, the attacker obtains (τ_1, σ_2) as in the previous case, but inputs $(\sigma_1' \neq \tau_1, \sigma_2')$ to V, and the latter computes $\tau_2' = (\sigma_2' - \mathsf{H}_{K_\mathsf{H}}(\sigma_1')) \circ K_{\mathbb{F}}^{-1}$.

Here, we indeed evaluate $\mathsf{H}_{K_\mathsf{H}}$ on *two* inputs. However, by Lemma 2, we see that for every possible values σ_1' and y', chosen adaptively depending on τ_1 and $\mathsf{H}_{K_\mathsf{H}}(\tau_1)$, $\mathsf{H}_{K_\mathsf{H}}(\sigma_1') = y'$ with probability at most $\delta/|\mathbb{F}|$. Therefore, for every possible t such that $\mathsf{V}'_K(c, (\sigma_1', t)) = \mathsf{accept}$, we have

$$\Pr[\,\tau_2' = t\,] = \Pr[\,\mathsf{H}_{K_\mathsf{H}}(\sigma') = K_{\mathbb{F}} \cdot t + \sigma_2'\,] \leq \delta/|\mathbb{F}|.$$

Now, due to ϵ-right tag sparsity, there are at most $\epsilon \cdot |\mathbb{F}|$ such right tags, and thus

$$\Pr\left[\,\mathrm{H}_3^A \Rightarrow \mathsf{true} \mid\ c' = c \wedge \sigma_1 \neq \sigma_1'\,\right] \leq \epsilon \cdot |\mathbb{F}| \cdot \frac{\delta}{|\mathbb{F}|} = \epsilon\delta\ . \tag{5}$$

Putting things together. To conclude the proof, we observe that all terms in Equations (3), (4) and (5) are upper bounded by $\epsilon \cdot \delta \cdot \frac{|\mathbb{F}|}{|\mathbb{F}|-1}$, and thus we also have $\Pr\left[\,\mathrm{H}_3^A \Rightarrow \mathsf{true}\,\right] \leq \epsilon \cdot \delta \cdot \frac{|\mathbb{F}|}{|\mathbb{F}|-1}$. □

Proof (Lemma 6). The attacker B for $\mathsf{ROR\text{-}CMA}(b)$ is very simple. It simulates the execution of H_b to the attacker A. Initially, B uses its input challenge c^* as c_{sid^*}. Then, when simulating queries to P on input c', it forwards them to its own oracle T, to obtain a pair (τ_1, τ_2). Finally, B uses the one available verification query to compute V's decision bit in session sid^*. Finally, B outputs the games H_b's output. By inspection, it is not hard to verify that $\Pr\left[\,\mathsf{ROR\text{-}CMA}(b)^B_{\mathsf{Auth}'} \Rightarrow \mathsf{true}\,\right] = \Pr\left[\,\mathrm{H}_b^A \Rightarrow \mathsf{true}\,\right]$, which concludes the proof of the lemma. □

4 Instantiations

In this section, we will provide examples of ror-cma-secure authentication protocols. All of them can be transformed to s-mim-secure authentication protocols using the transformation from Section 3. Table 2 summarizes the resulting protocols compactly.

Scheme	Assump.	$\mathsf{Gen}(1^n)$ / Response $\mathsf{P}(c)$ / Verify $\mathsf{V}(c,\sigma)$
$\mathsf{Auth}_{\mathsf{LPN}}$ §4.1	LPN	$\mathsf{Gen}(1^n)$: $(\mathbf{x}_1,\ldots,\mathbf{x}_5) \xleftarrow{\$} (\mathbb{F}_2^\ell)^5$ $\mathsf{P}(\mathbf{c}) = (\mathbf{R}, \mathbf{R}\cdot(\mathbf{M_c}\cdot\mathbf{x}_1+\mathbf{x}_2)+\mathbf{x}_3\circ\mathbf{e}+\mathsf{H}_{\mathbf{x}_4,\mathbf{x}_5}(\mathbf{R}) \in \mathbb{F}_2^{\ell\times n}\times\mathbb{F}_2^\ell$ $\mathsf{V}(\mathbf{c},(\mathbf{R},\mathbf{z}))$: $\lvert(\mathbf{z}-\mathsf{H}_{\mathbf{x}_4,\mathbf{x}_5}(\mathbf{R})-\mathbf{R}\cdot(\mathbf{M_c}\cdot\mathbf{x}_1+\mathbf{x}_2))\circ\mathbf{x}_3^{-1}\rvert$ small?
$\mathsf{AuthT}_{\mathsf{LPN}}$ §4.1	LPN	$\mathsf{Gen}(1^n)$: $(\mathbf{X}_1,\mathbf{X}_2,\mathbf{x}_3,\mathbf{x}_4) \xleftarrow{\$} (\mathbb{F}_2^{n\times\ell})^2\times(\mathbb{F}_2^\ell)^2$ $\mathsf{P}(\mathbf{c}) = (\mathbf{r}, (\mathbf{X}_1\cdot\mathbf{M_c}+\mathbf{X}_2)\cdot\mathbf{r}+\mathbf{x}_3\circ\mathbf{e}+\mathbf{x}_4) \in \mathbb{F}_2^\ell\times\mathbb{F}_2^n$ $\mathsf{V}(\mathbf{c},(\mathbf{r},\mathbf{z}))$: $\lvert(\mathbf{z}-\mathbf{x}_4-(\mathbf{X}_1\cdot\mathbf{M_c}+\mathbf{X}_2)\cdot\mathbf{r})\circ\mathbf{x}_3^{-1}\rvert$ small?
$\mathsf{Auth}_{\mathsf{Field\text{-}LPN}}$ §4.2	Field-LPN	$\mathsf{Gen}(1^n)$: $(\mathbf{x}_1,\ldots,\mathbf{x}_4) \xleftarrow{\$} (\mathbb{F}_{2^\ell})^4$ $\mathsf{P}(\mathbf{c}) = (\mathbf{r}, \mathbf{r}\circ(\mathbf{x}_1\circ\mathbf{c}+\mathbf{x}_2)+\mathbf{x}_3\circ\mathbf{e}+\mathbf{x}_4) \in \mathbb{F}_{2^\ell}\times\mathbb{F}_{2^\ell}$ $\mathsf{V}(\mathbf{c},(\mathbf{R},\mathbf{z}))$: $\lvert(\mathbf{z}-\mathbf{x}_4-\mathbf{r}\circ(\mathbf{x}_1\circ\mathbf{c}+\mathbf{x}_2))\circ\mathbf{x}_3^{-1}+\mathbf{x}_4)\rvert$ small?
$\mathsf{Auth}_{\mathsf{ddh}}$ §4.4	ddh	$\mathsf{Gen}(1^n)$: $(x_1,x_2,X) \xleftarrow{\$} \mathbb{F}_q^2\times\mathbb{G}$ $\mathsf{P}(c) = (R, X\cdot R^{x_1c+x_2}) \in \mathbb{G}\times\mathbb{G}$ $\mathsf{V}(c,(r,z))$: $X\cdot R^{x_1c+x_2} = z$?
$\mathsf{Auth}_{\mathsf{wprf}}$ §4.3	wprf	$(x_{0,0},\ldots,x_{\ell,1},\mathbf{x}_1,\mathbf{x}_2) \xleftarrow{\$} \mathbb{D}^{2\ell}\times\mathbb{F}^2$ $\mathsf{P}(c) = (r, \sum_{i=1}^{\ell} F(x_{i,c_i},r)+\mathsf{H}_{\mathbf{x}_1,\mathbf{x}_2}(r)) \in \mathbb{D}\times\mathbb{F}$ $\mathsf{V}(c,(r,z))$: $\sum_{i=1}^{\ell} F(x_{i,c_i},r)+\mathsf{H}_{\mathbf{x}_1,\mathbf{x}_2}(r) = z$?

