$GHB^{\#}$: A Provably Secure HB-Like Lightweight Authentication Protocol

Panagiotis Rizomiliotis and Stefanos Gritzalis

Dep. of Information and Communication Systems Engineering
University of the Aegean
Karlovassi, Samos, GR 83200, Greece
{prizomil,sgritz}@aegean.gr

Abstract. RFID technology constitutes a fundamental part of what is known as the Internet of Things; i.e. accessible and interconnected machines and everyday objects that form a dynamic and complex environment. In order to secure RFID tags in a cost-efficient manner, the last few years several lightweight cryptography-based tag management protocols have been proposed. One of the most promising proposals is the HB^{+} protocol, a lightweight authentication protocol that is supported by an elegant security proof against all passive and a subclass of active attackers based on the hardness of the Learning Parity with Noise (LPN) problem. However, the HB^{+} was shown to be weak against active man-in-the-middle (MIM) attacks and for that several variants have been proposed. Yet, the vast majority of them has been broken.

In this paper, we introduce a new variant of the HB^{+} protocol that can provably resist MIM attacks. More precisely, we improve the security of another recently proposed variant, the $HB^{\#}$ protocol by taking advantage of the properties of the well studied Gold power functions. The new authentication protocol is called $GHB^{\#}$ and its security can be reduced to the LPN problem. Finally, we show that the $GHB^{\#}$ remains practical and lightweight.

1 Introduction

Radio Frequency Identification (RFID) technology constitutes a fundamental part and key enabler of what is known as the Internet of Things (IoT); i.e. accessible and interconnected machines and everyday objects that form a dynamic and complex environment. In the IoT vision, the Internet extends into our everyday lives through a wireless network of uniquely identifiable objects or 'things'. RFID tags are much "smarter" and more efficient than the classical barcode and can provide us with the data needed to manage 'things', unmanageable until today; thus rendering RFID the most pervasive technology in human history. Each physical object is accompanied by a rich, globally accessible virtual object that contains both current and historical information on that object's physical properties, origin, ownership. When available ubiquitously and in real time, this information can dramatically streamline how we manufacture, distribute, manage, and recycle our goods.

F. Bao, P. Samarati, and J. Zhou (Eds.): ACNS 2012, LNCS 7341, pp. 489–506, 2012.

Applications ranging from inventory monitoring, and payment systems to supply-chain management and smart home devices are already taking advantage of the RFID technology. However, this rapid proliferation of RFID tags raises several security and privacy concerns. Given also that, in order to sustain the pervasiveness, the cost of the tag must remain as low as possible; i.e. space, as well as, peak and average power consumption limitations must be instituted, it was identified early on that new lightweight cryptographic protocols have to be deployed for their management.

In this context, several new lightweight schemes have been proposed in the last few years ([1]), mainly for secure tag authentication. Amongst the proposed solutions, the most prominent ones are the authentication schemes that are based on the conjectured hardness of the *Learning Parity in the presence of Noise (LPN)* problem, which is closely related to the well-studied problem of decoding random linear codes.

Definition 1. *(LPN Problem) Let* $\boldsymbol{A}$ *be a random* $(q \times k)$*-binary matrix, let* $\boldsymbol{x}$ *be a random* k*-bit vector, let* $\eta \in (0, 1/2)$ *be a noise parameter, and let* $\boldsymbol{\nu}$ *be a random* q*-bit vector such that* $wt(\boldsymbol{\nu}) \leq \eta q$*. Given* $\boldsymbol{A}$*,* η*, and* $\boldsymbol{z} = \boldsymbol{A} \cdot \boldsymbol{x}^t + \boldsymbol{\nu}^t$*, find a* k*-bit vector* $\boldsymbol{y}^t$ *such that* $wt(A \cdot \boldsymbol{y}^t + \boldsymbol{z}) \leq \eta q$*.*

In [12], Juels and Weis proposed HB^+, a symmetric key authentication scheme, inspired by *HB* ([11]), the work of Hopper and Blum for the secure identification of human beings. The HB^+ has very simple circuit representation, as it performs only a few dot-product and bit exclusive-or computations. In more detail, the prover (the Tag) and the verifier (the Reader) exchange random binary vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, and the prover based on this exchanged information and two secret vectors $\boldsymbol{x}$ and $\boldsymbol{y}$, produces and transmits to the verifier one bit $z = \boldsymbol{ax} \oplus \boldsymbol{by} + \nu$, where ν is one bit that follows a Bernoulli distribution with parameter $\eta \in (0, \frac{1}{2})$. The verifier accepts $z = \boldsymbol{ax} \oplus \boldsymbol{by}$. This basic interaction has soundness $\frac{1}{2}$ and completeness $1 - \eta$ and it is improved via sequential or parallel composition, i.e. the verifier accepts if after r repetitions of the basic round at most t times the condition is not satisfied.

Certainly, the most interesting feature of the protocol is the elegant proof that supports its security analysis. Specifically, in [12], a concrete reduction of the LPN problem to the security of the HB^+ protocol in two attack models was shown. In the first model the attacker is passive and can only eavesdrop the communication between the prover and the verifier, while in the second model she is active and she can also send queries to the prover. The original proof was further simplified and extended in [13], [14].

However, the above described attack models do not include more powerful adversaries, like the ones that can manipulate messages exchanged between the reader and the tag. Thus, it came as no surprise that soon after the introduction of the HB^+, it was shown ([7]) that there is a simple man-in-the-middle (MIM) attack that can easily reveal the secret vectors $\boldsymbol{x}$ and $\boldsymbol{y}$. Motivated by this MIM attack, several variants of HB^+ have been proposed ([6],[8], [2], [24], [22], [3], [20], [16], [25]). However, most of these schemes have been shown to be weak against a MIM attacker.

