# Secure and Efficient LCMQ Entity Authentication Protocol

Zhijun Li[*] Guang Gong[†] Zhiguang Qin[‡]

**Abstract**

The simple, computationally efficient HB-like entity authentication protocols based on the learning parity with noise (LPN) problem have attracted a great deal of attention in the past few years due to the broad application prospect in low-cost RFID tags. However, all previous protocols are vulnerable to a man-in-the-middle attack discovered by Ouafi, Overbeck, and Vaudenay. In this paper, we propose a lightweight authentication protocol named LCMQ and prove it secure in a general man-in-the-middle model. The technical core in our proposal is a special type of circulant matrix, for which we prove the linear independence of matrix vectors, present efficient algorithms on matrix operations, and describe a secure encryption against ciphertext-only attack. By combining all of those with LPN and related to the multivariate quadratic problem, the LCMQ protocol not only is provably secure against all probabilistic polynomial-time adversaries, but also outperforms all HB-like protocols, in terms of tag's computation overhead, storage expense, and communication cost.

**Index Terms.** learning parity with noise, circulant matrix, multivariate quadratic, lightweight entity authentication, HB, LCMQ

## 1 Introduction

In the past few years, designing lightweight, unconventional, secure entity authentication schemes [1, 2, 3, 4] for radio frequency identification (RFID) systems has been a hot topic in the cryptography and security communities due to the imperative practical demand and the formidable theoretical challenge. Typically, RFID systems consist of simple, low-cost tags that are attached to physical objects and powerful readers that queue data from tags. As an revolutionary, efficient technique for automated identification of physical entities using radio frequency transmissions, RFID systems are employed in a wide variety of applications, such as supply chain management, payment, inventory monitoring, electronic password; and new applications are emerging every year. It is widely expected that RFID tags will inevitably replace barcodes correctly affixed to most of our daily consumer products and RFID systems will prevail in the physical identification mechanism market.

The low production expenditure of RFID tags is critical and essential to the appealing of RFID systems [1]. Roughly speaking, RFID tag's price must be below ten cents to be considered affordable

[*] Zhijun Li is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, N2L3G1 Canada (e-mail: leezj@engmail.uwaterloo.ca).

[†] Guang Gong is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, N2L3G1 Canada (e-mail: ggong@calliope.uwaterloo.ca).

[‡] Zhiguang Qin is with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan, 611731, China (e-mail: qinzg@uestc.edu.cn).

for most RFID applications [5]. On the other hand, there are a number of security and privacy challenges which have to be addressed before the prevalence of RFID systems. Secure and efficient entity authentication is a crucial one, because it is a natural approach to prevent counterfeiting—the most severe attack to the identification devices. The low-cost RFID tags, which lack the computation, communication , storage, and energy capacities necessary for most conventional cryptographic primitives, call for new lightweight authentication schemes.

The HB-like authentication protocols [1, 6, 3] have gained much attention in this field. The lightweight computation requirement of imposing only bitwise operations on authentication participants, the solid security foundation on a well-studied learning parity with noise (LPN) hard problem, and their elegant security reductionist proofs make them very attractive for entity authentication in the resource-constrained devices. Unfortunately, Ouafi, Overbeck, and Vaudenay [7] discovered an advanced man-in-the-middle attack, which is beyond the scope of the security modes used in [1, 6, 3], efficiently breaks down all HB-like protocols, and renders this kind of lightweight approaches like a dead end. Even though patching those HB-like protocols against this specific attack is possible, as we learn in the next section on the evolution of those protocols as well as enormous lessons on cryptographic algorithms and protocols, a solid, dependable authentication protocol should be provable secure against all probabilistic polynomial-time attacks, while the efficiency in those protocols cannot be sacrificed.

Aside from the LPN-based approaches, SQUASH proposed by Shamir [4] might be tempting for RFID tags authentication, because of its simpleness and provable security equivalence to Rabin's public key encryption scheme. However, its security equivalence argument has been challenged by Ouafi and Vaudenay [8]. They successfully mounted an attack against a previous version of SQUASH: SQUASH-0, which uses a linear mixing function while SQUASH employs non-linear mapping. Even thought it is not clear how or whether this attack can be adopted to SQUASH, they demonstrated that the security equivalence claim between SQUASH and Rabin cryptosystem is invalid. The security of SQUASH remains an open problem.

**Contribution**. In this paper, we present an innovative, efficient entity authentication protocol named LCMQ (standing for the combination of learning parity with noise, circulant matrix, and multivariate quadratic), which is especially suitable for RFID systems. By a general man-in-the-middle model, we prove that it is secure against all probabilistic polynomial-time adversaries. The protocol security is still based on the hardness of the LPN problem, but the architecture cannot be categorized in the HB-like schemes. Instead, the protocol greatly benefits form the gentle properties and efficient algorithms of a special type of circulant matrix, to which the whole Section 3 is devoted. Furthermore, surprisingly, the protocol performance, in terms of computation, storage, and communication costs, outweighs all previous HB-like protocols, from the standpoint of RFID tags, while it merely requires readers to additionally perform one extended Euclidean algorithm per authentication, which is trivial for supposedly powerful readers.

**Notation**. All vectors and matrices discussed in this paper are binary. Subsequently, the operations on the vectors and matrices are over the finite field $GF(2)$. The following symbols will be used throughout the rest of the paper:

| | |
|---|---|
| $\boldsymbol{a} \oplus \boldsymbol{k}$ | Bitwise exclusive-or (XOR) operation on two vectors (or matrices) $\boldsymbol{a}$ and $\boldsymbol{k}$ |
| $\boldsymbol{a} \cdot \boldsymbol{k}$ | Inner-product of two vectors $\boldsymbol{a}$ and $\boldsymbol{k}$ |
| $\mathbf{A} \circ \mathbf{K}$ | Multiplication of two matrices $\mathbf{A}$ and $\mathbf{K}$ |
| $\boldsymbol{a} \,\|\, \boldsymbol{b}$ | Concatenation of two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ |
| $\mathsf{Hwt}(\boldsymbol{k})$ | Hamming weight of vector $\boldsymbol{k}$, that is, the number of ones in the bit vector |
| $\boldsymbol{\theta} \ggg i$ | Right cyclic shift operation on vector $\boldsymbol{\theta}$ by $i$ position |
| $\mathbf{0}_m$ | $m$-bit vector in which all bits are zeros |
| $\mathbf{1}_m$ | $m$-bit vector in which all bits are ones |
| $\boldsymbol{e}_i$ | $m$-bit vector in which only bit at position $i$ is one |
| $\bar{\boldsymbol{\theta}}$ | Compliment of vector $\boldsymbol{\theta}$, i.e., $\bar{\boldsymbol{\theta}} = \boldsymbol{\theta} \oplus \mathbf{1}_m$ |
| $\mathbb{S}_m$ | Set of all $m$-bit vectors except $\mathbf{0}_m$ and $\mathbf{1}_m$ |
| $\mathbb{S}_m^{\mathrm{e}}$ | Set of all vectors in $\mathbb{S}_m$ whose Hamming weights are even |
| $\mathbb{S}_m^{\mathrm{o}}$ | Set of all vectors in $\mathbb{S}_m$ whose Hamming weights are odd |

**Organization**. The rest of the paper is structured as follows. We begin with the definition of the LPN problem and the overview of an interesting journey of HB-like protocols in Section 2. Then Section 3 is focused on the technical core of the paper: a special type of circulant matrix, for which we prove the linear independence of matrix vectors, present efficient algorithms on matrix operations, and describe a secure encryption. After that, the LCMQ protocol is proposed and proven secure in a general man-in-the-middle model in Section 4. We discuss the protocol's performance and recommend practical parameters in Section 5. Finally, Section 6 concludes the work.

# 2 Previous LPN-Based Authentication Protocols

## 2.1 LPN Problem

Suppose the tag pre-shares a secret $m$-bit vector $\boldsymbol{k}$ with the reader for subsequent authentications. First the reader randomly generates a sequence of binary vectors $\boldsymbol{b}_0, \boldsymbol{b}_1, \cdots, \boldsymbol{b}_{q\text{-}1}$ and transmits those challenges to the tag, which responds with $y_i = \boldsymbol{b}_i \cdot \boldsymbol{k}$, for $i \in [0, q-1]$ accordingly. The reader accepts the tag's authentication if and only if $\boldsymbol{b}_i \cdot \boldsymbol{k} = y_i$. Unfortunately, after observing $m$ linearly-independent challenge-response pairs of $\langle \boldsymbol{b}_i, y_i \rangle$, an adversary can readily recover the authentication key $\boldsymbol{k}$ by the Gaussian elimination.

In the presence of noise, however, where each response bit $y_i$ is independently flipped by a noise bit one with probability $\eta \in (0, \frac{1}{2})$, determining $\boldsymbol{k}$ becomes much more difficult. This problem is known as Learning Parity with Noise, or the *LPN Problem*. Formally, it is defined as follows.

**Definition 1** (LPN Problem)**.** *View $m$ as a security parameter. Let $\boldsymbol{k}$ be an $m$-bit secret vector, $\eta \in (0, \frac{1}{2})$ be a noise level. For $i \in [0, q-1]$ ($q$ is a polynomial in $m$), let $\boldsymbol{b}_i$ be an $m$-bit random vector, and $v_i$ be a noise bit that follows the Bernoulli probability distribution of parameter $\eta$. Given $\eta$ and $q$ pairs $\langle \boldsymbol{b}_i, y_i = (\boldsymbol{b}_i \cdot \boldsymbol{k}) \oplus v_i \rangle$, recover $\boldsymbol{k}$.*

The LPN problem has long been studied as the following equivalent problems: syndrome decoding problem [9, 10] and minimal disagreement parity problem [11]. It has been proven that the LPN problem is NP-hard [10]. Moreover, finding a vector satisfying more than half of the challenge-response pairs, even though it looks like an easier problem, remains NP-hard [12]. Furthermore, Regev [13] introduced a natural extension of the LPN problem, referring to as the Learning With Error (LWE) problem, by generalizing binary field $GF(2)$ in the LPN problem into prime field $GF(p)$, where $p$ is a prime number. Impressively, Regev [13] proved the reduction from worse-case lattice problems, such as Shortest-Vector Problem (SVP), to the LWE problem. However, the reduction proof employs a quantum algorithm, which is, generally speaking, weaker than a classical reduction mechanism, as there is still no practical quantum computer available by now.

In reality, the security of LPN-based authentication protocols, similar to other NP-hard problems for application in the cryptography, still depends on the hardness of the average case of the LPN problem, while the NP-hard allegation only guarantees the intractability in the worse case. Intuitively, the combination of the key length $m$ and the noise level $\eta$ determines the security level of LPN instances. Blum, Kalai, and Wasserman [14] provided the first sub-exponential algorithm (BKW algorithm) for the LPN problem, which requires $2^{O(m/\log m)}$ equations/operations. Fossorier et al. [15] improved the BKW algorithm. At present, the fastest algorithm is the LF algorithm, another enhancement of BKM algorithm, presented by Levieil and Fouque [16]. According to the LF algorithm, a common parameter set for 80-bit security level is $(\eta = 0.25, m = 512)$. Should LPN-based protocols be widely employed, it is highly likely that algorithms of the LPN problem can be improved notably, then bigger key lengths are demanded, as we have witnessed the significant increase of RSA (and discrete logarithm) public key length in the three decades. Since typical LPN-based protocols involve $(m \times m \times O(m))$ matrix multiplication, such big values of $m$ would incur considerable computation and implementation costs so as to push protocols away from *lightweight*. Fortunately, in our proposed protocol, LPN instances are encrypted by a succinct secure scheme so that the protocol does not suffer from the restriction and a practical value of key length $m$ can be as low as 163.

## 2.2 The Journey of HB-Family Authentication Protocols

**HB and HB$^+$ Protocols**. Hopper and Blum [17] first presented a natural LPN-based authentication protocol (*HB protocol*), aimed at providing accessible identification for unassisted humans. The HB protocol is provably secure against passive eavesdroppers [17, 1, 6] under the assumption of the LPN problem's intractability. For an active attacker, the noise in LPN instances can be overcome simply by querying with identical challenges many times, which is referred to as the *JW attack* [1], and thus the authentication key would be easily retrieved. Focusing on the lightweight authentication for RFID systems, Juels and Weis [1] proposed the HB$^+$ authentication protocol, which prevents the JW attack by adding blinding vectors. One authentication procedure in HB$^+$ consists of $n$ rounds of three-pass interactions between the tag and the reader. One single round of HB$^+$ is outlined in Fig. 1. After $n$ rounds, the reader accepts the tag's authentication if and only if the number of unmatched challenge-response pairs does not exceed a threshold $\tau$.