Table 2: New s-mim-secure 2-round authentication protocols.

4.1 Instantiations from LPN

Learning Parity with Noise. For a parameter $0 < \gamma \le 1/2$, we define the Bernoulli distribution $\mathcal{B}_\gamma$ that assigns $e \xleftarrow{\$} \mathcal{B}_\gamma$ the values 1 and 0 with probabilities γ and $1-\gamma$, respectively. If $\mathcal{D}$ is a distribution over $\mathbb{D}$, then $\mathbf{x} \xleftarrow{\$} \mathcal{D}^n$ denotes the n-fold distribution where each component of $\mathbf{x} \in \mathbb{D}^n$ is chosen according to $\mathcal{D}$.

To define the $\mathsf{LPN}_{\ell,\gamma}$ problem in dimension $\ell \in \mathbb{N}$ and Bernoulli parameter $0 < \gamma \le 1/2$ we introduce the LPN advantage as the quantity

$$\mathbf{Adv}^{\mathsf{LPN}}(A) = \Pr\left[A^{\mathrm{LPN}_{\mathbf{s},\gamma}()} \Rightarrow \mathsf{true}\right] - \Pr\left[A^{\mathrm{LPN}_{\mathbf{s},1/2}()} \Rightarrow \mathsf{true}\right],$$

where $\mathbf{s} \xleftarrow{\$} \mathbb{F}_2^\ell$ and $\mathrm{LPN}_{\mathbf{s},\alpha}$ ($\alpha \in \{\gamma, 1/2\}$) returns $(\mathbf{r}, \mathbf{r}^T\cdot\mathbf{s}+e)$ for $\mathbf{r} \xleftarrow{\$} \mathbb{F}_2^\ell$ and $e \xleftarrow{\$} \mathcal{B}_\alpha$. Note that oracle $\mathrm{LPN}_{\mathbf{s},1/2}$ always returns uniform $(\mathbf{r},z) \xleftarrow{\$} \mathbb{F}_2^\ell\times\mathbb{F}_2$. We say that the $\mathsf{LPN}_{\ell,\gamma}$ is (t,q,ϵ)-hard if for all attackers A with time complexity t, making at most q oracle queries, we have $\mathbf{Adv}^{\mathsf{LPN}}(A) \le \epsilon$.

ROR-CMA secure Protocol. Let $n = O(\ell)$ denote the number of repetitions, γ the parameter of the Bernoulli distribution, and $\gamma' := 1/4 + \gamma/2$ controls the correctness error. The following authentication protocol $\mathsf{Auth}'_{\mathsf{LPN}} = \{\mathsf{Gen}', \mathsf{P}', \mathsf{V}'\}$ originates from [20]. It has associated key space $\mathcal{K} = (\mathbb{F}_2^\ell)^2$, tag space $\mathcal{T} = \mathcal{T}_1\times\mathcal{T}_2 = \mathbb{F}_2^{\ell\times n}\times\mathbb{F}_2^n$, and challenge space $\mathcal{C} = \mathbb{F}_2^\ell$.

- Key Generation. The key-generation algorithm Gen' outputs a secret key $K = (\mathbf{k}_1,\mathbf{k}_2) \xleftarrow{\$} (\mathbb{F}_2^\ell)^2$.
- Challenge. The challenge is generated by the verifier V' as $\mathbf{c} \xleftarrow{\$} \mathbb{F}_2^\ell$.
- Response. The response $\tau = (\tau_1,\tau_2)$ to challenge $\mathbf{c} \in \mathbb{F}_2^\ell$ is computed by the prover P' by sampling $\mathbf{R} \xleftarrow{\$} \mathbb{F}_2^{\ell\times n}$ and computing $\tau = (\mathbf{R}, \mathbf{R}^T\cdot(\mathbf{M_c}\cdot\mathbf{k}_1 +$

$\mathbf{k}_2) + \mathbf{e}$), where $\mathbf{e} \xleftarrow{\$} \mathcal{B}_\gamma^n$. (Recall that $\mathbf{M_c}$ is the matrix representation of the finite field multiplication with $\mathbf{c}$.)

– Verification. Given challenge $\mathbf{c} \in \mathbb{F}_2^\ell$ and response $\tau = (\mathbf{R}, \mathbf{z}) \in \mathbb{F}_2^{\ell \times n} \times \mathbb{F}_2^n$, the verifier V' outputs accept iff: $rank(\mathbf{R}) = n$ and $|\mathbf{R}^T \cdot (\mathbf{M_c} \cdot \mathbf{k}_1 + \mathbf{k}_2) - \mathbf{z}| \leq \gamma' n$.

With the choice of $\gamma' = 1/4 + \gamma/2$, $\mathsf{Auth}'_{\mathsf{LPN}}$ has $2^{-O(n)}$ completeness error [20, Th. 4]. Further, it has ϵ-sparse right tags, where $\epsilon = \Pr[\, \mathbf{z} \leq \gamma' n \mid \mathbf{z} \xleftarrow{\$} \mathbb{F}_2^n \,] \leq 2^{-O(n)}$, using the Hoeffding bound.