Recently, the first two HB^{+} variants that can provably resist MIM attacks have been proposed and both are based on equivalent to LPN problems. In [15], Kiltz et al. built on the Subspace LWE problem to construct two secure Message Authentication Code (MAC) schemes. However, both these schemes require the application of Pairwise Independent Permutation, while the secret keys are very long. In [4], Bosley et al., introduced the Learning Subspace with Noise (LSN) problem and they showed the equivalence between the LPN and LSN. Based on the LSN problem, they introduced an authentication protocol and they proved its security against MIM attacks. This protocol is equivalent to the second MAC scheme introduced in [15].

1.1 Our Contribution

In this paper, we propose a new variant of the HB^{+} protocol that can provably resist all known MIM attacks. More precisely, we improve the security of another recently proposed variant, the $HB^{\#}$ protocol ([8]), by taking advantage of the properties of the Gold power functions. The $HB^{\#}$ protocol was introduced by Gilbert et al. and it was provably secure against the attack that succeeded against HB^{+}([7]). However, Oaufi et al. [21] presented another MIM attack on $HB^{\#}$. The main objective of our work is to enhance the security of $HB^{\#}$ by adding some nonlinear components without increasing significantly its complexity.

The idea of using nonlinear functions to built secure LPN-based authentication protocols is not new. In [2], Bringer et al. proposed the HB^{++} protocol, a modified version of the HB^{+} protocol that could resist the attack in [7] using a specific family of nonlinear multi-output Boolean functions, the Gold functions. Gold functions can be efficiently implemented in hardware, they have been extensively studied in the literature and they possess very good cryptographic properties, like high nonlinearity and good derivative behaviour, and for that they constitute an excellent choice. However, HB^{++} was shown to be weak [9]. A more recent attempt to introduce nonlinear HB-like protocols by Madhavan et al. [18] was also unsuccessful ([23]).

Our protocol, called $GHB^{\#}$, is the first nonlinear variant of HB^{+} that it is provably resistant against MIM attacks. Our reduction is using rewinding, like in the case of the HB^{+} and $HB^{\#}$ protocol and the security is based on the hardness of the LPN problem. Moreover, we show that, despite the use of nonlinear functions, the $GHB^{\#}$ protocol remains as practical and lightweight as its direct ancestor $HB^{\#}$.

1.2 Outline

The paper is organized as follows. In Section 2, we establish the necessary background on vectorial Boolean functions with emphasis in the family of Gold functions. In the same section, we describe the $HB^{\#}$ protocol. In Section 3, we present the new authentication protocol and in Section 4, we provide efficient implementation guidelines and we compute the overall complexity. In Section 5, we provide the security analysis and we prove that the new protocol is secure against active

attackers that can interrogate a tag and/or modify all the exchanged messages between a tag and the reader. Finally, conclusions and topics for further research can be found in Section 6.

2 Background

2.1 Gold Functions

Vectorial Boolean functions constitute fundamental building blocks for many cryptographic algorithms and have been extensively studied in the literature. In this paper, we use a specific family of such functions, the so called *Gold functions* that possess very good cryptographic properties ([10], [5]). First we introduce some notation and then we present the necessary background.

Let $\mathbb{F}_2$ be the finite field with two elements and $\mathcal{B}_{n,m}$ the set of vectorial Boolean function with n inputs and m outputs; i.e. the set of multi-output Boolean from $\mathbb{F}_2^n$ to $\mathbb{F}_2^m$. We use normal, bold and capital bold letters, x, $\boldsymbol{x}$ and $\boldsymbol{M}$ to denote single elements, vectors and matrices, respectively. Also, normal and capital bold letters are used for single input (univariate) and multi input Boolean functions, respectively. The Hamming weight $\mathrm{wt}(\boldsymbol{x})$ of a vector $\boldsymbol{x} = [x(0), x(1), \cdots, x(n-1)]$ is the number of nonzero elements. Finally, $\mathbf{0}_m$ denotes the all zeros vector of length m and for real numbers $\eta, \psi \in \Re$, $]\eta, \psi[= \{x \in \Re | \eta < x < \psi\}$.

Definition 2. *([5]) A vectorial function $\boldsymbol{F} \in \mathcal{B}_{n,m}$ is balanced if it takes all values $\boldsymbol{y} \in \mathbb{F}_2^m$ the same number of times; i.e. 2^{n-m} times.*

Definition 3. *([5]) The derivative of a vectorial Boolean function $\boldsymbol{F} \in \mathcal{B}_{n,m}$ is defined as $D_{\boldsymbol{a}}\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{F}(\boldsymbol{x}) + \boldsymbol{F}(\boldsymbol{x} + \boldsymbol{a})$, $\boldsymbol{a} \in \mathbb{F}_2^n$ and $\boldsymbol{a} \neq \mathbf{0}_n$.*

Definition 4. *A vectorial Boolean function $\boldsymbol{F} \in \mathcal{B}_{n,m}$ is called almost perfect nonlinear (APN) if and only if for every $\boldsymbol{a} \in \mathbb{F}_2^n$, $\boldsymbol{a} \neq \mathbf{0}_n$ and $\boldsymbol{b} \in \mathbb{F}_2^m$ the equation $D_{\boldsymbol{a}}\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{b}$ has zero or 2 solutions.*

In [10], R. Gold introduced the so-called *Gold functions*, the power functions $x \rightarrow x^d$ on the field $\mathbb{F}_{2^n}$, where n odd and $d = 2^i + 1$, with $\gcd(i, n) = 1$ and $1 \leq i < \frac{n-1}{2}$. Gold proved that these univariate polynomials are APN functions, with very high nonlinearity, balanced and have quadratic algebraic degree. (Note that Gold functions have good cryptographic properties when n is an even integer.)