Juels and Weis [1] presented an elegant reductionist security proof of the HB$^+$ protocol in a limited active model: detection-based-model, which is primarily addressing active attacks similar to the JW attack. Originally, the security proof of HB$^+$ in [1] demands the sequential execution of $n$ rounds three-pass interactions. To overcome this limitation, Katz and Shin [6] brought an security proof of the HB$^+$ protocol in the case of parallel and concurrent executions. Moreover, Katz and Smith [18] extended the reduction results to a larger range of noise levels $\frac{1}{4} \leq \eta < \frac{1}{2}$ whereas the Katz-Shin proof [6] holds only on the condition of $\eta < \frac{1}{4}$. In spite of those gentle security proofs, Gilbert, Robshaw, and Sibert [19]

$$\begin{array}{ccc}
\text{Tag } (\boldsymbol{k}_1, \boldsymbol{k}_2) & & \text{Reader } (\boldsymbol{k}_1, \boldsymbol{k}_2) \\
\hline\hline
\boldsymbol{b}_i \in_R \{0,1\}^m & \xrightarrow{\quad \boldsymbol{b}_i \quad} & \\
 & \xleftarrow{\quad \boldsymbol{a}_i \quad} & \boldsymbol{a}_i \in_R \{0,1\}^m \\
v_i \in_R \{0, 1 | \Pr[v_i = 1] = \eta\}; & & \\
y_i = (\boldsymbol{a}_i \cdot \boldsymbol{k}_1) \oplus (\boldsymbol{b}_i \cdot \boldsymbol{k}_2) \oplus v_i & \xrightarrow{\quad y_i \quad} & \\
 & & (\boldsymbol{a}_i \cdot \boldsymbol{k}_1) \oplus (\boldsymbol{b}_i \cdot \boldsymbol{k}_2) \overset{?}{=} y_i
\end{array}$$

Figure 1: The $i$th round of the HB$^+$ authentication protocol, where $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$ are two $m$-bit vectors as authentication key, $\eta \in (0, \frac{1}{2})$, $\boldsymbol{b}_i$ is a *blinding vector*, $\boldsymbol{a}_i$ is a *challenge vector*

discovered a simple, effective man-in-the-middle (MIM) attack (referred to as the *GRS attack*), which is outside of the detection-based-model, and fully compromises the HB$^+$ protocol.

**GRS Attack**. In second pass of every round of one HB$^+$ authentication procedure, an MIM adversary intercepts challenge $\boldsymbol{a}_i$ from the reader, and transmits to the tag a modified challenge $\boldsymbol{a}_i \oplus \boldsymbol{\alpha}$, where $\boldsymbol{\alpha}$ is a constant vector for one authentication procedure. By observing this manipulated authentication procedure outcome—acceptance or rejection, the adversary learns the result of $\boldsymbol{\alpha} \cdot \boldsymbol{k}_1$, that is, one bit information of $\boldsymbol{k}_a$. The adversary simply repeats $m$ times of manipulating authentication procedures with linearly independent $\boldsymbol{\alpha}$'s, and completely recovers $\boldsymbol{k}_1$. Now the adversary is able to impersonate a valid tag by choosing $\mathbf{0}_m$ as the blinding vector; or the adversary can further determine $\boldsymbol{k}_2$ by acting as a tag to interact with a genuine reader, using a constant blinding vector $\boldsymbol{b}'$ in one authentication procedure, responding challenge $\boldsymbol{a}_i$ with $\boldsymbol{a}_i \oplus \boldsymbol{k}_1$, and learning the result of $\boldsymbol{b}' \oplus \boldsymbol{k}_2$ according to acceptance or rejection. In addition, although the original GRS attack is restricted to the interference of challenges from the reader to the tag, the same GRS manipulation strategy can be applied to blinding vectors to recover $\boldsymbol{k}_2$; after that, the adversary can launch the original JW attack to retrieve $\boldsymbol{k}_1$, totally breaking the protocol.

Even after a series of HB$^+$ enhancement protocols, such as HB$^{++}$ [20], HB$^*$ [21], HB-MP [22], modification of HB$^{++}$ [23] and HB-MP$^+$ [24] had been proposed, Gilbert, Robshaw, and Seurin [25] demonstrated that those variants still could be attacked in the linear time while increasing the computational complexity and/or reducing the practicality. The PUF-HB protocol [26] and the Trusted-HB protocol [2] make use of a physically unclonable circuit and a lightweight hash function family respectively, intending to thwart the GRS attack. However, the introduction of such ingredients into HB$^+$ might not fully meet the motivation of designing lightweight simple-bitwise-operation-based authentication protocols. Moreover, Frumkin and Shamir [27] have broken the security of Trusted-HB in realistic scenarios.

**Random-HB$^\#$ and HB$^\#$ Protocols**. Gilbert, Robshaw, and Seurin [3] presented these two protocols, which are resistant to the GRS attack. In contrast to secret vectors in HB$^+$, Random-HB$^\#$ employs two secret matrices $\mathbf{K}_1$ and $\mathbf{K}_2$. One Random-HB$^\#$ authentication consists of a blinding vector $\boldsymbol{b}$ from the tag, a challenge vector $\boldsymbol{a}$ from the reader, and then the tag's response vector $\boldsymbol{y} = (\boldsymbol{a} \circ \mathbf{K}_1) \oplus (\boldsymbol{b} \circ \mathbf{K}_2) \oplus \boldsymbol{v}$, where $\boldsymbol{v}$ is an $n$-bit noise vector each bit of which independently follows the Bernoulli distribution of parameter $\eta$. Similarly, the reader validates the tag's authentication iff $\mathsf{Hwt}((\boldsymbol{a} \circ \mathbf{K}_1) \oplus (\boldsymbol{b} \circ \mathbf{K}_2) \oplus \boldsymbol{y})$ does not exceed threshold $\tau$. The binding/challenge vectors rather than matrices in HB$^+$ exceedingly reduces the communication cost, but the secret matrices in Random-HB$^\#$ imposes too high storage burden to be practical in realistic systems. In order to overcome the drawback, they proposed to replace random matrices with Toeplitz matrices, which becomes the HB$^\#$ protocol.

Gilbert, Robshaw, and Seurin [3] defined a GRS-MIM-model, in which the MIM adversary is only allowed to manipulate the challenges from the reader to the tag, to prove that Random-HB$^\#$ and HB$^\#$ [3] are resist to the GRS attack. In addition, Random-HB$^\#$ is provably secure in the detection-based-model, while HB$^\#$ is conjectured to be secure [3]. The security reductionist proofs in [3] are rather impressive. However, as the GRS-MIM-model does not simulate a full man-in-the-middle adversary, a general MIM attack was discovered soon, breaking down both Random-HB$^\#$ and HB$^\#$, and making the perspectives of secure LPN-based authentication protocols gloomy.

**OOV Attack**. At AsiaCrypt 2008, Ouafi, Overbeck and Vaudenay [7] presented a general man-in-the-middle attack (referred to as *OOV attack*) against all current HB-like protocols. The basic OOV attack against Random-HB$^\#$/HB$^\#$ is conducted as follows. The attacker first eavesdrops on one successful execution of the protocol, obtaining a triplet $(\widehat{\boldsymbol{b}}, \widehat{\boldsymbol{a}}, \widehat{\boldsymbol{y}})$ satisfying $\widehat{\boldsymbol{y}} = (\widehat{\boldsymbol{a}} \circ \mathbf{K}_1) \oplus (\widehat{\boldsymbol{b}} \circ \mathbf{K}_2) \oplus \widehat{\boldsymbol{v}}$ and $\mathsf{Hwt}(\widehat{\boldsymbol{v}}) \leq \tau$. Then the MIM adversary manipulates many executions of the protocol by XORing interactions $(\boldsymbol{b}_i, \boldsymbol{a}_i, \boldsymbol{y}_i)$ with $(\widehat{\boldsymbol{b}}, \widehat{\boldsymbol{a}}, \widehat{\boldsymbol{y}})$; thus each authentication result is actually decided by whether $\mathsf{Hwt}(\boldsymbol{v}_i \oplus \widehat{\boldsymbol{v}}) \leq \tau$. Based on the overall success probability, the attacker can calculate the value of $\mathsf{Hwt}(\widehat{\boldsymbol{v}})$ with a high probability. After that, the adversary changes $\widehat{\boldsymbol{y}}$ by one bit to $\widehat{\boldsymbol{y}}'$, uses $(\widehat{\boldsymbol{b}}, \widehat{\boldsymbol{a}}, \widehat{\boldsymbol{y}}')$ to interfere with many executions of the protocol, and get the result of $\mathsf{Hwt}(\widehat{\boldsymbol{v}}')$ , where $\widehat{\boldsymbol{v}}' = (\widehat{\boldsymbol{a}} \circ \mathbf{K}_1) \oplus (\widehat{\boldsymbol{b}} \circ \mathbf{K}_2) \oplus \widehat{\boldsymbol{y}}'$. By comparing the values of $\mathsf{Hwt}(\widehat{\boldsymbol{v}})$ and $\mathsf{Hwt}(\widehat{\boldsymbol{v}}')$, one bit in noise vector $\widehat{\boldsymbol{v}}$ is determined. Repeating this process, the adversary eventually obtains the noise-free result of $(\widehat{\boldsymbol{a}} \circ \mathbf{K}_1) \oplus (\widehat{\boldsymbol{b}} \circ \mathbf{K}_2)$. The adversary collects enough equations that he can completely recover $\mathbf{K}_1$ and $\mathbf{K}_2$, breaking the protocol.

Ouafi, Overbeck and Vaudenay [7] also examined the lower bounds on the parameter sets for which the OOV attack is not effective. As concluded in [7], such parameters are unpractical to use in the low-cost devices. One may argue that since the OOV attack would cause many rejections, it can be relieved by setting up an upper bound of rejection number such that an authentication key shall be revoked once the number of failed authentication using the key exceeds the bound. This cumbersome approach counts on the outside mechanism, and is not satisfactory.

**Noise Modes and Error Rates**. For the HB-like protocols with the Bernoulli noise mode, there exist two types of authentication errors. A *false negative*, that is, the authentication of a legitimate tag being rejected, takes place when the number of incorrect responses exceeds the pass-threshold $\tau$. By contrast, a *false positive* is defined that the number of unmatched responses out of random bits is less than the pass-threshold $\tau$. In other words, we assume that an illegitimate tag only responses with random bits. The false negative rate $P_{\mathrm{FN}}$ and the false positive rate $P_{\mathrm{FP}}$ are determined [16, 3] by

$$P_{\mathrm{FN}} = \sum_{i=\tau+1}^{n} \binom{n}{i} \eta^i (1-\eta)^{n-i} \text{ and } P_{\mathrm{FP}} = \sum_{i=0}^{\tau} \binom{n}{i} 2^{-n} \quad . \tag{1}$$

Since the Bernoulli noise mode would cause a certain false negative rate in HB-like protocols, a natural method to overcome that drawback is to demand the tag to generate a noise vector $\boldsymbol{v}_i$ of bounded Hamming weight, that is $\mathsf{Hwt}(\boldsymbol{v}_i) \leq \tau$, as discussed in [6, 3]. We refer to it as the upper-bounded Binomial noise mode. Ouafi, Overbeck and Vaudenay [7] demonstrated another simple man-in-the-middle attack (referred to as *OOV2 attack*) against HB-like protocols with this noise mode. For one iteration $(\boldsymbol{b}_i, \boldsymbol{a}_i, \boldsymbol{y}_i)$ of Random-HB$^{\#}$ with this noise mode, an OOV2 attacker manipulates the response $\boldsymbol{y}_i$ such that the reader receives $\boldsymbol{y}_i \oplus \boldsymbol{r}_i$ rather than $\boldsymbol{y}_i$, where $\boldsymbol{r}_i$ is a random vector of Hamming weight 2. Let $w_i = \mathsf{Hwt}((\boldsymbol{a}_i \circ \mathbf{K}_1) \oplus (\boldsymbol{b}_i \circ \mathbf{K}_2) \oplus \boldsymbol{y}_i)$ be the Hamming weight of the noise added by the tag. If and only if $w = \tau - 1$ or $\tau$ and the attacker flipped two non-erroneous bits, which come from the only two non-zero elements in $\boldsymbol{r}_i$, in the response, the reader rejects the authentication. In other words, from one occurrence of rejection, the attackers learn two bits of $\mathbf{K}_1/\mathbf{K}_2$. Subsequently, all bits of secret matrices can be retrieved by conducting the process many times.

# 3 Linear Independence, Efficient Computation, and Encryption Scheme on A Special Type of Circulant Matrix

Traditionally, a circulant matrix is a square matrix in which each row vector is rotated one element to the right relative to the preceding row vector. That is, an $(m \times m)$ square circulant matrix with first row vector $\boldsymbol{\theta} = (\theta_0, \theta_1, \cdots, \theta_{m\text{-}1})$ is

$$\begin{bmatrix} \theta_0 & \theta_1 & \cdots & \theta_{m-1} \\ \theta_{m-1} & \theta_0 & \cdots & \theta_{m-2} \\ \vdots & \vdots & \ddots & \vdots \\ \theta_1 & \theta_2 & \cdots & \theta_0 \end{bmatrix} .$$

Let $n$ be an integer in $[1, m-1]$, we extend a circulant matrix into none-square cases: defining a landscape circulant matrix as an $(n \times m)$ matrix in which each row vector is a right cyclic shift by one of the row vector above, and a portrait circulant matrix as an $(m \times n)$ matrix in which each column vector is a right cyclic shift by one of the column vector before it, while referring to the original one as a square circulant matrix.

The technical core in our proposal is a special type of circulant matrix—*circulant-P2 matrix*, which is define as follow.

**Definition 2** (Circulant-P2 matrix)**.** *A circulant-P2 matrix is an $(m \times m)$ square circulant matrix, or an $(n \times m)$ landscape circulant matrix, or an $(m \times n)$ portrait circulant matrix, satisfying the following conditions.*

1. *It is a binary matrix.*
2. *$m$ is a prime number satisfying that 2 is a primitive element of finite field $GF(m)$.*
3. *Neither $\mathbf{0}_m$ nor $\mathbf{1}_m$ is a row vector (or a column vector) of a circulant-P2 matrix.*

Note that the definition above implies $n < m$. The second condition is central for circulant-P2 matrices. If and only if $2^i \mod m \neq 1$, $\forall 1 \leq i \leq m-2$, then 2 is a primitive element of finite field

$GF(m)$. We list all integers less than 2048 satisfying that condition: 3, 5, 11, 13, 19, 29, 37, 53, 59, 61, 67, 83, 101, 107, 131, 139, 149, 163, 173, 179, 181, 197, 211, 227, 269, 293, 317, 347, 349, 373, 379, 389, 419, 421, 443, 461, 467, 491, 509, 523, 541, 547, 557, 563, 587, 613, 619, 653, 659, 661, 677, 701, 709, 757, 773, 787, 797, 821, 827, 829, 853, 859, 877, 883, 907, 941, 947, 1019, 1061, 1091, 1109, 1117, 1123, 1171, 1187, 1213, 1229, 1237, 1259, 1277, 1283, 1291, 1301, 1307, 1373, 1381, 1427, 1451, 1453, 1483, 1493, 1499, 1523, 1531, 1549, 1571, 1619, 1621, 1637, 1667, 1669, 1693, 1733, 1741, 1747, 1787, 1861, 1867, 1877, 1901, 1907, 1931, 1949, 1973, 1979, 1987, 1997, 2027, 2029.