The proof of the following theorem is postponed to Appendix A.2.

Theorem 7. *If* $\mathsf{LPN}_{\ell,\gamma}$ *is* (t, nq, ϵ)*-hard, then* $\mathsf{Auth}'_{\mathsf{LPN}}$ *is* (t', q, ϵ)-ror-cma*-secure with* $t \approx t'$.

There exists an alternative ror-cma-secure authentication protocol [10, 20] which defines $\tau_2 = \mathbf{R}^T \cdot \mathbf{k}_{\downarrow \mathbf{c}} + \mathbf{e}$, where $\mathbf{k}_{\downarrow \mathbf{c}}$ is the projection of $\mathbf{k}$ with respect to all ℓ non-zero bits of $\mathbf{c} \in \mathcal{C} := \{\mathbb{F}_2^{2\ell} \ : \ |\mathbf{c}| = \ell\}$.

MiM secure Protocol. A s-mim-secure 2-round authentication protocol $\mathsf{Auth}_{\mathsf{LPN}}$ is obtained via the generic transformation from Section 3. An example instantiation using the almost pairwise independent hash function from Section 2 is given in Table 2.

Trade-off. For all LPN-based protocols there exists a natural trade-off between key-size and communication complexity, as we will explain now. In the ror-cma-secure protocol $\mathsf{AuthT}'_{\mathsf{LPN}}$ we can chose the key as $(\mathbf{K}_1, \mathbf{K}_2) \xleftarrow{\$} (\mathbb{Z}_2^{\ell \times n})^2$ and define the response to a challenge $\mathbf{c} \in F_2^\ell$ as $(\mathbf{r}, (\mathbf{M_c} \cdot \mathbf{K}_1 + \mathbf{K}_2) \cdot \mathbf{r} + \mathbf{e} \in \mathbb{F}_2^\ell \times \mathbb{F}_2^n$, where $\mathbf{r} \xleftarrow{\$} \mathbb{F}_2^\ell$. In the resulting s-mim-secure protocol we can use the specific pairwise independent hash function $\mathsf{H}_{\mathbf{S}_1, \mathbf{s}_2}(\mathbf{r}) := \mathbf{S}_1 \mathbf{r} + \mathbf{s}_2$, where $(\mathbf{S}_1, \mathbf{s}_2) \in \mathbb{F}_2^{\ell \times n} \times \mathbb{F}_2^n$. The response to a challenge $\mathbf{c}$ is computed as $\sigma = (\mathbf{r}, \mathbf{z})$, where

$$\begin{aligned} \mathbf{z} &= ((\mathbf{M_c} \cdot \mathbf{K}_1 + \mathbf{K}_2) \cdot \mathbf{r} + \mathbf{e}) \circ K_{\mathbb{F}} + \mathbf{S}_1 \mathbf{r} + \mathbf{s}_2 \\ &= (\mathbf{M_c} \cdot \mathbf{K}_1 \cdot \mathbf{M}_{K_{\mathbb{F}}} + \mathbf{K}_2 \cdot \mathbf{M}_{K_{\mathbb{F}}} + \mathbf{S}_1) \cdot \mathbf{r} + \mathbf{M}_{K_{\mathbb{F}}} \cdot \mathbf{e} + \mathbf{s}_2. \end{aligned}$$

This can be rewritten as $\mathbf{z} = (\mathbf{M_c} \cdot \mathbf{X}_1 + \mathbf{X}_2) \cdot \mathbf{r} + \mathbf{e} \circ \mathbf{x}_3 + \mathbf{x}_4$ using the substitutions

$$\mathbf{x}_1 := \mathbf{K}_1 \cdot \mathbf{M}_{K_{\mathbb{F}}}, \quad \mathbf{X}_2 := \mathbf{K}_2 \cdot \mathbf{M}_{K_{\mathbb{F}}} + \mathbf{S}_1, \quad \mathbf{x}_3 := K_{\mathbb{F}}, \quad \mathbf{x}_4 := \mathbf{s}_2.$$

The resulting protocol $\mathsf{AuthT}_{\mathsf{LPN}}$ is described in Table 2.

4.2 Instantiations from Field-LPN

Field Learning Parity with Noise. To define the $\mathsf{Field\text{-}LPN}_{\ell,\gamma}$ problem over the extension field $(\mathbb{F}_{2^\ell}, \circ, +)$ and Bernoulli parameter $0 < \gamma \leq 1/2$, we introduce the Field-LPN advantage as the quantity

$$\mathbf{Adv}^{\mathsf{Field\text{-}LPN}}(A) = \Pr\left[A^{\mathrm{FLPN}_{\mathbf{s},\gamma}()} \Rightarrow \mathsf{true} \right] - \Pr\left[A^{\mathrm{FLPN}_{\mathbf{s},1/2}()} \Rightarrow \mathsf{true} \right],$$

where $\mathbf{s} \xleftarrow{\$} \mathbb{F}_{2^\ell}$ and $\mathrm{FLPN}_{\mathbf{s},\alpha}$ returns $(\mathbf{r}, \mathbf{r} \circ \mathbf{s} + \mathbf{e})$ for $\mathbf{r} \xleftarrow{\$} \mathbb{F}_{2^\ell}$ and $\mathbf{e} \xleftarrow{\$} \mathcal{B}_\alpha^\ell$. Note that $\mathrm{FLPN}_{\mathbf{s},1/2}$ always returns uniform $(\mathbf{r}, \mathbf{z}) \xleftarrow{\$} (\mathbb{F}_{2^\ell})^2$. We say that the

$\mathsf{Field\text{-}LPN}_{\ell,\gamma}$ is (t, q, ϵ)-hard if for all attackers A with time complexity t making at most q oracle queries, we have $\mathbf{Adv}^{\mathsf{Field\text{-}LPN}}(A) \leq \epsilon$.

ROR-CMA secure Protocol. Let γ the parameter of the Bernoulli distribution, and $\gamma' := 1/4 + \gamma/2$ controls the correctness error. We use $\mathbb{F} = \mathbb{F}_{2^\ell}$ to denote the finite field. The following authentication protocol $\mathsf{Auth}'_{\mathsf{Field\text{-}LPN}} = \{\mathsf{Gen}', \mathsf{P}', \mathsf{V}'\}$ originates from [15]. It has associated key space $\mathcal{K} = \mathbb{F}^2$, split tag space $\mathcal{T} = \mathcal{T}_1 \times \mathcal{T}_2 = \mathbb{F} \times \mathbb{F}$, and challenge space $\mathcal{C} = \mathbb{F}$.