Gold functions have been defined and analysed as univariate functions, but it is well known, that they can be easily transformed to a vectorial Boolean function. Let $\{\alpha_0, \cdots, \alpha_{n-1}\}$ be a basis of $\mathbb{F}_{2^n}$ over $\mathbb{F}_2$, then any element $x \in \mathbb{F}_{2^n}$ can be written as $x = \sum_{i=0}^{n-1} x_i \alpha_i$, $x_i \in \mathbb{F}_2$. In this paper, we are going to use a normal basis $\{\gamma^{2^0}, \gamma^{2^1}, \cdots, \gamma^{2^{n-1}}\}$ of $\mathbb{F}_{2^n}$ over $\mathbb{F}_2$, where $\gamma \in \mathbb{F}_{2^n}$. It is well known that there is such basis for any $m > 1$ ([17]). Note that depending on the choice of the basis, the mapping from the univariate functions to the vectorial

Boolean functions differs. Thus, a Gold function $g = x^d$ can be written as a multi input Boolean function $\boldsymbol{G}$ as follows,

$$\boldsymbol{G}(x_0, x_1, \cdots, x_{n-1}) = g(\sum_{i=0}^{n-1} x_i \gamma^{2^i}).$$

Similarly, the output of the function can be written as a linear combination of the elements of the basis.

Definition 5. *([5]) Two functions are affine equivalent if one derives from the other with some left and right compositions with an affine permutation.*

We denote by $\mathcal{G}(n, d)$ the set of all multi-input vectorial Boolean functions that are affine equivalent to a Gold function with n inputs and outputs and with exponent d. That is, if $\boldsymbol{G}$ is the Gold multi-input Boolean function, then for every vectorial Boolean function $\boldsymbol{\Phi} \in \mathcal{G}(n, d)$, there are affine permutations $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ such that $\boldsymbol{\Phi} = \boldsymbol{P}_1 \circ \boldsymbol{G} \circ \boldsymbol{P}_2$. Every function $\boldsymbol{\Phi} \in \mathcal{G}(n, d)$ has all the aforementioned properties of Gold functions; i.e. $\boldsymbol{\Phi}$ is APN, balanced and quadratic ([5]).

Since every $\boldsymbol{\Phi} \in \mathcal{G}(n, d)$ is quadratic, it can be written as

$$\boldsymbol{\Phi}(x(0), \cdots, x(n-1)) = \boldsymbol{L}(x(0), \cdots, x(n-1)) \oplus \boldsymbol{Q}(x(0), \cdots, x(n-1)),$$

where $\boldsymbol{L}$ is a linear vectorial Boolean function and $\boldsymbol{Q}$ a purely quadratic vectorial Boolean function. We denote by $I_{\boldsymbol{\Phi}} \subset \{0, 1, \cdots, n-1\}$ the smallest subset of the input variable indexes of $\boldsymbol{\Phi}$, such that

$$\boldsymbol{Q}(x(0), x(1), \cdots, x(n-1)) = \mathbf{0}_m,$$

for all $\boldsymbol{x} \in K(\boldsymbol{\Phi})$, where

$$K(\boldsymbol{\Phi}) = \{\boldsymbol{x} \in \mathbb{F}_2^n |\ x(i) = 0,\ \forall\ i \in I_{\boldsymbol{\Phi}}\}.$$

Clearly, $K(\boldsymbol{\Phi})$ is the subspace of equations $x_i = 0$, $i \in I_{\boldsymbol{\Phi}}$, and the restriction of $\boldsymbol{\Phi}$ to this subspace is a linear vectorial function, i.e. $\boldsymbol{\Phi}(\boldsymbol{x}_1 \oplus \boldsymbol{x}_2) = \boldsymbol{\Phi}(\boldsymbol{x}_1) \oplus \boldsymbol{\Phi}(\boldsymbol{x}_2)$, for $\boldsymbol{x}_1, \boldsymbol{x}_2 \in K(\boldsymbol{\Phi})$.

2.2 The $HB^{\#}$ protocol

In this section, we briefly describe the $HB^{\#}$ protocol ([8]). We try to apply, as possible, the established notation. We use $x \xleftarrow{\$} X$ to denote the assignment to x of a value sampled from the uniform distribution on the finite set X. We use $Ber(\eta)$ to denote the Bernoulli distribution with parameter η, meaning that a bit $\nu \in Ber(\eta)$, then $Pr[\nu = 1] = \eta$ and $Pr[\nu = 0] = 1 - \eta$. A vector $\boldsymbol{\nu}$ randomly chosen among all the vectors of length m, such that $\boldsymbol{\nu}(i) \in Ber(\eta)$ and $\eta \in (0, 1/2)$, for $0 \leq i \leq m-1$, is denoted as $\boldsymbol{\nu} \xleftarrow{\$} Ber(m, \eta)$. Finally, we use $\boldsymbol{b} \xleftarrow{\$} \mathbb{F}_2^k$ to denote a random binary vector $\boldsymbol{b}$ of length k.

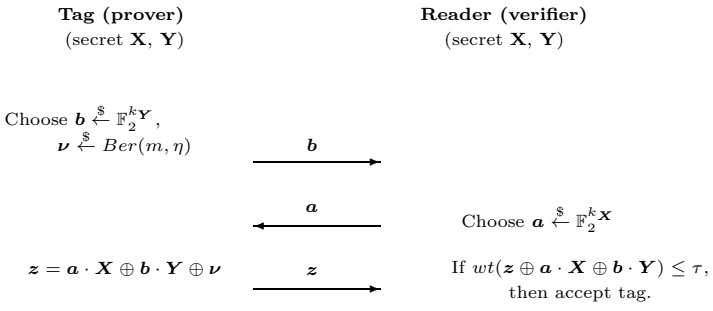

Fig. 1. The $HB^{\#}$ protocol

The $HB^{\#}$ protocol can be seen as a natural matrix extension of the HB^{+} protocol, where the prover and the verifier, instead of vectors, they share two binary matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ with size $k_{\boldsymbol{X}} \times m$ and $k_{\boldsymbol{Y}} \times m$ respectively. The protocol is again a three pass one, but now the verifier and prover need only one round to interact (Fig. 1). Like the HB^{+} protocol, the $HB^{\#}$ has low computational complexity $\mathcal{O}(k_{\boldsymbol{X}} \cdot m + k_{\boldsymbol{Y}} \cdot m)$, while it reduces the transmission costs to $(k_{\boldsymbol{Y}} + k_{\boldsymbol{X}} + m)$ bits in total and it provides more practical error rates. However, at the same time, it needs more memory bits for the secret keys, as the tag has to store the two secret matrices, i.e. $(k_{\boldsymbol{X}} \cdot m + k_{\boldsymbol{Y}} \cdot m)$ bits in total.