A *characteristic vector* of a square circulant-P2 matrix is defined as its first row vector. As for a landscape or portrait circulant-P2 matrix, since it is actually a truncated portion of a square circulant-P2 matrix, its characteristic vector is defined as the corresponding square circulant-P2 matrix's characteristic vector. For a circulant-P2 matrix with $m$-bit characteristic vector $\boldsymbol{\theta}$, we denote square, landscape, and portrait cases by $\mathbf{C}_{\boldsymbol{\theta}}$, $\mathbf{C}_{\boldsymbol{\theta}}^{[n\times m]}$, and $\mathbf{C}_{\boldsymbol{\theta}}^{[m\times n]}$ respectively.

## 3.1 Linear Independence

**Definition 3** (Equivalence Class)**.** *For two vectors in $\mathbb{S}_m$ (recall that $\mathbb{S}_m$ is the set of all $m$-bit vectors except $\mathbf{0}_m$ and $\mathbf{1}_m$), say $\boldsymbol{a}$ and $\boldsymbol{b}$, if $\exists i \in \{0, \cdots, m-1\}$ such that $\boldsymbol{b} = \boldsymbol{a} \ggg i$, then we define that $\boldsymbol{a}$ and $\boldsymbol{b}$ are cyclically shift equivalent and they are in an equivalence class.*

An equivalence class can be represented by any one of its members.

**Lemma 1.** *If $m$ is a prime number, then there are $\frac{2^m-2}{m}$ disjoint equivalence classes in $\mathbb{S}_m$. Each equivalence class contains $m$ elements.*

*Proof.* An equivalence class in $\mathbb{S}_m$ has at most $m$ elements; and any two different equivalence classes are disjoint—they do not share any common elements. Since $\mathbf{0}_m$ and $\mathbf{1}_m$ are not elements in $\mathbb{S}_m$, every equivalence class contains at least two elements. Suppose there is an equivalence class $\nleq$ that has less than $m$ elements. It means that there exists at least one element $\boldsymbol{\theta'}$ satisfying $\boldsymbol{\theta'} \ggg i = \boldsymbol{\theta'}$ where $1 < i < m$ ($i$ cannot be 1; otherwise the equivalence class only has one element). Due to the characteristic of equivalence class, the relation $\boldsymbol{\theta} \ggg i = \boldsymbol{\theta}$ holds for every element $\boldsymbol{\theta}$ in $\nleq$. Consequently, $i$ should be a factor of $m$. However, it contradicts the fact that $m$ is prime, since $m$ only has two factors 1 and $m$ while $1 < i < m$. Therefore, every equivalence class of $\mathbb{S}_m$ has exact $m$ elements, and there are $\frac{2^m-2}{m}$ disjoint equivalence classes in $\mathbb{S}_m$. □

A proof of Lemma 1 also can be found in [28]. For completeness, we present this proof.

**Lemma 2.** *If $m$ is prime and 2 is a primitive element of finite field $GF(m)$, then the polynomial $x^{m-1} + x^{m-2} + \cdots + x + 1$ is irreducible over $GF(2)$.*

This lemma is proven in [29].

**Lemma 3.** *If $m$ is a prime number satisfying that 2 is a primitive element of $GF(m)$, then any $m-1$ elements in every equivalence class of $\mathbb{S}_m$ are linearly independent. In other words, all row vectors in a landscape circulant-P2 matrix (and all column vectors in a portrait circulant-P2 matrix) are linearly independent.*

*Proof.* Let $\boldsymbol{\theta} = (\theta_0, \theta_1, \cdots, \theta_{m\text{-}1}) \in \mathbb{S}_m$, we may view the square circulant matrix $\mathbf{C}_{\boldsymbol{\theta}}$ as a linear feedback shift register sequence $\tilde{\boldsymbol{\theta}} = (\theta_0, \theta_1, \cdots, \theta_{m\text{-}1}, \theta_0, \theta_1, \cdots, \theta_{m\text{-}1}, \cdots)$ of characteristic polynomial $x^m + 1$ over finite field $GF(2)$, according to [28]. Note that $x^m + 1 = (x+1)(x^{m-1} + x^{m-2} + \cdots + x + 1)$ over $GF(2)$. Let $g(x) = x^{m-1} + x^{m-2} + \cdots + x + 1$. Since $m$ is prime, according to Lemma 1, sequence $\tilde{\boldsymbol{\theta}}$ has period $m$. Thus, we only need to consider the following two cases.

*Case 1:* $\theta_0 \oplus \theta_1 \oplus \cdots \oplus \theta_{m-1} = 0$

In this case, sequence $\tilde{\boldsymbol{\theta}}$ is generated by $g(x)$. Based on Lemma 2, $g(x)$ is irreducible over GF(2) if 2 is a primitive element of finite field $GF(m)$. Since the degree of $g(x)$ is equal to $m-1$, then any $m-1$ vectors in $\mathbf{C}_{\boldsymbol{\theta}}$ are linearly independent.

*Case 2:* $\theta_0 \oplus \theta_1 \oplus \cdots \oplus \theta_{m-1} = 1$

In this case, sequence $\tilde{\boldsymbol{\theta}}$ is not generated by $g(x)$ but by polynomial $x^m + 1$. Since $x^m + 1$ has degree $m$, then all $m$ vectors in $\mathbf{C}_{\boldsymbol{\theta}}$ are linearly independent.

In summary, if $m$ is a prime number and 2 is a primitive element of $GF(m)$, then any $m-1$ elements in every equivalence class of $\mathbb{S}_m$ are linearly independent. □

The above proof explicitly concludes the following lemma.

**Lemma 4.** *All $m$ row vectors in a square circulant-P2 matrix $\mathbf{C}_{\boldsymbol{\theta}}$ are linearly independent if and only if the Hamming weight of $\boldsymbol{\theta}$ is odd. Consequently, $\mathbf{C}_{\boldsymbol{\theta}}$ is invertible if only if the Hamming weight of $\boldsymbol{\theta}$ is odd.*

The inverse of a square circulant matrix, if it exists, is still a square circulant matrix.

**Lemma 5.** *A landscape circulant-P2 matrix always has a right inverse. That is, for an $(n \times m)$ landscape circulant-P2 matrix $\mathbf{C}$, there exists an $(m \times n)$ matrix $\mathbf{M}$ such that $\mathbf{C} \circ \mathbf{M} = \mathbf{I}_n$, where $\mathbf{I}_n$ is the $(n \times n)$ identity matrix. Likewise, an portrait circulant-P2 matrix always has a left inverse.*

*Proof.* According to Lemma 3, any $(n \times m)$ landscape circulant-P2 matrix has full rank: its rank is equal to $n$. Therefore, it has a right inverse. The argument for left inverse of a portrait circulant-P2 matrix is same. □

## 3.2 Matrix Operations and Properties

A better way to analyze operations on circulant-P2 matrices is to convert them to polynomials, as used in [30]. Every vector can be represented in a polynomial form, as described in the following definition.

**Definition 4** (Associate Polynomial and Associate Vector)**.** *For a vector $\boldsymbol{\theta} = (\theta_0, \theta_1, \cdots, \theta_{m\text{-}1})$, its* associate polynomial *$\theta(x)$ in $GF(2)[x]$ is defined as*

$$\theta(x) = \sum_{i=0}^{m-1} \theta_i x^i \ .$$

*Correspondingly, $\boldsymbol{\theta}$ is the associate vector of polynomial $\theta(x)$.*

Henceforth, we will freely use those two forms to represent a vector. If we define a vector by one form, then we can use the other representation without explicit explanations.

**Algorithm 1** Inverse of circulant-P2 matrix multiplication

```
Input: $m$-bit vector $\boldsymbol{\kappa} \in \mathbb{S}_m$, $m$-bit vector $\boldsymbol{z} = \boldsymbol{y} \circ \mathbf{C}_{\boldsymbol{\kappa}}^{[n\times m]}$
Output: $n$-bit vector $\boldsymbol{y}$
calculate $\kappa^{-1}(x)$ by extended Euclidean algorithm
$t(x) \Leftarrow z(x) * \kappa^{-1}(x) \mod f_m(x)$
if $\mathsf{Hwt}(\boldsymbol{\kappa})$ is odd then
    $\boldsymbol{y} \Leftarrow$ the leftmost $n$-bit sub-vector of $\boldsymbol{t}$
else
    $y_0 \Leftarrow t_0$
    $i \Leftarrow 1$
    while $i < n$ do
        $y_i \Leftarrow y_{i-1} \oplus t_i$
        $i \Leftarrow i + 1$
```

We define $f_m(x) = x^m + 1$, a polynomial in $GF(2)[x]$. Let $\boldsymbol{\phi}, \boldsymbol{\kappa}, \boldsymbol{z} \in \mathbb{S}_m$. We now work with polynomials modulo $f_m(x)$, so that the cyclic shift can be effected by polynomial multiplication module $f(x)$. That is, the vector $(\boldsymbol{\phi} \ggg i)$, or equivalently $\boldsymbol{\phi} \circ \mathbf{C}_{\boldsymbol{e}_i}$, where $0 \le i \le m-1$, is associated with the polynomial

$$\phi(x) * x^i \mod f_m(x) \ ;$$

reducing modulo $f_m(x)$ achieves the effect of the cyclic shift. Computing $\boldsymbol{\phi} \circ \mathbf{C}_{\boldsymbol{\kappa}}$ combines the several cyclic shifts on $\boldsymbol{\phi}$, each of which is decided by a different bit one in $\boldsymbol{\kappa}$. Subsequently, the computation of $\boldsymbol{z} = \boldsymbol{\phi} \circ \mathbf{C}_{\boldsymbol{\kappa}}$, or $\mathbf{C}_{\boldsymbol{z}} = \mathbf{C}_{\boldsymbol{\phi}} \circ \mathbf{C}_{\boldsymbol{\kappa}}$, can be performed by

$$z(x) = \phi(x) * \kappa(x) \mod f_m(x) \ .$$

It is clear from the above equation that $\boldsymbol{z} = \boldsymbol{\phi} \circ \mathbf{C}_{\boldsymbol{\kappa}} = \boldsymbol{\kappa} \circ \mathbf{C}_{\boldsymbol{\phi}}$.

An efficient method of calculating the right inverse for an landscape circulant-P2 matrix is described in Algorithm 1. Main technique in Algorithm 1 is adopted from [30], and we develop the solution for the case of $\mathsf{Hwt}(\boldsymbol{\kappa})$ being even. This algorithm applies to all kinds of circulant-P2 matrices' inverses if they exists.

*Correctness Proof of Algorithm 1.* Let $\boldsymbol{\phi} = \boldsymbol{y}||\mathbf{0}_{m-n}$, thus $\boldsymbol{z} = \boldsymbol{\phi} \circ \mathbf{C}_{\boldsymbol{\kappa}}$.

We can use the extended Euclidean algorithm on input polynomials $\kappa(x)$ and $f_m(x)$ to find polynomials $\kappa^{-1}(x)$—the general inverse of $\kappa(x)$— and $w(x)$ such that

$$\kappa^{-1}(x) * \kappa(x) + w(x) * f_m(x) = g(x) \ ,$$

where $g(x) = \gcd(\kappa(x), f_m(x))$.

If $\mathsf{Hwt}(\boldsymbol{\kappa})$ is odd, then $\mathbf{C}_{\boldsymbol{\kappa}}$ is invertible, according to Lemma 4. In other words, $g(x) = 1$. Therefore, $\phi(x) = t(x) = z(x) * \kappa^{-1}(x) \mod f_m(x)$.

If $\mathsf{Hwt}(\boldsymbol{\kappa})$ is even, $\mathbf{C}_{\boldsymbol{\kappa}}$ is not invertible, namely $g(x) \ne 1$. According to Lemma 2, the polynomial $x^{m-1} + x^{m-2} + \cdots + x + 1$ is irreducible; thus the factorization of $f_m(x)$ is equal to $(x+1)(x^{m-1} + x^{m-2} + \cdots + x + 1)$. Since $\boldsymbol{\kappa}$ is neither $\mathbf{0}_m$ nor $\mathbf{1}_m$, $\kappa(x)$ is not equal to $x^{m-1} + x^{m-2} + \cdots + x + 1$.

Therefore, $g(x) = x + 1$. Consequently, polynomial $t(x) = z(x) * s(x) \mod f_m(x)$ is associated with vector

$$\boldsymbol{t} = \boldsymbol{\phi} \circ \mathbf{C}_{\boldsymbol{g}} \ .$$

Since $\boldsymbol{\phi} = \boldsymbol{y}||\mathbf{0}_{m-n}$, then

$$y_i = \begin{cases} t_0 & \text{if } i = 0 \\ y_{i-1} \oplus t_i & \text{if } 1 \le i \le n-1 \end{cases} \ . \tag{2}$$

Let $\boldsymbol{t}'$ be the leftmost $n$-bit sub-vector of $\boldsymbol{t}$. For future reference, we denote by $\boldsymbol{y} = \mathsf{Tran}(\boldsymbol{t}')$ the transformation in Eqt. (2). Correspondingly, $\boldsymbol{t}' = \mathsf{Tran}^{-1}(\boldsymbol{y})$.