- Key Generation. The key-generation algorithm Gen' outputs a secret key $K = (\mathbf{k}_1, \mathbf{k}_2) \xleftarrow{\$} \mathbb{F}^2$.
- Challenge. The challenge is generated by the verifier V' as $\mathbf{c} \xleftarrow{\$} \mathbb{F}$.
- Response. The response $\tau = (\tau_1, \tau_2)$ to challenge $c \in \mathbb{F}$ is computed by the prover P' as $\tau = (\mathbf{r}, \mathbf{r} \circ (\mathbf{k}_1 \circ \mathbf{c} + \mathbf{k}_2) + \mathbf{e})$, where $\mathbf{r} \xleftarrow{\$} \mathbb{F}$, $\mathbf{e} \xleftarrow{\$} \mathcal{B}_\gamma^\ell$.
- Verification. Given challenge $c \in \mathbb{F}$ and response $\tau = (\mathbf{r}, \mathbf{z}) \in \mathbb{F}^2$, the verifier V' outputs accept iff $|\mathbf{r} \circ (\mathbf{k}_1 \circ \mathbf{c} + \mathbf{k}_2) - \mathbf{z}| \leq \gamma' n$.

As in the LPN case, $\mathsf{Auth}'_{\mathsf{Field\text{-}LPN}}$ has $2^{-O(\ell)}$ completeness error and $2^{-O(\ell)}$-sparse right tags. The proof of the following theorem is similar to that of Theorem 7 and is therefore omitted.

Theorem 8. *If* $\mathsf{Field\text{-}LPN}_{\ell,\gamma}$ *is* (t, q, ϵ)*-hard, then* $\mathsf{Auth}'_{\mathsf{Field\text{-}LPN}}$ *is* (t', q, ϵ)-ror-cma*-secure with* $t' \approx t$.

MiM secure Protocol. We now apply our generic transformation from Section 3 to $\mathsf{Auth}'_{\mathsf{Field\text{-}LPN}}$ to obtain a s-mim-secure protocol. The key consists of $(\mathbf{k}_1, \mathbf{k}_2, K_\mathbb{F}, \mathbf{s}_1, \mathbf{s}_2)$, where we use the concrete pairwise-independent hash function $\mathsf{H}_{\mathbf{s}_1,\mathbf{s}_2}(\mathbf{r}) = \mathbf{s}_1 \circ \mathbf{r} + \mathbf{s}_2$. The response to a challenge $\mathbf{c}$ is computed as $\sigma = (\mathbf{r}, \mathbf{z})$, where $\mathbf{z} = (\mathbf{r} \circ (\mathbf{k}'_1 \circ \mathbf{c} + \mathbf{k}'_2) + \mathbf{e}) \circ K_\mathbb{F} + \mathbf{s}_1 \circ \mathbf{r} + \mathbf{s}_2$
$= (\mathbf{r} \circ (\mathbf{k}'_1 \circ K_\mathbb{F} \circ \mathbf{c} + \mathbf{k}'_2 \circ K_\mathbb{F} + \mathbf{s}_1) + \mathbf{e} \circ K_\mathbb{F} + \mathbf{s}_2$. This can be written as $\mathbf{z} = (\mathbf{r} \circ (\mathbf{x}_1 \circ \mathbf{c} + \mathbf{x}_2) + \mathbf{e} \circ \mathbf{x}_3 + \mathbf{x}_4$ using the substitutions $\mathbf{x}_1 := \mathbf{k}_1 \circ K_\mathbb{F}$, $\mathbf{x}_2 := \mathbf{k}_2 \circ K_\mathbb{F} + \mathbf{s}_1$, $\mathbf{x}_3 := K_\mathbb{F}$, $\mathbf{x}_4 := \mathbf{s}_2$. The resulting simplified protocol $\mathsf{Auth}_{\mathsf{Field\text{-}LPN}}$ is given in Table 2.

4.3 Instantiations from weak PRFs

Weak Pseudorandom Function. Let $\mathcal{F}$ be a function family $F : \mathbb{K} \times \mathbb{D} \to \mathbb{F}$. To define the $\mathsf{wprf}_\mathcal{F}$ assumption over function family $\mathcal{F}$ we introduce the wprf advantage of an adversary A as the quantity

$$\mathbf{Adv}^{\mathsf{wprf}}_\mathcal{F}(A) = \Pr[\, A^{\mathrm{F}_x()} \Rightarrow \mathsf{true}\,] - \Pr[\, A^{\mathrm{U}()} \Rightarrow \mathsf{true}\,]\,,$$

where $x \xleftarrow{\$} \mathbb{K}$, F_x returns $(r, F(x, r))$ for $r \xleftarrow{\$} \mathbb{D}$, and U returns uniform $(r, z) \xleftarrow{\$} \mathbb{D} \times \mathbb{F}$. We say that $\mathcal{F}$ is a (t, q, ϵ)-weak PRF if for all attackers A with time complexity t, making at most q oracle queries, we have $\mathbf{Adv}^{\mathsf{wprf}}_\mathcal{F}(A) \leq \epsilon$.

ROR-CMA secure Protocol. We define an authentication protocols $\mathsf{Auth}'_{\mathsf{wprf}} = \{\mathsf{Gen}', \mathsf{P}', \mathsf{V}'\}$ with associated key space $\mathcal{K} = \mathbb{K}^\ell$, split tag space $\mathcal{T} = \mathcal{T}_1 \times \mathcal{T}_2 = \mathbb{D} \times \mathbb{F}$, and challenge space $\mathcal{C} = \{0,1\}^\ell$.