Security analysis. The $HB^{\#}$ protocol was designed to resist the attack introduced in [7] and for that it is supported by a proof of security against attackers that can modify only the messages sent by the reader to the tag during an execution of the protocol. To prove the security of the scheme, a natural matrix-based extension of the *HB* problem was introduced, the *MHB puzzle*.

Definition 6. *((k, m, η, q)-MHB puzzle, [8]) Let $\eta \in (0, 1/2)$ and m and q be polynomials in k. On input the security parameter 1^k, the puzzle generator G draws a random secret $(k \times m)$-binary matrix $\boldsymbol{X}$, q random vectors $(\boldsymbol{a}_1, \cdots, \boldsymbol{a}_q)$ of length k, computes for $1 \leq i \leq q$ the set of answers $\boldsymbol{z}_i = \boldsymbol{a}_i \cdot \boldsymbol{X} + \boldsymbol{\nu}_i$, where each bit of $\boldsymbol{\nu}_i$ is 1 with probability η, and draws a random vector a of length k constituting the challenge to the adversary. It outputs $\{(\boldsymbol{a}_i, \boldsymbol{z}_i)\}_{1 \leq i \leq q}$ and $\boldsymbol{a}$. The solver returns a vector $\boldsymbol{z}$. The secret is $\boldsymbol{X}$, and the verifier V accepts, if and only if, $\boldsymbol{z} = \boldsymbol{a} \cdot \boldsymbol{X}$.*

Using the theory of weakly verifiable puzzles the hardness of this extended problem was proved (Lemma 1) and a concrete reduction of the MHB puzzle to the security of the $HB^{\#}$ protocol was provided in [8]. We are going to use the MHB puzzle in our proposal as well.

Lemma 1. *([8]) Assume the hardness of the LPN problem. Then, the MHB puzzle is $(1 - \frac{1}{2^m})$-hard.*

Attack against $HB^{\#}$. In [21], it was shown that the protocol is not secure against a more general MIM attack. That is, when the attacker can manipulate

all the messages exchanged between a legitimate tag and the reader, and not only the messages sent by the reader, there is a key recovery attack that she can mount. In a few words, the attack goes as follows. The attacker obtains a valid triplet $(\hat{\boldsymbol{b}}, \hat{\boldsymbol{a}}, \hat{\boldsymbol{z}})$; i.e. a triplet that satisfies $\mathrm{wt}(\hat{\boldsymbol{z}} \oplus \hat{\boldsymbol{b}} \cdot \boldsymbol{X} \oplus \hat{\boldsymbol{a}} \cdot \boldsymbol{Y}) \leq \tau$. Using this triplet to modify several executions of the protocol between the reader and the same legitimate tag, the success of the i-th authentication depends on the condition $\mathrm{wt}(\boldsymbol{\nu}_i \oplus \hat{\boldsymbol{\nu}}) \leq \tau$, where $\hat{\boldsymbol{\nu}} = \hat{\boldsymbol{z}} \oplus \hat{\boldsymbol{b}} \cdot \boldsymbol{X} \oplus \hat{\boldsymbol{a}} \cdot \boldsymbol{Y}$ and $\boldsymbol{\nu}_i \in Ber(m, \eta)$ is the masking vector used by the tag. The overall success probability leaks information on the Hamming weight of $\hat{\boldsymbol{\nu}}$. In the second phase of the attack, the attacker modifies one bit of $\hat{\boldsymbol{y}}$ and computes the Hamming weight of the new vector $\hat{\boldsymbol{\nu}}'$. Thus, one bit of the vector $\hat{\boldsymbol{\nu}}$ can be estimated from the difference between $\mathrm{wt}(\hat{\boldsymbol{\nu}})$ and $\mathrm{wt}(\hat{\boldsymbol{\nu}}')$. After repeating the same procedure several times a set of linear equations is constructed involving $\hat{\boldsymbol{b}} \cdot \boldsymbol{X} + \hat{\boldsymbol{a}} \cdot \boldsymbol{Y}$ and the solution of this linear system reveals the secret keys $\boldsymbol{X}$ and $\boldsymbol{Y}$.

During the application of this attack many unsuccessful executions of protocol occur and a mechanism that detects this abnormal behaviour could provide a sufficient countermeasure. However, such a mechanism is not built-in property of the protocol and a new protocol has to be proposed. In the following section, we will show that the $GHB^{\#}$ is such a proposal.

3 The $GHB^{\#}$ protocol

In this section, we introduce a nonlinear variant of the $HB^{\#}$ protocol, a one round symmetric key protocol called the $GHB^{\#}$. Following the notation introduced in Section 2.2, the tag and the reader share two secret binary matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ of size $k_{\boldsymbol{X}} \times m$ and $k_{\boldsymbol{Y}} \times m$ respectively. The single round of the protocol appears in Fig. 2.

The tag and the reader exchange the randomly selected vectors $\boldsymbol{b}$ and $\boldsymbol{a}$ of length $k_{\boldsymbol{Y}}$ and $k_{\boldsymbol{X}}$, respectively. Then, the tag computes and sends the vector $\boldsymbol{z} = \boldsymbol{\Phi}(\boldsymbol{a} \cdot \boldsymbol{X}) \oplus \boldsymbol{\Phi}(\boldsymbol{b} \cdot \boldsymbol{Y}) \oplus \boldsymbol{\nu}$ of length m, where $\boldsymbol{\nu} \in Ber(m, \eta)$. If $\mathrm{wt}(\boldsymbol{z} \oplus \boldsymbol{\Phi}(\boldsymbol{a} \cdot \boldsymbol{X}) \oplus \boldsymbol{\Phi}(\boldsymbol{b} \cdot \boldsymbol{Y})) \leq \tau$, then the reader accepts the tag as authentic. Otherwise, the tag is rejected. The threshold $\tau = um$, where $u \in \left]\eta, \frac{1}{2}\right[$.