In either case, the algorithm correctly outputs $\boldsymbol{y}$. □

**Examples**. Two examples with small parameters are provided to demonstrate the algorithm. Let $m = 5$, $n = 4$, $\boldsymbol{y} = 1011$. Accordingly, we have $\boldsymbol{\phi} = 10110$ and $\phi(x) = 1 + x^2 + x^3, f_m(x) = x^5 + 1$. The following two examples are corresponding to the two cases respectively.

(i) $\boldsymbol{\kappa} = 10011$, a case that $\mathsf{Hwt}(\boldsymbol{\kappa})$ is odd.

Then $\kappa(x) = 1 + x^3 + x^4$; and $z(x) = \phi(x) * \kappa(x) \mod f_m(x) = x^4$ .

Given $\phi(x)$ and $z(x)$, determine $\kappa^{-1}(x) = x + x^3 + x^4$ by extended Euclidean algorithm, and then

$$\phi(x) = t(x) = \kappa^{-1}(x) * z(x) \mod f_m(x) = 1 + x^2 + x^3 \ .$$

Thus $\boldsymbol{\phi} = 10110$ and $\boldsymbol{y} = 1011$.

(ii) $\boldsymbol{\kappa} = 10010$, a case that $\mathsf{Hwt}(\boldsymbol{\kappa})$ is even.

Then $\kappa(x) = 1 + x^3$, and $z(x) = \phi(x) * \kappa(x) \mod f_m(x) = x + x^2$ .

Given $\phi(x)$ and $z(x)$, determine $\kappa^{-1}(x) = 1 + x^3$, and then

$$t(x) = \kappa^{-1}(x) * z(x) \mod f_m(x) = 1 + x + x^2 + x^4 \ .$$

That is, $\boldsymbol{t} = 11101$. By Eqt. (2), finally recover $\boldsymbol{y} = 1011$.

**Remark 1.** *In Algorithm 1, let $\boldsymbol{t}'$ be the leftmost $n$-bit sub-vector of $\boldsymbol{t}$. It is clear from Eqt. (2) that if $\boldsymbol{y}$ is uniformly distributed over $\{0,1\}^n$, then $\boldsymbol{t}'$ is uniformly distributed over $\{0,1\}^n$; and vice versa. Moreover, for an $n$-bit vector $\boldsymbol{\gamma}$, if $\boldsymbol{\gamma} \oplus \boldsymbol{y}$ is uniformly distributed over $\{0,1\}^n$, then $\boldsymbol{\gamma} \oplus \boldsymbol{t}'$ is uniformly distributed over $\{0,1\}^n$; and vice versa.*

**Fact 1.** *Let $m$ be a prime number satisfying 2 is a primitive element of $GF(m)$. For all vectors in $\mathbb{S}_m$, with respect to matrix multiplication of corresponding square circulant-P2 matrices (or equivalently, modular polynomial multiplication of their associate polynomials),*

1. *All vectors in $\mathbb{S}_m^{\mathrm{o}}$ constitute an Abelian multiplication group of size $2^{m-1}-1$, with identity element $\boldsymbol{e}_0$;*
2. *All vectors in $\mathbb{S}_m^{\mathrm{e}}$ constitute an Abelian multiplication group of size $2^{m-1}-1$, with identity element $\bar{\boldsymbol{e}}_0$;*
3. *The complement of a vector in $\mathbb{S}_m^{\mathrm{o}}$ is an element in $\mathbb{S}_m^{\mathrm{e}}$, and vice versa;*

4. *If vector $\boldsymbol{\theta} \in \mathbb{S}_m^{\mathrm{o}}$, then $\boldsymbol{\theta} \circ \mathbf{C}_{\bar{\boldsymbol{e}}_0} = \bar{\boldsymbol{\theta}}$; if $\boldsymbol{\theta} \in \mathbb{S}_m^{\mathrm{e}}$, then $\boldsymbol{\theta} \circ \mathbf{C}_{\bar{\boldsymbol{e}}_0} = \boldsymbol{\theta}$;*

5. *For two vectors $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ in $\mathbb{S}_m$, $\boldsymbol{\theta} \circ \mathbf{C}_{\boldsymbol{\phi}} \in \mathbb{S}_m^{\mathrm{o}}$ if and only if $\boldsymbol{\theta}, \boldsymbol{\phi} \in \mathbb{S}_m^{\mathrm{o}}$.*

*Proof.* Fact 1.3 is obvious. From the correctness proof of Algorithm 1, we can easily get Fact 1.1. As for Fact 1.4, let $\boldsymbol{\theta}' = \boldsymbol{\theta} \circ \mathbf{C}_{\bar{\boldsymbol{e}}_0}$, then $\theta_i'$ is equal to $\theta_i \oplus$ parity of $\mathsf{Hwt}(\boldsymbol{\theta})$ (0 for even, 1 for odd) for $i = 0, 1, \cdots, m-1$. Therefore, if $\boldsymbol{\theta} \in \mathbb{S}_m^{\mathrm{o}}$, $\boldsymbol{\theta}' = \boldsymbol{\theta} \oplus \mathbf{1}_m = \bar{\boldsymbol{\theta}}$; if $\boldsymbol{\theta} \in \mathbb{S}_m^{\mathrm{e}}$, $\boldsymbol{\theta}' = \boldsymbol{\theta}$, concluding this fact. Facts 1.2 and 1.5 can be directly derived from Facts 1.1, 1.3, and 1.4. □

### 3.3 A Secure Encryption Against Ciphertext-Only Attack

Now we are ready to introduce a symmetric-key encryption scheme based on circulant-P2 matrix:

$$\boldsymbol{z} = \boldsymbol{\theta} \circ \mathbf{C}_{\boldsymbol{\kappa}}^{[(m-1)\times m]} \quad , \tag{3}$$

where plaintext $\boldsymbol{\theta}_i$ is an $(m-1)$-bit random vector and $\boldsymbol{\theta} \neq \mathbf{0}_{m-1}$, encryption key $\boldsymbol{\kappa}$ is randomly selected from $\mathbb{S}_m^{\mathrm{e}}$, and ciphertext $\boldsymbol{z}$ is, subsequently, an element in $\mathbb{S}_m^{\mathrm{e}}$. Accordingly, the sizes of plaintext space, key space, and ciphertext space are all the same: $2^{m-1} - 1$. The encryption operation can alternatively represented by $\boldsymbol{z} = \mathsf{Enc}(\boldsymbol{\theta}_i, \boldsymbol{\kappa})$; and the corresponding decryption, denoted by $\boldsymbol{\theta} = \mathsf{Dec}(\boldsymbol{z}_i, \boldsymbol{\kappa})$ is performed via Algorithm 1.

It is easy to see from the properties of circulant-P2 matrix that by choosing a random vector $\kappa' \in \mathbb{S}_m^{\mathrm{e}}$, a valid $\boldsymbol{\theta}'$ satisfying $\boldsymbol{z} = \boldsymbol{\theta}' \circ \mathbf{C}_{\boldsymbol{\kappa}'}^{[(m-1)\times m]}$ can always be retrieved via Algorithm 1; for any different $\boldsymbol{\theta}'$, a different $\boldsymbol{\kappa}'$ can be found to map them to any $\boldsymbol{z}$, and vice versa. This fact guarantees the scheme's security against ciphertext-only attack. In other words, given a ciphertext, an adversary cannot learn any useful information about the encryption key and the plaintext, because each ciphertext is corresponding to $2^{m-1} - 1$ distinct combinations of plaintext-key pairs. Thus every plaintext/key is equally possible for any ciphertext. Hence the encryption is semantically secure against ciphertext-only attack as long as plaintexts are random.

Alternatively, we may use the encryption: $\boldsymbol{z} = \boldsymbol{\theta} \circ \mathbf{C}_{\boldsymbol{\kappa}}$, where plaintext $\boldsymbol{\theta}$, key $\boldsymbol{\kappa}$, and ciphertext $\boldsymbol{z}$ all belong to $\mathbb{S}_m^{\mathrm{o}}$. The arguments above apply to it, and the encryption scheme with random plaintexts is semantically secure ciphertext-only attack.

In practice, a stand-alone encryption scheme only secure against ciphertext-only attack is rarely useful. However, coupled with the hardness of the LPN problem, this scheme can lead to a succinct, highly efficient, and secure entity authentication scheme, which we will describe in next section.

# 4 LCMQ Protocol

## 4.1 Protocol Description

The LCMQ protocol with the Bernoulli noise mode is illuminated in Fig. 2. In this scheme, two $m$-bit vectors $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$ are shared by a tag and a reader as a pair of symmetric authentication keys with one condition that the parities of both keys' Hamming weights are known to the public. Because of the inherent requirement of circulant-P2 matrix that the interaction expansion $n$ should be less than the key length $m$, and the impact of $n$ on false rates according to Eqt. (1), $n = m - 1$ is recommended in most cases. Similar to conventional identification schemes, one LCMQ authentication procedure consists of

$$\begin{array}{ccc}
\text{Tag } (\boldsymbol{k}_1,\boldsymbol{k}_2) & & \text{Reader } (\boldsymbol{k}_1,\boldsymbol{k}_2)\\
\hline\hline
 & \xleftarrow{\quad \boldsymbol{a}\quad} & \boldsymbol{a}\in_R \mathbb{S}^{\mathrm{e}}_m\\
\boldsymbol{b}\in_R\mathbb{S}_m; & & \\
\boldsymbol{v}\in_R \{\{0,1\}^n|\Pr[v_j=1] & & \\
=\eta, \text{ where } 0\le j\le n-1\}; & & \\
\boldsymbol{y}=(\boldsymbol{b}\circ\mathbf{C}^{[m\times n]}_{\boldsymbol{k}_1})\oplus\boldsymbol{v}; & & \\
\boldsymbol{r}\in_R\{0,1\}^{m-n-1}; & & \\
\boldsymbol{z}=(\boldsymbol{y}||\boldsymbol{r})\circ\mathbf{C}^{[(m\text{-}1)\times m]}_{\boldsymbol{k}_2\oplus\boldsymbol{a}} & \xrightarrow{\quad\boldsymbol{b},\boldsymbol{z}\quad} & \\
 & & \boldsymbol{y}||\boldsymbol{r}=\mathsf{Dec}(\boldsymbol{z},\boldsymbol{k}_2\oplus\boldsymbol{a});\\
 & & \text{ACCEPT iff } \mathsf{Hwt}((\boldsymbol{b}\circ\mathbf{C}^{[m\times n]}_{\boldsymbol{k}_1})\oplus\boldsymbol{y})\le\tau
\end{array}$$

Figure 2: LCMQ authentication protocol, where $m$ is a prime number satisfying that 2 is a primitive element of $GF(m)$, $\boldsymbol{k}_1 \xleftarrow{\$} \mathbb{S}_m$ and the parity of $\mathsf{Hwt}(\boldsymbol{k}_1)$ is public, $\boldsymbol{k}_2 \xleftarrow{\$} \mathbb{S}^{\mathrm{e}}_m$, $n$ is the interaction expansion and $n < m$, noise level $\eta \in (0, \frac{1}{2})$, integer pass-threshold $\tau \in (\eta n, \frac{n}{2})$

two passes: a challenge $\boldsymbol{a}$ by the reader and a response pair $(\boldsymbol{b}, \boldsymbol{z})$ by the tag, rendering $3m$ bits transmission payload. The computation in the tag's side mainly involves two vector/matrix multiplications of roughly $m^3$ XOR operations, while the reader additionally, to calculate an inverse of a circulant-P2 matrix, needs to perform the extended Euclidean algorithm, which is surely a trivial requirement to the supposedly powerful reader. As we will argue later, $m = 163$ would suffice to provide 80-bit security, and the LCMQ protocol achieves outstanding performances in terms of all metrics: storage expense, computational payload, communication cost, and implementation expenditure. Most important, we will prove LCMQ secure against general man-in-the-middle attacks. All of those promising properties of LCMQ make it very suitable for the authentication of RFID systems.

In order to prevent malicious behaviors, tag should check if $\boldsymbol{a}$ belongs to $\mathbb{S}^{\mathrm{e}}_m$, and reader should check if $\boldsymbol{b} \in \mathbb{S}_m$ and if $\boldsymbol{z} \in \mathbb{S}^{\mathrm{e}}_m$, upon receiving them; if any of those abnormalities takes place, the participant would terminate this round of authentication. In addition, if $\boldsymbol{y}||\boldsymbol{r} = \mathbf{0}_{m-1}$, technically, tag should repeat its procedure of generating new $\boldsymbol{y}||\boldsymbol{r}$. Since the probability of such an event is negligible, equal to $1/(2^{m-1} - 1)$, tag need not bother to take this countermeasure. Note that it is impossible that $\boldsymbol{k}_2 \oplus \boldsymbol{a} = \mathbf{1}_m$ since both $\boldsymbol{k}_2$ and $\boldsymbol{a}$ are in $\mathbb{S}^{\mathrm{e}}_m$. If $\boldsymbol{k}_2 \oplus \boldsymbol{a} = \mathbf{0}_m$, the LCMQ protocol fails, but such a case only takes place with the negligible probability $1/(2^{m-1} - 1)$. Therefore, we can safely presume that it would never happen and will not consider it in the rest of the paper for simplicity.

The proposed LCMQ protocol, thought it is still LPN-based, has a different architecture from the previous HB-like protocols [1, 3]. By the encryption $\boldsymbol{z} = \mathsf{Enc}(\boldsymbol{y}||\boldsymbol{r}, \boldsymbol{k}_2 \oplus \boldsymbol{a})$, protocol LCMQ conceals the LPN answer $\boldsymbol{y}$ from adversaries such that it can use a smaller key length, which is a vital factor to determine protocols computation and communication performances. More important, the encryption/decryption operations provide an implicit integrity mechanism for $(\boldsymbol{a}, \boldsymbol{z})$. Benefited from linear independence of circulant-P2 matrix vectors, any alteration on $(\boldsymbol{a}, \boldsymbol{z})$ will render the authentication to fail with a overwhelming probability, as the case of manipulating $\boldsymbol{b}$. In addition, there is no correlation effect of simultaneously manipulating $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{z}$ (The error bits introduced by changing one can be canceled off, to a notable extent, by the error bits from altering others.), thus the LCMQ protocol overcomes the flaw in the HB-like protocols [1, 3] that renders them subject to the OOV attack.