- Key Generation. The key-generation algorithm Gen' outputs a secret key $K = (x_{1,0}, \ldots, x_{\ell,0}, x_{1,1}, \ldots, x_{\ell,1}) \xleftarrow{\$} \mathbb{K}^{2\times\ell}$.
- Challenge. The challenge is generated by the verifier V' as $c \xleftarrow{\$} \{0,1\}^\ell$.
- Response. The response $\tau = (\tau_1, \tau_2)$ to challenge $c \in \{0,1\}^\ell$ is computed by the prover P' as $\tau = (r, z = \sum_{i=1}^{\ell} F(x_{i,c_i}, r))$, where $r \xleftarrow{\$} \mathbb{D}$.
- Verification. Given challenge $c \in \{0,1\}^\ell$ and response $\tau = (r, z) \in \mathbb{D} \times \mathbb{F}$, the verifier V' outputs accept iff $\sum_{i=1}^{\ell} F(x_{i,c_i}, r) = z$.

The protocol has perfect completeness and $1/|\mathbb{F}|$-sparse right tags. It is easy to extend $\mathsf{Auth}'_{\mathsf{wprf}}$ to *randomized* weak PRFs (with additive noise), as defined in [22]. This way we obtain protocols from a more general class of assumptions, such as Toeplitz-LPN [22]. The proof of the following theorem is in Appendix A.2.

Theorem 9. *If $\mathcal{F}$ is a (t, q, ϵ)-weak PRF, then $\mathsf{Auth}'_{\mathsf{wprf}}$ is $(t', q, \epsilon/\ell)$-*ror-cma*-secure with $t' \approx t$.*

4.4 Instantiation from DDH

The DDH Problem. Let $\mathcal{G}$ be a family of groups with $\mathcal{G}_n = (\mathbb{G}, g, p)$, where $\mathbb{G}$ is a cyclic group of prime-order p with $\lceil \log p \rceil = n$ and g generates $\mathbb{G}$. To define the $\mathsf{ddh}_{\mathcal{G}}$ problem over group family $\mathcal{G}$ we introduce the ddh advantage as the quantity

$$\mathbf{Adv}^{\mathsf{ddh}}_{\mathcal{G}}(A) = \Pr\left[A^{\mathrm{DDH}_x()} \Rightarrow \mathsf{true}\right] - \Pr\left[A^{\mathrm{U}()} \Rightarrow \mathsf{true}\right],$$

where $x \xleftarrow{\$} \mathbb{Z}_p$ and DDH_x returns (R, R^x) for $R \xleftarrow{\$} \mathbb{Z}_p$, and U returns uniform $(R, Z) \xleftarrow{\$} \mathbb{G}^2$. We say that $\mathsf{ddh}_{\mathcal{G}}$ is (t, q, ϵ)-hard if for all attackers A with time complexity t making at most q oracle queries, we have $\mathbf{Adv}^{\mathsf{ddh}}(A) \leq \epsilon$. Note that classical ddh hardness is exactly $(t', 1, \epsilon')$-hardness of $\mathsf{ddh}_{\mathcal{G}}$ and by the random self-reducibility of ddh we have that $\mathsf{ddh}_{\mathcal{G}}$ is (t, q, ϵ)-hard iff it is $(t', 1, \epsilon')$-hard with $t \approx t'$ and $\epsilon \approx \epsilon'$.

ROR-CMA secure Protocol. We define an authentication protocol $\mathsf{Auth}'_{\mathsf{ddh}} = \{\mathsf{Gen}', \mathsf{P}', \mathsf{V}'\}$ with associated key space $\mathcal{K} = \mathbb{Z}_p^2$, split tag space $\mathcal{T} = \mathcal{T}_1 \times \mathcal{T}_2 = \mathbb{G} \times \mathbb{G}$, and challenge space $\mathcal{C} = \mathbb{Z}_p$.

- Key Generation. The key-generation algorithm Gen' outputs a secret key $K = (y_1, y_2) \xleftarrow{\$} \mathbb{Z}_p^2$.
- Challenge. The challenge is generated by the verifier V' as $\mathbf{c} \xleftarrow{\$} \mathbb{Z}_p$.
- Response. The response $\tau = (\tau_1, \tau_2)$ to challenge $c \in \mathbb{F}_p$ is computed by the prover P' as $\tau = (R, R^{y_1 \cdot c + y_2})$, where $R \xleftarrow{\$} \mathbb{G}$.
- Verification. Given challenge $c \in \mathbb{Z}_p$ and response $\tau = (R, Z) \in \mathbb{G}^2$, the verifier V' outputs accept iff $R^{y_1 \cdot c + y_2} = Z$.

The protocol $\mathsf{Auth}'_{\mathsf{ddh}}$ has perfect completeness and $1/p$-sparse right tags.

Theorem 10. *If* $\mathsf{ddh}_{\mathcal{G}}$ *is* (t, q, ϵ)*-hard, then* $\mathsf{Auth}'_{\mathsf{ddh}}$ *is* (t', q, ϵ)-ror-cma*-secure with* $t' \approx t$.

The proof is similar to the one of Theorem 7 and is omitted.

MiM secure Protocol. We now apply our generic transformation from Section 3 to $\mathsf{Auth}'_{\mathsf{ddh}}$ to obtain a s-mim-secure protocol. By using the field structure of $\mathbb{Z}_p$ in the exponent, we can use the concrete pairwise-independent hash function $\mathsf{H}_{s_1,S_2}(R) = R^{s_1} \cdot S_2 \in \mathbb{G}$, where $(s_1, S_2) \in \mathbb{Z}_p \times \mathbb{G}$. The key of $\mathsf{Auth}_{\mathsf{ddh}}$ consists of $(y_1, y_2, K_{\mathbb{F}}, s_1, S_2)$. We now show that the key of $\mathsf{Auth}_{\mathsf{ddh}}$ can be shrinked by two elements, see Table 2. The response to a challenge c is computed as $\sigma = (R, Z)$, where $Z = (R^{y_1 \cdot c + y_2})^{K_{\mathbb{F}}} \cdot R^{s_1} S_2 = R^{y_1 K_{\mathbb{F}} \cdot c + y_2 K_{\mathbb{F}} + s_1} S_2$. This can be written as $Z = R^{x_1 c + x_2} S_2$ using the substitutions $x_1 := y_1 K_{\mathbb{F}}, \quad x_2 := y_2 K_{\mathbb{F}} + s_1, \quad X := S_3$. The resulting simplified protocol $\mathsf{Auth}_{\mathsf{ddh}}$ is given in Table 2.

Acknowledgements

David Cash was partially supported by NSF grant CNS-1453132.

Eike Kiltz was supported by a Sofja Kovalevskaja Award of the Alexander von Humboldt Foundation and ERC Project ERCC (FP7/615074).