The function $\boldsymbol{\Phi}$ is publicly known and it can be any multi-input function that belongs to $\mathcal{G}(m, d)$; i.e. it is affine equivalent to a Gold multi-input vectorial Boolean function $\boldsymbol{G}$, as defined in Section 2.1. The choice of the specific univariate Gold function x^d used for the construction of $\boldsymbol{G}$ does not influence the security of the protocol or its complexity. In Section 4, we give design directives for the efficient hardware implementation of $\boldsymbol{\Phi}$ and we compute the hardware cost that brings to the protocol.

The error rates of the new protocol are computed similarly to the ones of $HB^{\#}$. In more detail, the false rejection rate P_{FR} of the protocol; i.e. the probability to reject a legitimate tag, equals the probability $\mathrm{wt}(\boldsymbol{\nu}) > \tau$ and it is given by

$$P_{FR} = \sum_{i=\tau+1}^{m} \binom{m}{i} \eta^i (1-\eta)^{m-i}.$$

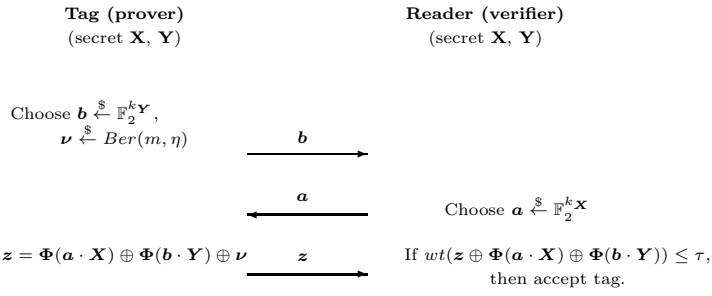

Fig. 2. The $GHB^{\#}$ protocol

It is common practice, in most *HB*-like protocols, to use an extra step in which $\boldsymbol{\nu}$ is used only when its Hamming weight is at most τ; i.e. the completeness error is $P_{FR} = 0$.

Finally, the false acceptance rate P_{FA}; i.e. the probability to accept a randomly selected response $\boldsymbol{z}$, equals the probability a binary vector of length m to have Hamming weight at most τ. That is that, the soundness error is given by:

$$P_{FA} = \sum_{i=0}^{\tau} \binom{m}{i} 2^{-m}.$$

4 Complexity Analysis and Implementation Issues

Next, we compute the overall storage, communication and computation complexity. The main challenge for the *GHB* protocol is to efficiently implement the function $\boldsymbol{\Phi}$ and for that we provide implementation directives.

Storage Complexity. The memory cost for the tag; i.e. the storage for the two secret matrices, is $(k_{\boldsymbol{X}} \cdot m + k_{\boldsymbol{Y}} \cdot m)$ bits.

Communication Complexity. The protocol requires $(k_{\boldsymbol{Y}} + k_{\boldsymbol{X}} + m)$ bits to be transfered in total.

Computational Complexity. We concentrate on the computationally weaker of the two entities; i.e. the tag. We distinguish two main operations, the multiplication of the random vectors with the secret matrices and the application of the function $\boldsymbol{\Phi}$ for the computation of $\boldsymbol{z}$. The two multiplications require in total approximately $\mathcal{O}(k_{\boldsymbol{X}} \cdot m + k_{\boldsymbol{Y}} \cdot m)$ basic binary operations. This is, also, the computational complexity of the $HB^{\#}$ protocol. For the implementation of $\boldsymbol{\Phi}$, we propose the following approach.

Let $\{\gamma, \gamma^2, \gamma^{2^2}, \cdots, \gamma^{2^{m-1}}\}$ be a normal basis of $\mathbb{F}_{2^m}$ over $\mathbb{F}_2$, where $\gamma \in \mathbb{F}_{2^m}$. It is well known that there is such basis for any $m > 1$ ([17]). The implementation of a Gold function x^{2^i+1} requires one exponentiation and one multiplication. By $\otimes$ we demote the multiplication of two field elements of the field. When a field element $x \in \mathbb{F}_{2^m}$ is represented in normal form; i.e. $x = \sum_{i=0}^{m-1} x(i)\gamma^{2^i}$, the

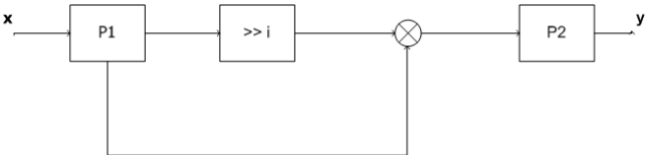

Fig. 3. The implementation of the function $\boldsymbol{\Phi}$

exponentiation can be performed by right cyclic shift of the binary representation $\boldsymbol{x}$. Thus, x^{2^i} is obtained by simply shifting $\boldsymbol{x}$ to the right by i steps.

Concerning the multiplication $x \cdot x^{2^i}$, one of the most straightforward ways to perform efficiently a normal basis multiplication is the one proposed by Massey and Omura [19]. More precisely, for each normal basis there is an $m \times m$ matrix M called the multiplication matrix of the normal basis and if $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is, respectively, the binary vector representation of the elements $x_1, x_2 \in \mathbb{F}_{2^m}$ with respect to the basis, then the binary representation of the product $y = x_1 x_2$ is computed as $y(m-1-i) = h(x_1^{2^i}, x_2^{2^i})$, for $0 \leq i \leq m-1$, where $h(x_1, x_2) = \boldsymbol{x}_1 M \boldsymbol{x}_2^T$. The complexity of the operation is determined by the number C_N of ones of M. It is proved that the number of required AND and XOR gates is C_N and $C_N - 1$ respectively. When an optimal normal basis is used, then $C_N = 2m + 1$, and we have the least possible complexity.