## 4.2 Security Models Definitions

To formally define security models, we denote an LCMQ authentication system by a pair of probabilistic functions $(\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}, \mathcal{R}_{\boldsymbol{k}_1,\boldsymbol{k}_2,n,\tau})$, namely a tag function $\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}$ and a reader function $\mathcal{R}_{\boldsymbol{k}_1,\boldsymbol{k}_2,n,\tau}$.

The fundamental objective of an adversary in the entity authentication protocol is to impersonate the tag. By replying a random vector as an authentication response, the probability that an adversary impersonating the tag will success is the false positive rate $P_{\mathrm{FP}}$. This is the best soundness error we can achieve for the LCMQ protocol. Therefore, we define the advantage of an adversary $\mathcal{A}$ against LCMQ in a model as its overall success probability over $P_{\mathrm{FP}}$ in impersonating the tag.

**Definition 5** (DET-Model)**.** *In the DET-model, which is identical to the detection-based-model used in [1, 6, 18, 3], the DET attack is carried out in two phases:*

- Phase 1*: Adversary $\mathcal{A}$ interacts q times with the tag $\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}$. On the ith invocation, $\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}$ takes a challenge vector $\boldsymbol{a}_i$ from $\mathcal{A}$ as input, selects a random vector $\boldsymbol{b}_i \in \mathbb{S}_m$, generates a noise vector $\boldsymbol{v}_i$ according to the Bernoulli noise mode, and calculates $\boldsymbol{y}_i = (\boldsymbol{b}_i \circ \mathbf{C}_{\boldsymbol{k}_1}^{[m\times n]}) \oplus \boldsymbol{v}_i$. Furthermore, $\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}$ internally generates a random $(m\text{-}n\text{-}1)$-bit vector $\boldsymbol{r}_i$, and computes $\boldsymbol{z}_i = (\boldsymbol{y}_i||\boldsymbol{r}_i) \circ \mathbf{C}_{\boldsymbol{k}_2\oplus\boldsymbol{a}_i}^{[(m\text{-}1)\times m]}$. Then $\boldsymbol{b}_i$ and $\boldsymbol{z}_i$ are transmitted to $\mathcal{A}$.*

- Phase 2*: Adversary $\mathcal{A}$ receives a random challenge $\widehat{\boldsymbol{a}}$ from $\mathcal{R}_{\boldsymbol{k}_1,\boldsymbol{k}_2,n,\tau}$, and then outputs $(\widehat{\boldsymbol{b}}, \widehat{\boldsymbol{z}})$ corresponding to $\widehat{\boldsymbol{a}}$, intended to pass the verification of $\mathcal{R}_{\boldsymbol{k}_1,\boldsymbol{k}_2,n,\tau}$ with advantage*

$$\begin{aligned}\mathsf{Adv}_{\mathcal{A}}^{\mathrm{DET}}(m,\eta,n,\tau) \overset{def}{=} \ & \mathsf{Pr}[\boldsymbol{k}_1 \xleftarrow{\$} \mathbb{S}_m, \boldsymbol{k}_2 \xleftarrow{\$} \mathbb{S}_m^{\mathrm{e}}, \mathcal{A}^{\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}}(1^m) : \\ & \langle \mathcal{A}, \mathcal{R}_{\boldsymbol{k}_1,\boldsymbol{k}_2,n,\tau}, \mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n} \rangle = \mathrm{ACCEPT}] - P_{\mathrm{FP}} \ .\end{aligned}$$

**Definition 6** (MIM-model)**.** *In the MIM-model, the MIM attack is carried out in two phases:*

- Phase 1*: Adversary $\mathcal{A}$ manipulates any communications between the tag $\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}$ and the reader $\mathcal{R}_{\boldsymbol{k}_1,\boldsymbol{k}_2,n,\tau}$ for q executions. Fig. 3 depicts the ith manipulation, which simulates a full MIM attacker. We define three interference vectors: $\boldsymbol{\alpha}_i = \boldsymbol{a}_i \oplus \boldsymbol{a}'_i, \boldsymbol{\beta}_i = \boldsymbol{b}_i \oplus \boldsymbol{b}'_i$ and $\boldsymbol{\zeta}_i = \boldsymbol{z}_i \oplus \boldsymbol{z}'_i$.*

- Phase 2*: Adversary $\mathcal{A}$ receives a random challenge $\widehat{\boldsymbol{a}}$ from $\mathcal{R}_{\boldsymbol{k}_1,\boldsymbol{k}_2,n,\tau}$, and then outputs $(\widehat{\boldsymbol{b}}, \widehat{\boldsymbol{z}})$ corresponding to $\widehat{\boldsymbol{a}}$, intended to pass the verification of $\mathcal{R}_{\boldsymbol{k}_1,\boldsymbol{k}_2,n,\tau}$ with advantage*

$$\begin{aligned}\mathsf{Adv}_{\mathcal{A}}^{\mathrm{MIM}}(m,\eta,n,\tau) \overset{def}{=} \ & \mathsf{Pr}[\boldsymbol{k}_1 \xleftarrow{\$} \mathbb{S}_m, \boldsymbol{k}_2 \xleftarrow{\$} \mathbb{S}_m^{\mathrm{e}}, \mathcal{A}^{\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n},\mathcal{R}_{\boldsymbol{k}_1,\boldsymbol{k}_2,n,\tau}}(1^m) : \\ & \langle \mathcal{A}, \mathcal{R}_{\boldsymbol{k}_1,\boldsymbol{k}_2,n,\tau}, \mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n} \rangle = \mathrm{ACCEPT}] - P_{\mathrm{FP}} \ .\end{aligned}$$

The MIM-model is a very strong security from the adversary's perspective and it is easy to see that the DET-model is a limited version of the MIM-model. An authentication protocol provably secure in MIM-model will naturally resist all probabilistic polynomial time (PPT) attacks. In the following, we first provide a concrete security reductionist proof from the DET-model to the MIM-model for the LCMQ protocol; then we prove its security in the DET-model based on the hardness of the LPN problem, with some reasonable assumptions.

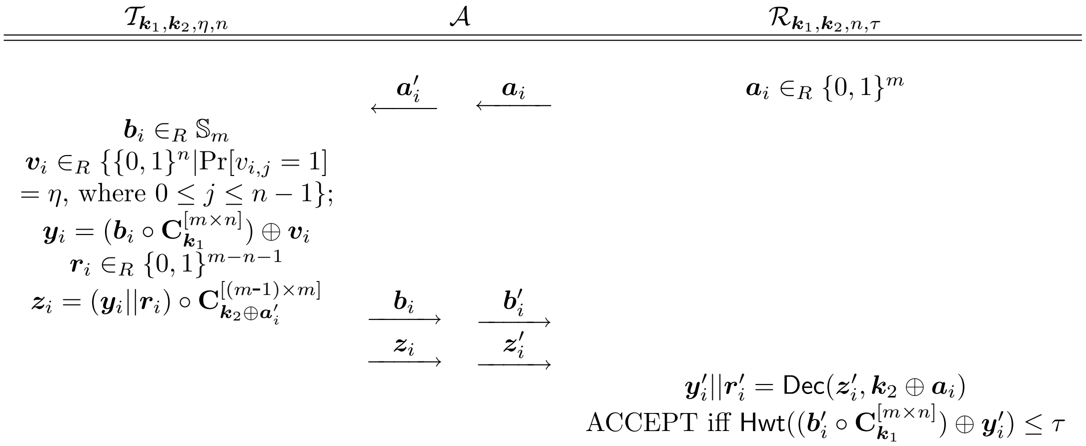
$\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}$
$\mathcal{A}$
$\mathcal{R}_{\boldsymbol{k}_1,\boldsymbol{k}_2,n,\tau}$
$\boldsymbol{a}_i \in_R \{0,1\}^m$
$\xleftarrow{\boldsymbol{a}'_i}$
$\xleftarrow{\boldsymbol{a}_i}$
$\boldsymbol{b}_i \in_R \mathbb{S}_m$
$\boldsymbol{v}_i \in_R \{\{0,1\}^n|\Pr[v_{i,j}=1]$
$=\eta$, where $0 \le j \le n-1\}$;
$\boldsymbol{y}_i = (\boldsymbol{b}_i \circ \mathbf{C}^{[m\times n]}_{\boldsymbol{k}_1}) \oplus \boldsymbol{v}_i$
$\boldsymbol{r}_i \in_R \{0,1\}^{m-n-1}$
$\boldsymbol{z}_i = (\boldsymbol{y}_i||\boldsymbol{r}_i) \circ \mathbf{C}^{[(m-1)\times m]}_{\boldsymbol{k}_2\oplus\boldsymbol{a}'_i}$
$\xrightarrow{\boldsymbol{b}_i}$
$\xrightarrow{\boldsymbol{b}'_i}$
$\xrightarrow{\boldsymbol{z}_i}$
$\xrightarrow{\boldsymbol{z}'_i}$
$\boldsymbol{y}'_i||\boldsymbol{r}'_i = \mathsf{Dec}(\boldsymbol{z}'_i, \boldsymbol{k}_2 \oplus \boldsymbol{a}_i)$
ACCEPT iff $\mathsf{Hwt}((\boldsymbol{b}'_i \circ \mathbf{C}^{[m\times n]}_{\boldsymbol{k}_1}) \oplus \boldsymbol{y}'_i) \le \tau$

Figure 3: $i$th manipulation in the MIM-model

## 4.3 Reduction from DET-model to MIM-model

**Theorem 1.** *If there is an adversary $\mathcal{A}$ attacking the LCMQ protocol in the MIM-model, modifying $q$ executions of the protocol between an honest tag and an honest reader, running in time $t$, and achieving $\mathsf{Adv}^{\mathrm{MIM}}_{\mathcal{A}}(m,\eta,n,\tau) \ge \delta$, then there exists an adversary $\mathcal{A}'$ attacking the LCMQ protocol in the DET-model, interacting at most $q$ oracle queries, running in time $O(t)$, and achieving $\mathsf{Adv}^{\mathrm{DET}}_{\mathcal{A}'}(m,\eta,n,\tau) \ge \delta - q\epsilon(P_{\mathrm{FP}}+\delta)$ for some negligible function $\epsilon$, under the assumption that $P_{\mathrm{FP}}$ and $P_{\mathrm{FN}}$ are negligible. Hence, if protocol LCMQ is secure in the DET-model, then it is provably secure in the MIM-model.*

*Proof.* In Phase 1, $\mathcal{A}'$ can readily simulate the honest tag for $\mathcal{A}$ since $\mathcal{A}'$ has access to $\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}$. The main challenge lies on how to simulate $\mathcal{R}_{\boldsymbol{k}_1,\boldsymbol{k}_2,n,\tau}$ for $\mathcal{A}$. Similar to the proof method for the Random-HB$^{\#}$ protocol [3], $\mathcal{A}'$ launches Phase 1 of adversary $\mathcal{A}$, and simulates the tag and the reader $q$ times as follows:

1. $\mathcal{A}'$ sends a random vector $\boldsymbol{a}_i$ as the challenge of the simulated reader, and let $\mathcal{A}$ modify it to $\boldsymbol{a}'_i$; then $\mathcal{A}'$ forwards $\boldsymbol{a}'_i$ to $\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}$.

2. $\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}$ faithfully responds with $(\boldsymbol{b}_i, \boldsymbol{z}_i = (((\boldsymbol{b}_i \circ \mathbf{C}^{[n\times m]}_{\boldsymbol{k}_1}) \oplus \boldsymbol{v}_i)||\boldsymbol{r}_i) \circ \mathbf{C}^{[(m-1)\times m]}_{\boldsymbol{k}_2\oplus\boldsymbol{a}'_i})$ to $\mathcal{A}'$, which relays $(\boldsymbol{b}_i, \boldsymbol{z}_i)$ to $\mathcal{A}$. Then $\mathcal{A}$ alters them to $(\boldsymbol{b}'_i, \boldsymbol{z}'_i)$, and uses $(\boldsymbol{b}'_i, \boldsymbol{z}'_i)$ as the authentication response to $\mathcal{A}'$.

3. During the interactions, if $\boldsymbol{a}'_i \notin \mathbb{S}^{\mathrm{e}}_m$ or $\boldsymbol{b}'_i \notin \mathbb{S}_m$ or $\boldsymbol{z}'_i \notin \mathbb{S}^{\mathrm{e}}_m$, $\mathcal{A}'$ terminates the iteration and proceeds with the next, abiding by the protocol specification.

4. If $\boldsymbol{a}'_i = \boldsymbol{a}_i$ and $\boldsymbol{b}'_i = \boldsymbol{b}_i$ and $\boldsymbol{z}'_i = \boldsymbol{z}_i$, $\mathcal{A}'$ outputs "ACCEPT" to $\mathcal{A}$ as the authentication result of the simulated reader; if $\boldsymbol{a}'_i = \boldsymbol{a}_i$ and $\boldsymbol{z}'_i = \boldsymbol{z}_i$ and $\boldsymbol{\beta} = \mathbf{1}_m$ and $\mathsf{Hwt}(\boldsymbol{k}_1)$ is odd, $\mathcal{A}'$ outputs "ACCEPT" too; elsewise, it outputs "REJECT".