Stefano Tessaro was partially supported by NSF grants CNS-1423566 and the Glen and Susanne Culler Chair.

This work was done in part while David Cash and Stefano Tessaro were visiting the Simons Institute for the Theory of Computing, supported by the Simons Foundation and by the DIMACS/Simons Collaboration in Cryptography through NSF grant CNS-1523467.

References

1. A. Akavia, A. Bogdanov, S. Guo, A. Kamath, and A. Rosen. Candidate weak pseudorandom functions in ac^0 ; mod_2. In *ITCS*, pages 251–260, 2014.
2. B. Applebaum, D. Cash, C. Peikert, and A. Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In S. Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 595–618, Santa Barbara, CA, USA, Aug. 16–20, 2009. Springer, Heidelberg, Germany.
3. M. Bellare and P. Rogaway. Entity authentication and key distribution. In D. R. Stinson, editor, *CRYPTO'93*, volume 773 of *LNCS*, pages 232–249, Santa Barbara, CA, USA, Aug. 22–26, 1994. Springer, Heidelberg, Germany.
4. M. Bellare and P. Rogaway. The security of triple encryption and a framework for code-based game-playing proofs. In S. Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 409–426, St. Petersburg, Russia, May 28 – June 1, 2006. Springer, Heidelberg, Germany.
5. D. J. Bernstein and T. Lange. Never trust a bunny. In *RFIDSec*, pages 137–148, 2012.
6. A. Blum, M. L. Furst, M. J. Kearns, and R. J. Lipton. Cryptographic primitives based on hard learning problems. In D. R. Stinson, editor, *CRYPTO'93*, volume 773 of *LNCS*, pages 278–291, Santa Barbara, CA, USA, Aug. 22–26, 1994. Springer, Heidelberg, Germany.

7. A. Blum, A. Kalai, and H. Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. In *32nd ACM STOC*, pages 435–440, Portland, Oregon, USA, May 21–23, 2000. ACM Press.
8. J. Bringer, H. Chabanne, and E. Dottax. HB^{++}: a Lightweight Authentication Protocol Secure against Some Attacks. In *SecPerU*, pages 28–33, 2006.
9. I. Damgård and S. Park. Towards optimally efficient secret-key authentication from PRG. Cryptology ePrint Archive, Report 2014/426, 2014. `http://eprint.iacr.org/2014/426`.
10. Y. Dodis, E. Kiltz, K. Pietrzak, and D. Wichs. Message authentication, revisited. In D. Pointcheval and T. Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 355–374, Cambridge, UK, Apr. 15–19, 2012. Springer, Heidelberg, Germany.
11. D. N. Duc and K. Kim. Securing HB^{+} Against GRS Man-in-the-Middle Attack. In *SCIS*, 2007.
12. A. Fiat and A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. In A. M. Odlyzko, editor, *CRYPTO'86*, volume 263 of *LNCS*, pages 186–194, Santa Barbara, CA, USA, Aug. 1987. Springer, Heidelberg, Germany.
13. H. Gilbert, M. J. B. Robshaw, and Y. Seurin. $HB^{\sharp}$: Increasing the security and efficiency of HB^{+}. In N. P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 361–378, Istanbul, Turkey, Apr. 13–17, 2008. Springer, Heidelberg, Germany.
14. H. Gilbert, M. J. B. Robshaw, and Y. Seurin. How to encrypt with the LPN problem. In L. Aceto, I. Damgård, L. A. Goldberg, M. M. Halldórsson, A. Ingólfsdóttir, and I. Walukiewicz, editors, *ICALP 2008, Part II*, volume 5126 of *LNCS*, pages 679–690, Reykjavik, Iceland, July 7–11, 2008. Springer, Heidelberg, Germany.
15. S. Heyse, E. Kiltz, V. Lyubashevsky, C. Paar, and K. Pietrzak. Lapin: An efficient authentication protocol based on ring-LPN. In A. Canteaut, editor, *FSE 2012*, volume 7549 of *LNCS*, pages 346–365, Washington, DC, USA, Mar. 19–21, 2012. Springer, Heidelberg, Germany.
16. N. J. Hopper and M. Blum. Secure human identification protocols. In C. Boyd, editor, *ASIACRYPT 2001*, volume 2248 of *LNCS*, pages 52–66, Gold Coast, Australia, Dec. 9–13, 2001. Springer, Heidelberg, Germany.
17. A. Jain, S. Krenn, K. Pietrzak, and A. Tentes. Commitments and efficient zero-knowledge proofs from learning parity with noise. In X. Wang and K. Sako, editors, *ASIACRYPT 2012*, volume 7658 of *LNCS*, pages 663–680, Beijing, China, Dec. 2–6, 2012. Springer, Heidelberg, Germany.
18. A. Juels and S. A. Weis. Authenticating pervasive devices with human protocols. In V. Shoup, editor, *CRYPTO 2005*, volume 3621 of *LNCS*, pages 293–308, Santa Barbara, CA, USA, Aug. 14–18, 2005. Springer, Heidelberg, Germany.
19. J. Katz, J. S. Shin, and A. Smith. Parallel and concurrent security of the HB and HB+ protocols. *Journal of Cryptology*, 23(3):402–421, July 2010.
20. E. Kiltz, K. Pietrzak, D. Cash, A. Jain, and D. Venturi. Efficient authentication from hard learning problems. In K. G. Paterson, editor, *EUROCRYPT 2011*, volume 6632 of *LNCS*, pages 7–26, Tallinn, Estonia, May 15–19, 2011. Springer, Heidelberg, Germany.
21. É. Levieil and P.-A. Fouque. An improved LPN algorithm. In R. D. Prisco and M. Yung, editors, *SCN 06*, volume 4116 of *LNCS*, pages 348–359, Maiori, Italy, Sept. 6–8, 2006. Springer, Heidelberg, Germany.