Finally, since the function $\boldsymbol{\Phi}$ belongs to $\mathcal{G}(m, d)$, any function affine equivalent to a Gold function can be used. This function can be implemented from the proposed construction for the Gold functions by multiplying the input and the output vectors by the $m \times m$ matrix that corresponds to the left and right affine permutations, respectively. If P_1 and P_2 are these two matrices, then the total computation is given in Fig. 3. The complexity for each one of the permutations can vary from constant to at most $\mathcal{O}(m^2)$. Thus, the computational complexity of the $\boldsymbol{\Phi}$ function varies from $\mathcal{O}(m)$ to $\mathcal{O}(m^2)$ depending on the choice of the permutations and the total computational complexity of the protocol is at most $\mathcal{O}(k_{\boldsymbol{X}} \cdot m + k_{\boldsymbol{Y}} \cdot m + m^2)$.

To summarize the $GHB^{\#}$ protocol has the same communication and storage complexity as the its predecessor $HB^{\#}$, while it requires at least $\mathcal{O}(m)$ (and at most $\mathcal{O}(m^2)$) more basic binary computations (Table 1). In Table 2, we use practical parameters that have been proposed for the $HB^{\#}$ protocol in order to compare the efficiency of the two protocols.

Table 1. Complexity Comparison between $GHB^{\#}$ and $HB^{\#}$

	Security	Stor. Compl.	Comm. Compl.	Comp. Compl.
$HB^{\#}$	Active	$\mathcal{O}(k_{\boldsymbol{X}} \cdot m + k_{\boldsymbol{Y}} \cdot m)$	$\mathcal{O}(k_{\boldsymbol{X}} + k_{\boldsymbol{Y}} + m)$	$\mathcal{O}(k_{\boldsymbol{X}} \cdot m + k_{\boldsymbol{Y}} \cdot m)$
$GHB^{\#}$	MIM	$\mathcal{O}(k_{\boldsymbol{X}} \cdot m + k_{\boldsymbol{Y}} \cdot m)$	$\mathcal{O}(k_{\boldsymbol{X}} + k_{\boldsymbol{Y}} + m)$	$\mathcal{O}(k_{\boldsymbol{X}} \cdot m + k_{\boldsymbol{Y}} \cdot m + m)$

Table 2. Comparison between the $GHB^{\#}$ and $HB^{\#}$ protocols for practical parameters

$k_{\boldsymbol{X}}$	$k_{\boldsymbol{Y}}$	m	η	τ	Stor. $GHB^{\#}$	Stor. $HB^{\#}$	Comm. $GHB^{\#}$	Comm. $HB^{\#}$	Comp. $GHB^{\#}$	Comp. $HB^{\#}$
80	512	1163	0.25	405	$688k$	$688k$	$1.7k$	$1.7k$	$689k$	$688k$
80	512	441	0.125	113	$261k$	$261k$	$1k$	$1k$	$261k$	$261k$

5 Security Analysis

5.1 Definition of Security Models

Following the notation used in [8], we use $\mathcal{R}_{\boldsymbol{X},\boldsymbol{Y},\tau}$ to denote the algorithm that it is run by the reader (verifier) and $\mathcal{T}_{\boldsymbol{X},\boldsymbol{Y},\eta}$ the one run by a legitimate tag (prover). We use $\boldsymbol{X} \xleftarrow{\$} \mathbb{F}_2^{(m_1,m_2)}$ to indicate the random selection of a $m_1 \times m_2$ binary matrix $\boldsymbol{X}$.

All the attacks against HB^{+} and its variants are active ones; i.e. the attacker can interact with the reader and/or the tag and change some of the messages exchanged between the two legitimate entities. We distinguish two models of security, the $DET - model$ and the $MIM - model$. In each of the models the adversary runs in two stages. In the first stage she has some interaction with the prover and/or the verifier and in the second she interacts only with the verifier and wins if the verifier returns *accept*. We define the advantage of an attacker $\mathcal{A}$ against $GHB^{\#}$ in the models as the overhead success probability over P_{FA}; i.e. the best possible soundness error we can hope for is the success probability when the attacker does not perform any action during the first phase of the attack and just sends a randomly selected $\boldsymbol{z}$ in the second phase. Note that, P_{FA} is negligible for the chosen values of τ and for security $m = \Theta(k)$, where k is the security parameter. In the $DET - model$ the attacker interacts only with an honest prover for a polynomial number of times. More precisely,

Definition 7. *(DET-model). In the $DET - model$ the attack is carried in two phases:*

- **Phase 1.** *Adversary $\mathcal{A}$ interacts q times with the honest tag $\mathcal{T}_{\boldsymbol{X},\boldsymbol{Y},\eta}$. More precisely, on the i-th invocation, $\mathcal{T}_{\boldsymbol{X},\boldsymbol{Y},\eta}$ internally generates a random blinding vector $\boldsymbol{b}_i$, it takes a challenge $\boldsymbol{a}_i$ from $\mathcal{A}$ as input and outputs $\boldsymbol{z}_i = \boldsymbol{\Phi}(\boldsymbol{a}_i \cdot \boldsymbol{X}) \oplus \boldsymbol{\Phi}(\boldsymbol{b}_i \cdot \boldsymbol{Y}) \oplus \boldsymbol{\nu}_i$ and sends the message to $\mathcal{A}$.*
- **Phase 2.** *Adversary $\mathcal{A}$ interacts with the reader $\mathcal{R}_{\boldsymbol{X},\boldsymbol{Y},\tau}$ trying to impersonate the tag $\mathcal{T}_{\boldsymbol{X},\boldsymbol{Y},\eta}$ with advantage*

$$Adv_{\mathcal{A}}^{DET}(k_{\boldsymbol{X}}, k_{\boldsymbol{Y}}, m, \eta, \tau, q) =$$
$$Pr[\boldsymbol{X} \xleftarrow{\$} \mathbb{F}_2^{(k_{\boldsymbol{X}},m)}, \boldsymbol{Y} \xleftarrow{\$} \mathbb{F}_2^{(k_{\boldsymbol{Y}},m)}, \mathcal{A}^{\mathcal{T}_{\boldsymbol{X},\boldsymbol{Y},\eta}}(1^k) : \langle \mathcal{A}, \mathcal{R}_{\boldsymbol{X},\boldsymbol{Y},\tau} \rangle = ACC] - P_{FA}.$$

In the $MIM - model$ the attacker can interact with both the prover and the verifier and learn the verifier's decision, *accept* or *reject*.