After Phase 1, $\mathcal{A}'$ launches Phase 2 of $\mathcal{A}$. Since Phase 2 in the DET-model is identical to that in the MIM-model, $\mathcal{A}'$ just replicates $\mathcal{A}$'s behavior with the real reader, with the same objective of passing the authentication. Therefore, if $\mathcal{A}$ achieves $\mathsf{Adv}^{\text{MIM}}_{\mathcal{A}}(m,\eta,n,\tau) \geq \delta$, then the probability of $\mathcal{A}'$ successfully impersonating a valid tag is equal to the success probability of $\mathcal{A}$, i.e., $P_{\text{FP}}+\delta$, on the condition that the reader is correctly simulated by $\mathcal{A}'$ in Phase 1.

We denote by $P_{\text{err}}$ the probability of $\mathcal{A}'$ wrongly simulating the reader for $\mathcal{A}$ in one iteration of Phase 1. Executions in Phase 1 can be divided into four different cases:

*Case 1*: $\boldsymbol{a}'_i=\boldsymbol{a}_i$ and $\boldsymbol{z}'_i=\boldsymbol{z}_i$ and $\boldsymbol{b}'_i=\boldsymbol{b}_i$.

In this case, $\mathcal{A}'$ outputs "ACCEPT", and fails at simulating the reader with a probability equal to the false negative rate $P_{\text{FN}}$.

*Case 2*: $\boldsymbol{a}'_i=\boldsymbol{a}_i$ and $\boldsymbol{z}'_i=\boldsymbol{z}_i$ but $\boldsymbol{b}_i\neq\boldsymbol{b}_i$.

Thus $\boldsymbol{y}'_i=\boldsymbol{y}_i$. Since

$$(\boldsymbol{b}'_i\circ\mathbf{C}^{[m\times n]}_{\boldsymbol{k}_1})\oplus\boldsymbol{y}_i=((\boldsymbol{b}'_i\oplus\boldsymbol{b}_i)\circ\mathbf{C}^{[m\times n]}_{\boldsymbol{k}_1})\oplus\boldsymbol{v}_i=(\boldsymbol{\beta}_i\circ\mathbf{C}^{[m\times n]}_{\boldsymbol{k}_1})\oplus\boldsymbol{v}_i=(\boldsymbol{k}_1\circ\mathbf{C}^{[m\times n]}_{\boldsymbol{\beta}_i})\oplus\boldsymbol{v}_i\ ,$$

the authentication result in this case is equivalently decided by

$$\mathsf{Hwt}((\boldsymbol{k}_1\circ\mathbf{C}^{[m\times n]}_{\boldsymbol{\beta}_i})\oplus\boldsymbol{v}_i)\leq\tau\ .$$

Let

$$\boldsymbol{d}'_i=\boldsymbol{k}_1\circ\mathbf{C}^{[m\times n]}_{\boldsymbol{\beta}_i} \tag{4}$$

be the error vector added by $\mathcal{A}$ by changing $\boldsymbol{b}_i$.

If $\boldsymbol{\beta}_i\in\mathbb{S}_k$, according to Lemma 3, all column vectors in the portrait circular-P2 matrix $\mathbf{C}^{[m\times n]}_{\boldsymbol{\beta}_i}$ are linearly independent. Following the same argument in Theorem 2 of [6], $\boldsymbol{d}'_i$ is uniformly distributed over $\{0,1\}^n$, as the column vectors of $\mathbf{C}^{[m\times n]}_{\boldsymbol{\beta}_i}$ are linearly independent. Thus the resulting error vector $\boldsymbol{d}'_i\oplus\boldsymbol{v}_i$ follows the uniform distribution over $\{0,1\}^n$ since $\mathcal{A}$ has no extra knowledge about the random noise vector $\boldsymbol{v}_i$. As a result, the probability of $\mathcal{A}'$ wrongly outputting "REJECT" is exactly the same as the false positive rate $P_{\text{FP}}$.

If $\boldsymbol{\beta}_i=\mathbf{1}_m$ and $\mathsf{Hwt}(\boldsymbol{k}_1)$ is even, then $\boldsymbol{d}'_i=\mathbf{0}_n$, and the probability of $\mathcal{A}'$ wrongly outputting "ACCEPT" is exactly $P_{\text{FN}}$. When $\boldsymbol{\beta}_i=\mathbf{1}_m$ and $\mathsf{Hwt}(\boldsymbol{k}_1)$ is odd, then $\boldsymbol{d}'_i=\mathbf{1}_n$. Consequently, the probability of $\mathcal{A}'$ wrongly outputting "REJECT" is $\sum_{i=\tau+1}^{n}\binom{n}{i}(1-\eta)^i\eta^{n-i}$, which is always less than $P_{\text{FN}}$ since $\eta n<\tau<n/2$.

Overall in this case, $P_{\text{err}}\leq\max(P_{\text{FP}},P_{\text{FN}})$.

*Case 3*: $\boldsymbol{b}'_i=\boldsymbol{b}_i$ and at least one of $\boldsymbol{\alpha}_i$ and $\boldsymbol{\zeta}_i$ is not equal to $\mathbf{0}_m$.

The error vector introduced by the adversary through changing $\boldsymbol{z}_i$ and/or $\boldsymbol{a}_i$ is denoted by

$$\boldsymbol{d}''_i=\boldsymbol{y}_i\oplus\boldsymbol{y}'_i\ . \tag{5}$$

Correspondingly, the authentication result is decided by

$$\mathsf{Hwt}(\boldsymbol{d}''_i\oplus\boldsymbol{v}_i)\leq\tau\ .$$

Recall that $\boldsymbol{a}_i,\boldsymbol{a}'_i,\boldsymbol{z}_i,\boldsymbol{z}'_i\in\mathbb{S}^{\text{e}}_m$; thus $\boldsymbol{\alpha}_i\neq\mathbf{1}_m$ and $\boldsymbol{\zeta}_i\neq\mathbf{1}_m$. Let $\boldsymbol{\kappa}_i=\boldsymbol{k}_2\oplus\boldsymbol{a}_i$, $\boldsymbol{\kappa}'_i=\boldsymbol{k}_2\oplus\boldsymbol{a}'_i$, $s_i(x)$ and $s'_i(x)$ be the general inverses of $\kappa_i(x)$ and $\kappa'_i(x)$ respectively. In addition, let $\boldsymbol{\lambda}_i=\boldsymbol{s}_i\oplus\boldsymbol{s}'_i$. It is clear that $\boldsymbol{\lambda}_i\in\mathbb{S}^{\text{e}}_m$ if $\boldsymbol{\alpha}_i\in\mathbb{S}^{\text{e}}_m$ and $\boldsymbol{\lambda}_i=\mathbf{0}_m$ if $\boldsymbol{\alpha}_i=\mathbf{0}_m$. Moreover, $\mathcal{A}$ does not know any addition information about $s_i(x)$ and $s'_i(x)$; otherwise, $\mathcal{A}$ must have recovered some of $\boldsymbol{k}_2$.

From equations

$$\begin{aligned} s_i(x)(k_2(x)+a_i(x)) &\equiv 1+x \mod f_m(x) \ , \\ (s_i(x)+\lambda_i(x))(k_2(x)+a_i(x)+\alpha_i(x)) &\equiv 1+x \mod f_m(x) \ , \end{aligned}$$

we have $\lambda_i(x)(k_2(x)+a_i(x))+s_i(x)\alpha_i(x) \equiv 0 \mod f_m(x)$, and then

$$\lambda_i(x)(1+x)+s_i^2(x)\alpha_i(x) \equiv 0 \mod f_m(x) \ . \tag{6}$$

If $\mathcal{A}$ knows $\lambda_i(x)$, he can recover $s_i^2(x)$ by the equation above, and then $s_i(x)$ and $k_2(x)$ are leaked. Therefore, even though $\mathcal{A}$ can freely choose $\boldsymbol{\alpha}_i$, if $\boldsymbol{k}_2$ is unknown to him, $\mathcal{A}$ should not have useful information about $\boldsymbol{\lambda}_i$.

Let $t_i(x) = z_i(x) * s'_i(x) \mod f_m(x)$ and $t'_i(x) = z'_i(x) * s_i(x) \mod f_m(x)$; let $\boldsymbol{\gamma}_i, \boldsymbol{\gamma}'_i$ be the leftmost $n$-bit sub-vectors of $\boldsymbol{t}_i, \boldsymbol{t}'_i$ respectively. Then we have

$$\begin{aligned} t'_i(x)+t_i(x) &= z'_i(x)*s_i(x)+z_i(x)*(s'_i(x)-s_i(x)+s_i(x)) \mod f_m(x) \\ &= (z'_i(x)+z_i(x))*s_i(x)+z_i(x)*(s'_i(x)+s_i(x)) \mod f_m(x) \\ &= \zeta_i(x)*s_i(x)+z_i(x)*\lambda_i(x) \mod f_m(x) \ . \end{aligned}$$

Subsequently,

$$\boldsymbol{d}''_i = \mathsf{Tran}(\boldsymbol{\gamma}'_i \oplus \boldsymbol{\gamma}_i) = \mathsf{Tran}((\boldsymbol{s}_i \circ \mathbf{C}^{[m\times n]}_{\boldsymbol{\zeta}_i}) \oplus (\boldsymbol{z}_i \circ \mathbf{C}^{[m\times n]}_{\boldsymbol{\lambda}_i})) \ . \tag{7}$$

If $\boldsymbol{\zeta}_i \in \mathbb{S}_m$, then $\boldsymbol{s}_i \circ \mathbf{C}^{[m\times n]}_{\boldsymbol{\zeta}_i}$ is uniformly distributed over $\{0,1\}^n$; if $\boldsymbol{\alpha}_i \in \mathbb{S}_m$, which implies $\boldsymbol{\lambda}_i \in \mathbb{S}_m$, then $\boldsymbol{z}_i \circ \mathbf{C}^{[m\times n]}_{\boldsymbol{\lambda}_i}$ is uniformly distributed over $\{0,1\}^n$. Note that $\boldsymbol{s}_i$ and $\boldsymbol{\lambda}_i$ (if $\boldsymbol{\alpha}_i \neq \mathbf{0}_m$) are unknown to $\mathcal{A}$, and $\mathcal{A}$ receives $\boldsymbol{z}_i$ only after he has revealed his decision of $\boldsymbol{\alpha}_i$. Therefore, if only one of $\boldsymbol{\alpha}_i$ and $\boldsymbol{\zeta}_i$ is in $\mathbb{S}_m$, then $\boldsymbol{d}''_i$ is uniformly distributed oven $\{0,1\}^n$, by Remark 1.

If both $\boldsymbol{\alpha}_i$ and $\boldsymbol{\zeta}_i$ are not equal to $\mathbf{0}_m$, since

$$(t'_i(x)+t_i(x))(1+x) \equiv s_i(x)(\zeta_i(x)(1+x)+s_i(x)\alpha_i(x)) \mod f_m(x) \ ,$$

and $\mathcal{A}$ cannot choose a valid pair of $(\boldsymbol{\alpha}_i, \boldsymbol{\zeta}_i)$ satisfying $\zeta_i(x)(1+x)+s_i(x)\alpha_i(x) \equiv 0$ or $1+x+x^2+\ldots+x^{m-1}$ mod $f_m(x)$ without knowledge of $s_i(x)$, then $\boldsymbol{d}''_i$ is still uniformly distributed oven $\{0,1\}^n$.

As a result, in this case, $\mathcal{A}'$ erroneously outputs "REJECT" with probability $P_{\mathrm{FP}}$.

*Case 4*: $\boldsymbol{b}_i \neq \boldsymbol{b}'_i$ and at least one of $\boldsymbol{\alpha}_i$ and $\boldsymbol{\zeta}_i$ is not equal to $\mathbf{0}_m$.

This case is the combination of Case 2 and Case 3, and the authentication result is determined by

$$\mathsf{Hwt}(\boldsymbol{d}'_i \oplus \boldsymbol{d}''_i \oplus \boldsymbol{v}_i) \leq \tau \ ,$$

where $\boldsymbol{d}'_i$ and $\boldsymbol{d}''_i$ are defined in (4) and (5) respectively. Applying the deductions in the previous two cases, $\boldsymbol{d}'_i$ and $\boldsymbol{d}''_i$ are uniformly distributed over $\{0,1\}^n$. Because $\boldsymbol{k}_1$ used in (4) and $\boldsymbol{k}_2$ used in (5) are independent, and these is no relation between $\boldsymbol{d}'_i$ and $\boldsymbol{d}''_i$, then $\boldsymbol{d}'_i \oplus \boldsymbol{d}''_i$ is still uniformly distributed over $\{0,1\}^n$. Consequently, the probability of $\mathcal{A}'$ wrongly outputting "REJECT" is $P_{\mathrm{FP}}$.

Summing all cases up, $\mathcal{A}'$ fails at simulating the reader in one execution at most with probability $\epsilon = \max(P_{\mathrm{FN}}, P_{\mathrm{FP}})$. Thus the probability of $\mathcal{A}'$ correctly simulating the reader in Phase 1 is not less than $1 - q\epsilon$, and adversary $\mathcal{A}'$ impersonates a valid tag at least with probability $(P_{\text{FP}} + \delta)(1 - q\epsilon)$. Therefore, $\mathcal{A}'$ can achieve advantage

$$\mathsf{Adv}_{\mathcal{A}'}^{\text{DET}}(m, \eta, n, \tau) \geq (P_{\text{FP}} + \delta)(1 - q\epsilon) - P_{\text{FP}} = \delta - q\epsilon(P_{\text{FP}} + \delta) \ .$$

With properly chosen parameters such that $P_{\text{FN}}$ and $P_{\text{FP}}$ are negligible, if $\delta$ is non-negligible, then $\mathsf{Adv}_{\mathcal{A}'}^{\text{DET}}(m, \eta, n, \tau)$ is non-negligible. Thus if protocol LCMQ is secure in the DET-model, then it is secure in the MIM-model. □

## 4.4 Security in DET-Model

We first prove that the intractability of the LPN problem implies the *pseudorandomness* of $\boldsymbol{y}_i$ in the LCMQ protocol, and then use it to prove the LCMQ protocol's security in the DET-model.