22. V. Lyubashevsky and D. Masny. Man-in-the-middle secure authentication schemes from LPN and weak PRFs. In R. Canetti and J. A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 308–325, Santa Barbara, CA, USA, Aug. 18–22, 2013. Springer, Heidelberg, Germany.
23. U. M. Maurer and J. Sjödin. A fast and key-efficient reduction of chosen-ciphertext to known-plaintext security. In M. Naor, editor, *EUROCRYPT 2007*, volume 4515 of *LNCS*, pages 498–516, Barcelona, Spain, May 20–24, 2007. Springer, Heidelberg, Germany.
24. U. M. Maurer and S. Tessaro. Basing PRFs on constant-query weak PRFs: Minimizing assumptions for efficient symmetric cryptography. In J. Pieprzyk, editor, *ASIACRYPT 2008*, volume 5350 of *LNCS*, pages 161–178, Melbourne, Australia, Dec. 7–11, 2008. Springer, Heidelberg, Germany.
25. J. Munilla and A. Peinado. HB-MP: A Further Step in the HB-Family of Lightweight Authentication Protocols. *Computer Networks*, 51(9):2262–2267, 2007.
26. K. Ouafi, R. Overbeck, and S. Vaudenay. On the security of HB# against a man-in-the-middle attack. In J. Pieprzyk, editor, *ASIACRYPT 2008*, volume 5350 of *LNCS*, pages 108–124, Melbourne, Australia, Dec. 7–11, 2008. Springer, Heidelberg, Germany.
27. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In H. N. Gabow and R. Fagin, editors, *37th ACM STOC*, pages 84–93, Baltimore, Maryland, USA, May 22–24, 2005. ACM Press.

A Omitted proofs

A.1 Proof of Theorem 7

Proof. Let A be an adversary in the $\mathsf{ROR\text{-}CMA}^{A}_{\mathsf{Auth}'}(b)$ security game. We define an adversary $B^{\mathrm{LPN}_{\mathbf{s},\alpha}()}$ against the $\mathsf{LPN}_{\ell,\gamma}$ problem, where $\alpha \in \{\gamma, \frac{1}{2}\}$ is unknown.

Adversary $B^{\mathrm{LPN}_{\mathbf{s},\alpha}}$:	**Procedure** $T(\mathbf{c})$:
$\mathbf{k}'_2 \xleftarrow{\$} \mathbb{Z}_2^\ell$	If $\mathbf{c} = \mathbf{c}^*$ then
$\mathbf{c}^* \xleftarrow{\$} \mathbb{F}_2^\ell$	$\quad \mathbf{z} \xleftarrow{\$} \mathcal{B}_\gamma^n$
$(\tau^*, state') \xleftarrow{\$} A^{T(\cdot)}(1^k, \mathbf{c}^*)$	$\quad \mathbf{R} \xleftarrow{\$} \mathbb{F}_2^{\ell \times n}$
Parse $\tau^* = (\mathbf{R}^*, \mathbf{z}^*) \in \mathbb{F}_2^{\ell \times n} \times \mathbb{F}_2^n$	Else
If $\lvert(\mathbf{R}^*)^T \cdot \mathbf{k}'_2 - \mathbf{z}^*\rvert \leq \gamma' n$ and $rank(\mathbf{R}) = n$	$\quad (\tilde{\mathbf{R}}, \mathbf{z}) \xleftarrow{\$} \mathrm{LPN}^n_{\mathbf{s},\alpha}()$
$\quad d \leftarrow \mathsf{accept}$.	$\quad \mathbf{R}^T \leftarrow \tilde{\mathbf{R}}^T \cdot (\mathbf{M}_{\mathbf{c}} - \mathbf{M}_{\mathbf{c}^*})^{-1}$
Else $d \leftarrow \mathsf{reject}$	$\tau_1 \leftarrow \mathbf{R}$
Ret $A(state, d)$	$\tau_2 \leftarrow \mathbf{z} + \mathbf{R}^T \mathbf{k}'_2$
	Ret (τ_1, τ_2)

Note that due to the finite field properties of the linear map $\mathbf{M}_{\mathbf{c}}$, matrix $\mathbf{M}_{\mathbf{c}} - \mathbf{M}_{\mathbf{c}^*}$ is always invertible for $\mathbf{c} \neq \mathbf{c}^*$. Adversary B implicitly defines $\mathbf{k}_1 := \mathbf{s}$ and $\mathbf{k}_2 := -\mathbf{M}_{\mathbf{c}^*} \cdot \mathbf{k}_1 + \mathbf{k}'_2$, where $\mathbf{s}$ is the LPN secret. As $\mathbf{k}'_2$ is uniform, the key $\mathbf{k} = (\mathbf{k}_1, \mathbf{k}_2)$ has the correct distribution. The definition of $K = (\mathbf{k}_1, \mathbf{k}_2)$ implies that

$$K(\mathbf{c}) := \mathbf{M}_{\mathbf{c}} \cdot \mathbf{k}_1 + \mathbf{k}_2 = (\mathbf{M}_{\mathbf{c}} - \mathbf{M}_{\mathbf{c}^*}) \cdot \mathbf{k}_1 + \mathbf{k}'_2. \tag{6}$$

As $K(\mathbf{c}^*) = \mathbf{k}'_2$, the bit d is always computed correctly by B. We now consider the distribution of $\mathrm{T}(\mathbf{c})$. First note that τ_1 is always a uniform matrix in $\mathbb{F}_2^{\ell \times n}$. For $\mathbf{c} = \mathbf{c}^*$, $\mathbf{z}$ is Bernoulli distributed and, using Equation (6), $\tau_2 = \mathbf{R}^T\mathbf{k}'_2 + \mathbf{z}$ is distributed as computed by prover P'. Further, for $\mathbf{c} \neq \mathbf{c}^*$ we have

$$\begin{aligned}\tau_2 &= \tilde{\mathbf{R}}^T \cdot \mathbf{s} + \mathbf{e} + \mathbf{R}^T\mathbf{k}'_2 \\ &= \mathbf{R}^T \cdot (\mathbf{M}_{\mathbf{c}} - \mathbf{M}_{\mathbf{c}^*}) \cdot \mathbf{k}_1 + \mathbf{e} + \mathbf{R}^T\mathbf{k}'_2 \\ &= \mathbf{R}^T \cdot (\mathbf{M}_{\mathbf{c}} \cdot \mathbf{k}_1 + \mathbf{k}_2) + \mathbf{e},\end{aligned}$$

where $\mathbf{e} \xleftarrow{\$} \mathcal{B}_\alpha^\ell$. If $\alpha = \frac{1}{2}$, then τ_1 and τ_2 are uniformly distributed and

$$\Pr[\, B^{\mathrm{LPN}_{\mathbf{s},1/2}()} \Rightarrow \mathsf{true}\,] = \Pr[\, \mathsf{ROR\text{-}CMA}^A_{\mathsf{Auth}'}(0) \Rightarrow \mathsf{true}\,].$$