Definition 8. *(MIM-model). In the $MIM - model$ the attack is carried in two phases and the adversary can manipulate all messages exchanged between the tag and the reader:*

– **Phase 1.** *On the i-th invocation,* $\mathcal{T}_{\boldsymbol{X},\boldsymbol{Y},\eta}$ *internally generates a random blinding vector* $\boldsymbol{b}_i$ *and sends it to the adversary* $\mathcal{A}$*. The reader* $\mathcal{R}_{\boldsymbol{X},\boldsymbol{Y},\tau}$ *receives a modified blinding vector* $\hat{\boldsymbol{b}}_i = \bar{\boldsymbol{b}} \oplus \boldsymbol{b}_i$ *from* $\mathcal{A}$*. Then, the reader generates a challenge vector* $\boldsymbol{a}_i$ *and sends it to the adversary* $\mathcal{A}$*. The tag receives a modified challenge vector* $\hat{\boldsymbol{a}}_i = \bar{\boldsymbol{a}} \oplus \boldsymbol{a}_i$ *from* $\mathcal{A}$ *and replies with* $\boldsymbol{z}_i = \boldsymbol{\Phi}(\hat{\boldsymbol{a}}_i \cdot \boldsymbol{X}) \oplus \boldsymbol{\Phi}(\boldsymbol{b}_i \cdot \boldsymbol{Y}) \oplus \boldsymbol{\nu}_i$*,* $\boldsymbol{\nu}_i \in Ber(m, \eta)$*. The reader receives a modified vector* $\hat{\boldsymbol{z}}_i = \bar{\boldsymbol{z}} \oplus \boldsymbol{z}_i$ *and if* $\mathrm{wt}(\hat{\boldsymbol{z}}_i \oplus \boldsymbol{\Phi}(\boldsymbol{a}_i \cdot \boldsymbol{X}) \oplus \boldsymbol{\Phi}(\hat{\boldsymbol{b}}_i \cdot \boldsymbol{Y})) \leq \tau$*, then the reader outputs accept. Otherwise, it outputs reject. The adversary interferes for q executions of the of the protocol.*
– **Phase 2.** *Adversary* $\mathcal{A}$ *interacts with the reader* $\mathcal{R}_{\boldsymbol{X},\boldsymbol{Y},\tau}$ *trying to impersonate the tag* $\mathcal{T}_{\boldsymbol{X},\boldsymbol{Y},\eta}$ *with advantage*

$$Adv_{\mathcal{A}}^{MIM}(k_{\boldsymbol{X}}, k_{\boldsymbol{Y}}, m, \eta, \tau, q) = Pr[\boldsymbol{X} \xleftarrow{\$} \mathbb{F}_2^{(k_{\boldsymbol{X}},m)},$$
$$\boldsymbol{Y} \xleftarrow{\$} \mathbb{F}_2^{(k_{\boldsymbol{Y}},m)}, \mathcal{A}^{\mathcal{T}_{\boldsymbol{X},\boldsymbol{Y},\eta},\mathcal{R}_{\boldsymbol{X},\boldsymbol{Y},\tau}}(1^k) : \langle \mathcal{A}, \mathcal{R}_{\boldsymbol{X},\boldsymbol{Y},\tau} \rangle = ACC] - P_{FA}.$$

Note 1. As we have all ready described in Section 1, most variants of the HB^+ protocol are secure under the $DET-model$. However, the attack presented in [7], the *GRS attack*, against the HB^+ protocol was easily applied to most of these variants ([9]). This attack is included in the $MIM-model$, but the adversary in the first phase is limited to modify only the messages that the reader sends. The $HB^{\#}$ protocol was the first one provably secure against the GRS attack, but it was shown to be weak under the $MIM-model$ ([21]).

5.2 Security Proofs

Next we prove that the $GHB^{\#}$ protocol is secure under both the $DET-model$ and the $MIM-model$ given the hardness of the LPN problem. Our reduction is using rewinding, like in the case of the HB^+ and $HB^{\#}$ protocols and the security is based on the MHB puzzle. We say that a function in x is negligible if it vanishes faster then the inverse of any polynomial in x.

Lemma 2. *There is* $\hat{\boldsymbol{\Phi}} \in \mathcal{G}(m, d)$*, such that for each* $\boldsymbol{x}_1, \boldsymbol{x}_2 \in K(\hat{\boldsymbol{\Phi}})$*, it holds that* $\hat{\boldsymbol{\Phi}}(\boldsymbol{x}_1 \oplus \boldsymbol{x}_2) = \hat{\boldsymbol{\Phi}}(\boldsymbol{x}_1) \oplus \hat{\boldsymbol{\Phi}}(\boldsymbol{x}_2) = \left(\boldsymbol{y} || \boldsymbol{0}_{|I_{\hat{\boldsymbol{\Phi}}}|}\right)$*, for some* $\boldsymbol{y} \in \mathbb{F}_2^{m-|I_{\hat{\boldsymbol{\Phi}}}|}$.

Proof. The linearity derives directly for the definition of the subspace $K(\hat{\boldsymbol{\Phi}})$. Next, we prove the existence of such $\hat{\boldsymbol{\Phi}}$.

By definition, every $\boldsymbol{x} \in K(\boldsymbol{\Phi})$ has $|I_{\boldsymbol{\Phi}}|$ entries fixed to 0 and every Boolean function $\boldsymbol{\Phi} \in \mathcal{G}(m, d)$ is a linear function in the subspace $\boldsymbol{x} \in K(\boldsymbol{\Phi})$. That is, for $\boldsymbol{x} \in K(\boldsymbol{\Phi})$, $\boldsymbol{\Phi}$ can be seen as a function with $m - |I_{\boldsymbol{\Phi}}|$ input variables and m outputs.