Let $\mathtt{Bio}_{n,\eta}$ denote the distribution of $n$-bit vector in which each bit independently follows the Bernoulli distribution of parameter $\eta$, $\boldsymbol{k} \in \mathbb{S}_m$, let $\mathcal{D}_{\boldsymbol{k},n,\eta}$ denote the probability distribution of $(m+n)$-bit string:

$$\{\boldsymbol{b} \xleftarrow{\$} \mathbb{S}_m, \boldsymbol{v} \xleftarrow{\$} \mathtt{Bio}_{n,\eta} : (\boldsymbol{b}, \boldsymbol{y} \leftarrow (\mathbf{C}_{\boldsymbol{b}}^{[n\times m]} \circ \boldsymbol{k}^T)^T \oplus \boldsymbol{v})\} \ ,$$

and let $\mathcal{U}_{m+n}$ denote the distribution of $(m+n)$-bit string:

$$\{\boldsymbol{b} \xleftarrow{\$} \mathbb{S}_m, \boldsymbol{y} \xleftarrow{\$} \{0,1\}^n : (\boldsymbol{b}, \boldsymbol{y})\} \ .$$

**Lemma 6.** *Assuming the intractability of the LPN problem, $\mathcal{D}_{\boldsymbol{k},1,\eta}$ and $\mathcal{U}_{m+1}$ are indistinguishable for all PPT algorithms.*

*Proof.* Lemma 1 in [6] has proven that if $\boldsymbol{b}$ is uniformly chosen from $\{0,1\}^m$ in $\mathcal{D}_{\boldsymbol{k},1,\eta}$ and $\mathcal{U}_{m+1}$, and there is no restriction on $m$ having to be a prime number satisfying 2 is a primitive element of $GF(m)$, then $\mathcal{D}_{\boldsymbol{k},1,\eta}$ and $\mathcal{U}_{m+1}$ are indistinguishable for all PPT algorithms, assuming the intractability of the LPN problem. Apparently, the discrepancy between $\boldsymbol{b} \xleftarrow{\$} \mathbb{S}_m$ and $\boldsymbol{b} \xleftarrow{\$} \{0,1\}^m$ is ignorable. As for the requirement of $m$ being prime, it is trivial according to the prime number theorem, which describes the asymptotic distribution of the prime numbers. Even though there is no deterministic number theory result regarding the distribution of a special class of prime number $m$ satisfying 2 is a primitive element of $GF(m)$, according to Artin conjecture [31], the set of such primes is infinite and its density inside the set of primes is equal to Artin's constant, which can be expressed as an infinite product

$$C_{\text{Artin}} = \prod_{q \text{ prime}} \left(1 - \frac{1}{q(q-1)}\right) = 0.3739558136\ldots$$

Therefore, we conclude that $\mathcal{D}_{\boldsymbol{k},1,\eta}$ and $\mathcal{U}_{m+1}$ are indistinguishable for all PPT algorithms. □

**Lemma 7.** *If there is no PPT algorithm capable of distinguishing $\mathcal{D}_{\boldsymbol{k},1,\eta}$ from $\mathcal{U}_{m+1}$, then $\mathcal{D}_{\boldsymbol{k},n,\eta}$ and $\mathcal{U}_{m+n}$ are indistinguishable for all PPT algorithms.*

*Proof.* For $0 \leq i < j \leq n$, let $\mathcal{H}^{i,j}$ denote a hybrid probability distribution

$$\{b_0, b_1, \ldots, b_{m-1}, r_0, r_1, \ldots, r_{i-1}, y_i, y_{i+1}, \ldots, y_{j-1}, r_j, r_{j+1}, \ldots, r_{n-1}\},$$

where all $b_*$'s and $y_*$'s are corresponding to those in $\mathcal{D}_{\boldsymbol{k},n,\eta}$ , all $r_*$'s are random, independent, uniformly selected over $\{0,1\}$, and the convention holds that string $(r_j, r_{j+1}, \dots, r_{n-1})$ is null if $j = n$. In addition, we denote by $p^{i,j}$ the maximal advantage of any PPT algorithm distinguishing $\mathcal{H}^{i,j}$ from $\mathcal{U}_{m+n}$. It is clear that $\mathcal{H}^{0,n} = \mathcal{D}_{\boldsymbol{k},n,\eta}$.

Let $\delta$ be the upper bound of any PPT algorithm's advantage distinguishing $\mathcal{D}_{\boldsymbol{k},1,\eta}$ from $\mathcal{U}_{m+1}$, we will prove $p^{0,n} \leq n\delta$ by induction.

*Basic Case*: Since $\mathcal{H}^{i,i+1}$ is essentially $\mathcal{D}_{\boldsymbol{k}\ggg i,1,\eta}$ inserting $n-1$ random bits, $p^{i,i+1} \leq \delta$.

*Inductive Step*: Assuming $p^{i,j} = (j-i)\eta$. Because $y_i, y_{i+1}, \dots, y_j$ are linearly independent by Lemma 3, we have

$$p^{i,j+1} \leq \max_{i<l<j+1} (p^{i,l} + p^{l,j+1}) = (j+1-i)\delta \ .$$

If $\delta$ is negligible, then $p^{0,n} \leq n\delta$ is also negligible. In other words, $\mathcal{D}_{\boldsymbol{k},n,\eta}$ and $\mathcal{U}_{m+n}$ are indistinguishable for all PPT algorithms if there is no PPT algorithm capable of distinguishing $\mathcal{D}_{\boldsymbol{k},1,\eta}$ from $\mathcal{U}_{m+1}$. □

**Theorem 2.** *If $\mathcal{D}_{\boldsymbol{k},n,\eta}$ and $\mathcal{U}_{m+n}$ are indistinguishable for all PPT algorithms, then all PPT adversaries are only able to attack the LCMQ protocol in the DET-model with a negligible advantage. Therefore, if the LPN problem is intractable, the LCMQ protocol is secure in the DET-model.*

*Proof.* For the LCMQ protocol in Phase 1 of the DET-model, let $\boldsymbol{\kappa}_i = \boldsymbol{k}_2 \oplus \boldsymbol{a}_i$, and $\boldsymbol{\theta}_i = \boldsymbol{y}_i || \boldsymbol{r}_i$. In Section 3.4, we have demonstrated that if $\boldsymbol{\kappa}_i$ is secret, then the encryption $\boldsymbol{z}_i = \mathsf{Enc}(\boldsymbol{\theta}_i, \boldsymbol{\kappa}_i)$ is secure against ciphertext-only attack as long as ciphertext $\boldsymbol{\theta}_i$ is random. If $\mathcal{D}_{\boldsymbol{k},n,\eta}$ and $\mathcal{U}_{m+n}$ are indistinguishable for all PPT algorithms, then $\boldsymbol{\theta}$ is random for any PPT adversary in the DET-model. Even though an adversary $\mathcal{A}$ now can freely choose $\boldsymbol{a}_i$ in the LCMQ protocol, this encryption is still secure against $\mathcal{A}$. To see this point, we can think in this way: During $i$th invocation of Phase 1 in the DET-model, $\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}$ may response to the adversary's challenge $\boldsymbol{a}_i$ with $(\boldsymbol{b}_i \overset{\$}{\leftarrow} \mathbb{S}_m, \boldsymbol{z}_i \overset{\$}{\leftarrow} \mathbb{S}^{\mathrm{e}}_m)$; regardless of any value of $\boldsymbol{a}_i$, the decryption result $\boldsymbol{\theta}_i = \mathsf{Dec}(\boldsymbol{z}_i, \boldsymbol{k} \oplus \boldsymbol{a}_i)$ follows the uniform distribution over $\mathbb{S}^{\mathrm{e}}_m$. In this regard, it simulates the action by $\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}$ from the point of view of $\mathcal{A}$. In other words, we here use a random oracle: the real value of $\boldsymbol{y}_i$ does not come from $(\boldsymbol{b}_i \circ \mathbf{C}^{[m\times n]}_{\boldsymbol{k}_1}) \oplus \boldsymbol{v}_i$, but is determined by the decryption result $\boldsymbol{\theta}_i = \mathsf{Dec}(\boldsymbol{z}_i, \boldsymbol{k}_2 \oplus \boldsymbol{a}_i)$. Clearly, those random responses will not leak any useful information to $\mathcal{A}$.

Now $\mathcal{A}$ proceeds with Phase 2, required to output $(\widehat{\boldsymbol{b}}, \widehat{\boldsymbol{z}})$ corresponding to a random challenge $\widehat{\boldsymbol{a}}$. The only solid chance that $\mathcal{A}$ can pass the authentication is that $\widehat{\boldsymbol{a}}$ appears as at least one of $q$ challenges in Phase 1. This event happens with negligible probability

$$1 - \left(1 - \frac{1}{2^{m-1}-1}\right)^q \approx 1 - e^{-\frac{q}{2^{m-1}-1}} \ .$$

Otherwise, if $\widehat{\boldsymbol{a}}$ is different from all $\boldsymbol{a}_i$'s, as we have proven in Cases Three and Four of Theorem 1, $\mathcal{A}$ only can pass the authentication with probability $P_{\mathrm{FP}}$.

Therefore, no PPT adversary can achieve non-negligible advantage, and the LCMQ protocol is secure in the DET model, assuming the hardness of the LPN problem. □

The LCMQ protocol can be proven secure in an enhanced DET-model—*adaptive-DET-model* in which Phase 1 is identical to that in the DET-model, but in Phase 2 the adversary $\mathcal{A}$ is equipped with more capacity: after receiving the challenge $\widehat{\boldsymbol{a}}$ and before outputting the response, $\mathcal{A}$ is permitted to query $\mathcal{T}_{\boldsymbol{k}_1,\boldsymbol{k}_2,\eta,n}$ $q_2$ times, only with one obvious restriction that $\mathcal{A}$ cannot use $\widehat{\boldsymbol{a}}$ as a challenge. The MIM-model can be extended to an adaptive-MIM-model in the same way. By similar security proofs , the LCMQ protocol is still secure in both adaptive models.

The relation between adaptive-DET-model and DET-Model is analogue to that between adaptive chosen-ciphertext attack (CCA2) and chosen-ciphertext attack (CCA) [32] for public-key encryption schemes. As CCA2 is generally more preferred than CCA, the security guarantee of a protocol in the adaptive-DET-model is more desired in realistic applications. As a matter of fact, those previous HB-like authentication protocols, due to the fact that all of them are vulnerable to the OOV attack, are not secure in the adaptive-DET-model, while they are (provably or presumably) secure in the DET-model.

From the security proofs of the LCMQ protocol, we can see that challenge $\boldsymbol{a}$ does not need to be random; instead, being unique would suffice for $\boldsymbol{a}$. Thus a reader can use a nonce (number used once) as a challenge. In addition, the ID of the tag to be verified can be implicitly embedded as part of a challenge vector, which is very useful in practice.

# 5 Practical Parameters and Discussions

Aimed at providing 80-bit security, we examine the sufficient value of key length $m$ along with noise level $\eta$ in the LCMQ protocol. The LF algorithm [16], as the best algorithm to solving LPN instances by far, renders the HB-like protocols to take $m \geq 512$ with noise level $\eta = 0.25$. In contrast, the proposed LCMQ protocol does not suffer from the limitation since adversaries cannot get access to LPN instances.

According to the LCMQ security proofs in the DET model, $m \geq 81$ would suffice to provide 80-bit security (The additional one bit in the key length is determined by the fact that the parities of $\mathsf{Hwt}(\boldsymbol{k}_1)$ and $\mathsf{Hwt}(\boldsymbol{k}_2)$ are known to the public.) and the precise value of noise level $\eta$ seems insignificant as long as there are noises. Of course, this is only because we use LPN instances as a random oracle in the proof of Theorem 2. Essentially, the fundamental problem that an adversary confronts in the DET-model is described below.

**Definition 7** (LCMQ problem)**.** *Let $m$ be a prime number satisfying that 2 is a primitive element of $GF(m)$, $n < m$, $\eta \in (0, \frac{1}{2})$ be a noise level, $\boldsymbol{k}_1 \xleftarrow{\$} \mathbb{S}_m$ and the parity of $\boldsymbol{k}_1$'s Hamming weight is public, $\boldsymbol{k}_2 \xleftarrow{\$} \mathbb{S}^{\mathrm{e}}_m$. Given $q$ pairs $\langle \boldsymbol{b}_i, \boldsymbol{z}_i = (((\boldsymbol{b}_i \circ \mathbf{C}^{[n\times m]}_{\boldsymbol{k}_1}) \oplus \boldsymbol{v}_i)||\boldsymbol{r}_i) \circ \mathbf{C}^{[(m\text{-}1)\times m]}_{\boldsymbol{k}_2}\rangle$, for $i = 0, 1, \cdots, q-1$, where $\boldsymbol{b}_i \xleftarrow{\$} \mathbb{S}_m$, $\boldsymbol{v}_i \xleftarrow{\$} \mathtt{Bio}_{n,\eta}$, and $\boldsymbol{r}_i \xleftarrow{\$} \{0,1\}^{m-n-1}$, recover $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$.*

The noise-free instances $\langle \boldsymbol{b}_i \xleftarrow{\$} \mathbb{S}_m, \boldsymbol{z}'_i = ((\boldsymbol{b}_i \circ \mathbf{C}^{[n\times m]}_{\boldsymbol{k}_1})||\boldsymbol{r}_i) \circ \mathbf{C}^{[(m\text{-}1)\times m]}_{\boldsymbol{k}_2}\rangle$ constitute a concrete multivariate quadratic system in $2(m-1)$ variants, which is related to another hard problem—the multivariate quadratic (MQ) problem [33].