If $\alpha = \gamma$, then $\tau = (\tau_1, \tau_2)$ is distributed as computed by prover P'. Hence $\Pr\left[\, B^{\mathrm{LPN}_{\mathbf{s},\gamma}()} \Rightarrow \mathsf{true}\,\right] = \Pr[\, \mathsf{ROR\text{-}CMA}^A_{\mathsf{Auth}'}(1) \Rightarrow \mathsf{true}\,]$. The last two equations provide $\mathbf{Adv}^{\mathsf{LPN}}(B) = \mathbf{Adv}^{\mathsf{ror\text{-}cma}}_{\mathsf{Auth}'}(A)$, where the running time of B is approximately that of A. □

A.2 Proof of Theorem 9

Proof. Let A be an attacker in the $\mathsf{ROR\text{-}CMA}(1)$ game. We now describe games $\mathsf{G}_0, \ldots, \mathsf{G}_\ell$ that are exactly like the $\mathsf{ROR\text{-}CMA}(1)$ game, but with modified procedure $T(c)$. For $j \in \{0, \ldots, \ell - 1\}$, let $S_j : \{0,1\}^j \to \mathbb{F}$ be a random function, where $S_0(\varepsilon)$ is defined to be 0. Note that S_j can be efficiently simulated by lazy evaluation.

main G_j:	**Procedure** $T(\mathbf{c})$:	$//\mathsf{G}_j$
$K \xleftarrow{\$} \mathsf{Gen}'(1^k)$	$r \xleftarrow{\$} \mathbb{D}$	
$c^* \xleftarrow{\$} \{0,1\}^\ell$	If $c_{\mid j} = c^*_{\mid j}$ then	
$(\tau^*, state) \xleftarrow{\$} A^{T(\cdot)}(c^*)$	$z = \sum_{i=1}^{\ell} F(x_{i,c_i}, r)$	
$d \leftarrow \mathsf{V}'_K(c^*, \tau^*)$	Else	
Ret $A(state, d)$	$z = S_j(c_{\mid j}) + \sum_{i=j+1}^{\ell} F(x_{i,c_i}, r)$	
	Ret $\tau = (\tau_1 \leftarrow r, \tau_2 \leftarrow z)$	

Note that in game G_0 all tags τ are computed correctly by T and hence $\mathsf{G}_0 = \mathsf{ROR\text{-}CMA}(1)$. Furthermore, in game G_ℓ, all tags except for challenge $\mathbf{c}^*$ are uniform and hence $\mathsf{G}_\ell = \mathsf{ROR\text{-}CMA}(0)$. The following lemma completes the proof of Theorem 9.

Lemma 11. *For any $j \in \{0, \ldots, \ell - 1\}$, there exists an attacker B_j such that*

$$\Pr\left[\, \mathsf{G}_j^A \Rightarrow \mathsf{true}\,\right] - \Pr\left[\, \mathsf{G}_{j+1}^A \Rightarrow \mathsf{true}\,\right] \leq \mathbf{Adv}^{\mathsf{wprf}}_{\mathcal{F}}(B).$$

To prove the lemma, we define an adversary $B = B_j^{\mathrm{O}()}$ $(0 \leq j \leq \ell - 1)$ against $\mathcal{F}$, where $\mathrm{O} \in \{\mathrm{F}_x, \mathrm{U}\}$.

Adversary B^{O}:	**Procedure** $T(\mathbf{c})$:
$c^* \xleftarrow{\$} \{0,1\}^\ell$	If $c_{\mid j+1} = c^*_{\mid j+1}$ then
$x_{i,k} = \begin{cases} \text{undefined} & i = j+1 \wedge k \neq c^*_{j+1} \\ \text{uniform in } \mathbb{K} & \text{otherwise} \end{cases}$	$r \xleftarrow{\$} \mathbb{D}; z = \sum_{i=1}^{\ell} F(x_{i,c_i}, r)$
$(\tau^*, state) \xleftarrow{\$} A^{T(\cdot)}(c^*)$	Else
Parse $\tau^* = (r^*, z^*) \in \mathbb{D} \times \mathbb{F}$	if $c_{j+1} \neq c^*_{j+1}$ then $(r, z') \xleftarrow{\$} \mathrm{O}()$
If $\sum_{i=1}^{\ell} F(x_{i,c^*_i}, r^*) = z^*$	Else $r \xleftarrow{\$} \mathbb{D}; z' = F(x_{j+1,c_{j+1}}, r)$
$d \leftarrow \mathsf{accept}$.	$z = S_j(c_{\mid j}) + z' + \sum_{i=j+2}^{\ell} F(x_{i,c_i}, r)$
Else $d \leftarrow \mathsf{reject}$	$\tau_1 \leftarrow r$
Ret $A(state, d)$	$\tau_2 \leftarrow z$
	Ret (τ_1, τ_2)

Adversary B knows all secrets $x_{i,k}$ except $x_{j+1,1-c^*_j}$ which he defines implicitly as the secret x from the F_x oracle. In particular, he knows x_{i,c^*_i} and the bit d is always computed correctly. It remains to analyze the distribution of $\mathrm{T}(\mathbf{c})$. If $c_{j+1} = c^*_{j+1}$, then the output of $\mathrm{T}(\mathbf{c})$ in games G_j and G_{j+1} is identical. We now analyze the case $c_{j+1} \neq c^*_{j+1}$. If $\mathrm{O} = \mathrm{F}_x$, then $z = S_j(c_{\mid j}) + F(x, r) + \sum_{i=j}^{\ell} F(x_{i,c_i}, r) = S_j(c_{\mid j}) + \sum_{i=j+1}^{\ell} F(x_{i,c_i}, r)$ and hence $\Pr[\, B^{\mathrm{F}_x()} \Rightarrow \mathsf{true}\,] = \Pr[\, \mathsf{G}^A_j \Rightarrow \mathsf{true}\,]$. If $\mathrm{O} = \mathrm{U}$, then $z = S_j(c_{\mid j}) + z' + \sum_{i=j+1}^{\ell} F(x_{i,c_i}, r) = S_j(c_{\mid j+1}) + \sum_{i=j+1}^{\ell} F(x_{i,c_i}, r)$ and hence $\Pr[\, B^{\mathrm{U}()} \Rightarrow \mathsf{true}\,] = \Pr[\, \mathsf{G}^A_{j+1} \Rightarrow \mathsf{true}\,]$. □