**Definition 8** (MQ Problem)**.** *Given a system of $w$ quadratic equations in $s$ variables over a finite field, find any valid solutions satisfying all equations.*

Generally speaking, the hardness of MQ problem depends on the relative values of $w$ and $s$. When $w = 1$, it is a trivial case and a solution can be readily retrieved. If $w$ is significantly smaller than $s$, as an underdefined system, finding a solution is fairly easy [34]. When $w$ is much greater than $t$, as an overdefined system, the MQ Problem becomes easy too. Specifically, if there are $\frac{t(t+1)}{2} + 1$ (for $GF(2)$) or $\frac{t(t+3)}{2} + 1$ (for all other finite fields) linearly independent equations available, the MQ problem can be solved by linearization of running time $O(t^6)$ [35]. For general values of $w$ and $s$, the MQ problem is known to be NP-hard, even for quadratic equations oven $GF(2)$ [33, 36]. This problem has been used as the security foundation of cryptographic algorithms, such as the UOV [37] and Sflash [38] signature schemes, and the QUAD [35] stream cipher.

It is clear that the LCMQ problem is essentially a circulant-P2-matrix-based multivariate quadratic system with noise. The name of LCMQ exactly stands for LPN, CM, and MQ. In Section 2.1, we have learned that a little noise ingredient turns a simple solving-linear-equation task into an NP-hard problem, for which only sub-exponential algorithms are discovered. In presence of noise within an MQ system of a nice property of linearly independence in circulant-P2 matrices, we argue that the computational complexity of solving LCMQ problem is close to multiplication of those of solving two hard problems. In fact, the LCMQ problem of parameter $m$, with a same noise level, should be harder than the LPN problem of parameter $(m-1)^2$ because noise $\boldsymbol{v}_i$ in the LCMQ problem is encrypted and then is expanded all over $\boldsymbol{z}_i$. Therefore, we are highly confident that there is no sub-exponential algorithm solving the LCMQ problem, and an adversary only can rely on exhaustive search (matching in the middle) to recover the two $m$-bit keys. Thus $m = 83$ with even small noise level $\eta$ should be sufficient for 80-bit security.

On the other hand, the security proof in the MIM-model demands negligible false rates, ruling out too small choices of $m$. In practice, the LCMQ protocol may use the upper-bounded Bernoulli noise mode, which eliminates the false negative. Although the HB-like protocols with this noise mode are vulnerable to the OOV2 attack, as described in Section 2.2, it can be safely used in the LCMQ protocol, as we have proven the LCMQ protocol is secure against this kind of man-in-the-middle attacks. Recall the false positive rate $P_{\mathrm{FP}} = \sum_{i=0}^{\tau} \binom{n}{i}$, thus $n$ should always be $m-1$ in practice.

Overall, we recommend the parameter set

$$m = 163, n = 162, \eta = 0.08, \tau = 19$$

with the upper-bounded Bernoulli noise mode so as to maintain $P_{\mathrm{FP}} \leq 2^{-80}$, for 80-bit security. With such a small value of $m$, the LCMQ protocol outperforms all previous HB-like protocols in terms of metrics of storage, computation, communication, and implementation while provably prevents all PPT attacks. All of those make it very tempting as a lightweight, reliable, secure entity authentication for RFID systems.

**Two-as-One Variation**. If the performance is ridiculously vital for some applications and thus a smaller value of $m$ is desired while the security level is allowed to slightly sacrifice, a variation of LCMQ protocol, by combining two paralleled authentications as one, might be of help. In this variation, the reader sends two vectors $(\boldsymbol{a}_1, \boldsymbol{a}_2)$ as challenge, and then tag responds with $(\boldsymbol{b}_1, \boldsymbol{z}_1, \boldsymbol{b}_2, \boldsymbol{z}_2)$. Consequently, the reader verifies the authentication by checking $\mathsf{Hwt}((\boldsymbol{b}_1 \circ \mathbf{C}_{\boldsymbol{k}_1}^{[m\times n]}) \oplus \boldsymbol{y}_1) + \mathsf{Hwt}((\boldsymbol{b}_2 \circ \mathbf{C}_{\boldsymbol{k}_1}^{[m\times n]}) \oplus \boldsymbol{y}_2) \leq \tau$.

A parameter set as low as ($m = 83, n = 82, \eta = 0.08, \tau = 19$) with the upper-bounded Bernoulli noise mode can be used. This variation does not affect the security proofs in the DET-model, but the security proof in the MIM-model encounter issues. Even thought it is true that $P_{FP} \leq 2^{-80}$ for that parameter set, $\epsilon$ in Theorem 1 is no longer equal to $\max(P_{\mathrm{FN}}, P_{\mathrm{FP}})$, but notably bigger than it, as an adversary may manipulate only one set of $(\boldsymbol{a}_1, \boldsymbol{b}_1, \boldsymbol{z}_1)$ and $(\boldsymbol{a}_2, \boldsymbol{b}_2, \boldsymbol{z}_2)$. For this variation, we recommend ($m = 107, n = 106, \eta = 0.1, \tau = 30$). Nevertheless, the variation may be attractive for extremely resource-constrained systems.

# 6 Conclusion

In this paper, we present an lightweight, efficient, practical, and secure entity authentication protocol for RFID systems. Built on the learning parity with noise problem, a special type of circulant matrix named circulant-P2 matrix, and the multivariate quadratic problem, the proposed LCMQ protocol outweighs all previous HB-like protocols in terms of the provable security in a general man-in-the-middle model and the tag's computation, storage, and communication costs. As a technique core of the paper, the vector linear independence, gentle properties, and efficient algorithms on matrix operation of circulant-P2 matrix may also be used to construct other cryptographic primitives and secure protocols. That would be an interesting research direction.

# References

[1] A. Juels and S. A. Weis, "Authenticating Pervasive Devices with Human Protocols," in *Advances in Cryptology CRYPTO 2005, Updated version available at: http://www.rsasecurity.com/rsalabs/staff/bios/ajuels/publications/pdfs/lpn.pdf.* LNCS 3621, 2005, pp. 293–308.

[2] J. Bringer and H. Chabanne, "Trusted-HB: A Low-Cost Version of HB+ Secure Against Man-in-the-Middle Attacks," *IEEE Transactions on Information Theory*, vol. 54, no. 9, pp. 4339–4342, 2008.

[3] H. Gilbert, M. J. Robshaw, and Y. Seurin, "HB$^{\#}$: Increasing the Security and Efficiency of HB$^{+}$," in *Advances in Cryptology EUROCRYPT 2008, Full version available at: Cryptology ePrint Archive: Report 2008/028*, 2008.

[4] A. Shamir, "SQUASH A New MAC with Provable Security Properties for Highly Constrained Devices Such as RFID Tags," in *Fast Software Encryption*, 2008, pp. 144–157.

[5] P. Peris-Lopez, J. Hernandez-Castro, J. Estevez-Tapiador, and A. Ribagorda, "M$^2$AP: A Minimalist Mutual-Authentication Protocol for Low-Cost RFID Tags," in *Ubiquitous Intelligence and Computing.* LNCS 4159, 2006, pp. 912–923.

[6] J. Katz and J. Shin, "Parallel and Concurrent Security of the HB and HB+ Protocols," in *Advances in Cryptology - EUROCRYPT 2006.* LNCS 4004, 2006, pp. 73–87.

[7] K. Ouafi, R. Overbeck, and S. Vaudenay, “On the Security of HB$^{\#}$ against a Man-in-the-Middle Attack,” in *Advances in Cryptology - ASIACRYPT 2008*. LNCS 5350, 2008, pp. 108–124.

[8] K. Ouafi and S. Vaudenay, “Smashing SQUASH-0,” in *Advances in Cryptology - EUROCRYPT 2009*, 2009, pp. 300–312.

[9] F. MacWilliams and N. Sloane, *The Theory of Error-Correcting Codes*. North-Holland Amsterdam, 1977.

[10] E. R. Berlekamp, R. J. McEliece, and H. C. van Tilborg, “On the inherent intractability of certain coding problems,” *IEEE Transactions on Information Theory*, vol. 24, no. 3, pp. 384–386, 1978.

[11] J. M. Crawford and M. J. Kearns, “The Minimal Disagreement Parity Problem as a Hard Satisfiability Problem,” Computational Intelligence Research Laboratory and AT&T Bell Labs, Available at http://www.cs.cornell.edu/selman/docs/crawford-parity.pdf, 1995.

[12] J. Håstad, “Some optimal inapproximability results,” in *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, El Paso, Texas, United States, 1997.

[13] O. Regev, “On lattices, learning with errors, random linear codes, and cryptography,” in *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*. Baltimore, MD, USA: ACM, 2005.

[14] A. Blum, A. Kalai, and H. Wasserman, “Noise-tolerant learning, the parity problem, and the statistical query model,” *Journal of the ACM (JACM)*, vol. 50, no. 4, pp. 506–519, 2003.

[15] M. Fossorier, M. Mihaljevi, H. Imai, Y. Cui, and K. Matsuura, “An Algorithm for Solving the LPN Problem and Its Application to Security Evaluation of the HB Protocols for RFID Authentication,” in *Progress in Cryptology - INDOCRYPT 2006*. LNCS 4329, 2006, pp. 48–62.

[16] E. Levieil and P.-A. Fouque, “An Improved LPN Algorithm,” in *Security and Cryptography for Networks*. LNCS 4116, 2006, pp. 348–359.

[17] N. Hopper and M. Blum, “Secure Human Identification Protocols,” in *Advances in Cryptology - ASIACRYPT 2001*. LNCS 2248, 2001, pp. 52–66.

[18] J. Katz and A. Smith, “Analyzing the HB and HB$^{+}$ Protocols in the “Large Error” Case,” Cryptology ePrint Archive, Report 2006/326, Tech. Rep., 2006.

[19] H. Gilbert, M. Robshaw, and H. Sibert, “An Active Attack Against HB$^{+}$ - A Provably Secure Lightweight Authentication Protocol,” Cryptology ePrint Archive: Report 2005/237, Tech. Rep., 2005.

[20] J. Bringer, H. Chabanne, and E. Dottax, “HB$^{++}$: a Lightweight Authentication Protocol Secure against Some Attacks,” in *Second International Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing (SecPerU 2006)*, H. Chabanne, Ed., 2006, pp. 28–33.

[21] D. N. Duc and K. Kim, “Securing HB$^{+}$ Against GRS Man-in-the-Middle Attack,” in *Proceedings of Symposium on Cryptography and Information Security (SCIS 2007)*, Sasebo, Japan, 2007.

[22] J. Munilla and A. Peinado, "HB-MP: A further step in the HB-family of lightweight authentication protocols," *Computer Networks*, vol. 51, no. 9, pp. 2262–2267, 2007.

[23] S. Piramuthu, "HB and Related Lightweight Authentication Protocols for Secure RFID Tag/Reader Authentication," in *CollECTeR Europe Conference*, 2006.

[24] X. Leng, K. Mayes, and K. Markantonakis, "HB-MP+ Protocol: An Improvement on the HB-MP Protocol," in *IEEE International Conference on RFID*, 2008, pp. 118–124.

[25] H. Gilbert, M. J. Robshaw, and Y. Seurin, "Good Variants of HB$^+$ are Hard to Find," in *Financial Crypt 2008*, 2008.

[26] G. Hammouri and B. Sunar, "PUF-HB: A Tamper-Resilient HB Based Authentication Protocol," in *Applied Cryptography and Network Security.* LNCS 5037, 2008, pp. 346–365.

[27] D. Frumkin and A. Shamir, "Un-Trusted-HB: Security Vulnerabilities of Trusted-HB," in *The 5th Workshop on RFID Security (RFIDSec 09)*, 2009.

[28] S. W. Golomb and G. Gong, *Signal Design for Good Correlation: For Wireless Communication, Cryptography, and Radar.* Cambridge University Press, 2004.

[29] R. Lidl and H. Niederreiter, *Finite Fields (Encyclopedia of Mathematics and its Applications).* Addison-Wesley, (Revised version, Cambridge University Press, 1997.), 1983, vol. 20.

[30] R. L. Rivest, "On the invertibility of the XOR of rotations of a binary word," Unpublished draft of July 18, 2009. Revised November 10, 2009, 2009.

[31] D. R. Heath-Brown, "Artin's conjecture for primitive roots," *Quarterly Journal of Mathematics*, vol. 37, no. 1, pp. 27–38, 1986.

[32] D. Bleichenbacher, "Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS #1," in *Advances in Cryptology - CRYPTO'98.* LNCS 1462, 1998, pp. 1–12.

[33] M. R. Garey and D. S. Johnson., *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, 1979, vol. Chapter 7.2 The Polynomial Hierarchy.

[34] N. Courtois, L. Goubin, W. Meier, and J.-D. Tacier, "Solving Underdefined Systems of Multivariate Quadratic Equations," 2002, pp. 211–227.

[35] C. Berbain, H. Gilbert, and J. Patarin, "QUAD: A Practical Stream Cipher with Provable Security," in *Advances in Cryptology - EUROCRYPT 2006.* LNCS 4004, 2006, pp. 109–128.

[36] A. S. Fraenkel and Y. Yesha, "Complexity of solving algebraic equations," *Inf. Process. Lett.*, vol. 10, no. 4/5, pp. 178–179, 1980.

[37] A. Kipnis, J. Patarin, and L. Goubin, "Unbalanced Oil and Vinegar Signature Schemes," in *Advances in Cryptology - EUROCRYPT'99.* LNCS 1592, 1999, pp. 206–222.

[38] M.-L. Akkar, N. Courtois, R. Duteuil, and L. Goubin, "A Fast and Secure Implementation of Sflash," in *PKC 2003.* LNCS 2567, 2002, pp. 267–278.