Since the number of inputs is less than the number of outputs, $|I_{\boldsymbol{\Phi}}|$ of the output bits can written as a linear combination of the other $m - |I_{\boldsymbol{\Phi}}|$; i.e. there is a linear transformation $\boldsymbol{M}$ that can be applied to the output of $\boldsymbol{\Phi}$ and results to $|I_{\hat{\boldsymbol{\Phi}}}|$ zero output bits, for $\boldsymbol{x} \in K(\boldsymbol{\Phi})$. Also, as any permutation $\boldsymbol{P}$ of the outputs is acceptable, these zero outputs can be put last. From the composition $\hat{\boldsymbol{\Phi}} = \boldsymbol{P} \circ \boldsymbol{M} \circ \boldsymbol{\Phi}$, of $\boldsymbol{\Phi}$ with the linear transformation and the permutation with $\hat{\boldsymbol{\Phi}}$, the result follows. □

Theorem 1. *(Security in the DET-model) If there is an adversary $\mathcal{A}^{\#}$ that can attack the GHB$^{\#}$ protocol, with parameters $(k_{\boldsymbol{X}}, k_{\boldsymbol{Y}}, m^{\#}, \eta, \tau)$, in the DET-model by interacting with an honest tag $q^{\#}$ times, running time $T^{\#}$ and achieving advantage at least $\delta^{\#}$, then, there is an adversary $\mathcal{A}$ that can solve the $(k_{\boldsymbol{Y}}, m^{\#} - |I_{\boldsymbol{\Phi}}|, \eta, q)$ − MHB puzzle with parameters in running time $T = 2qT^{\#}$ and success probability $\delta > (\frac{1}{2^m} + \frac{\delta^{\#}}{4})$, where $q = m^{\#}q^{\#}L(2 + \log_2 q^{\#})$, $L \geq \frac{2}{\epsilon'^2} \ln(\frac{1}{1-e^{-\frac{ln2}{m}}})$, $\epsilon' = \frac{{\delta^{\#}}^3}{16}(\frac{1}{2} - \frac{\tau}{m})^3(\frac{1}{2} - \frac{1}{k_{\boldsymbol{X}}})$ and $\boldsymbol{\Phi}$ is the Gold linearly equivalent function used in GHB$^{\#}$.*

Proof. The adversary $\mathcal{A}$ has obtained q pairs $(\boldsymbol{b}_i, \boldsymbol{z}_i)$ from the *MHB* puzzle generator, where $\boldsymbol{z}_i = \boldsymbol{b}_i \cdot \boldsymbol{Y} \oplus \boldsymbol{\nu}_i$, $1 \leq i \leq q$, and $\boldsymbol{Y}$ is a randomly selected $k \times m$ binary matrix. Let $\boldsymbol{b}$ be the k-bit challenge vector of the puzzle; i.e. she has to compute $\boldsymbol{z} = \boldsymbol{b} \cdot \boldsymbol{Y}$. We will show how the adversary $\mathcal{A}$ can solve the *MHB* puzzle using the algorithm of the adversary $\mathcal{A}^{\#}$. The proof is a modified version of the proof introduced for the security reduction in the DET-model in [8].

During the attack, $\mathcal{A}^{\#}$ interrogates a legitimate tag and $\mathcal{A}$ simulates the behaviour of the tag algorithm. The function $\boldsymbol{\Phi}$ is chosen to be one that has the properties described in Lemma 2. Let $\boldsymbol{X}^{\#}$ and $\boldsymbol{Y}^{\#}$ be the two secret matrices shared between the tag and the reader. The $k_{\boldsymbol{X}} \times m^{\#}$ matrix $\boldsymbol{X}^{\#}$ has all the entries randomly selected except the i-th column $\boldsymbol{X}^{\#}(:, i)$, for all $i \in I_{\boldsymbol{\Phi}}$, and the s-th row $\boldsymbol{X}^{\#}(s, :)$, for a random row $1 \leq s \leq k_{\boldsymbol{X}}$, that are all zero. Similarly, the matrix $\boldsymbol{Y}^{\#}$ is a $k_{\boldsymbol{Y}} \times m^{\#}$ binary matrix that has also the i-th column $\boldsymbol{Y}^{\#}(:, i)$ all zero, for all $i \in I_{\boldsymbol{\Phi}}$, while all the other entries of the matrix are randomly selected.

$\mathcal{A}$ divides the q pairs, that she has obtained from the *MHB* puzzle, in m sets with $Lq^{\#}(1 + r)$ pairs each. L is the number of estimations for each bit of the vector $\boldsymbol{z}$ that the adversary gets from each set and r defines the size of a pool of extra pairs that she can use in each estimation. The vector $\boldsymbol{e}$ of length m stores intermediate values and it is initialised $\boldsymbol{e} = \boldsymbol{0}_m$. We use '||' to denote the concatenation of vectors.
For $j_0 = 0$ to $m - 1$ do:

1. For $j_1 = 0$ to $L - 1$ do:
   (a) Phase I: For $j_2 = 0$ to $q^{\#} - 1$ do:
      i. $\mathcal{A}$ selects random bit $c \in \mathbb{F}_2$ and sends $\boldsymbol{b}^{\#}_{j_0,j_1,j_2} = \boldsymbol{b}_{(j_0L+j_1)q^{\#}+j_2} \oplus c \cdot \boldsymbol{b}$.
      ii. Let $\boldsymbol{a}^{\#}_{j_0,j_1,j_2}$ be the challenge vector send by the attacker $\mathcal{A}^{\#}$.
      iii. If $\boldsymbol{a}^{\#}_{j_0,j_1,j_2}(r) = c$, then $\mathcal{A}$ replies with

$$\boldsymbol{z}^{\#}_{j_0,j_1,j_2} = \boldsymbol{\Phi}(\boldsymbol{a}^{\#}_{j_0,j_1,j_2}\boldsymbol{X}^{\#}) \oplus \boldsymbol{z}^{\#}_{(j_0L+j_1)q^{\#}+j_2}, \tag{1}$$

where the second term is given by

$$\boldsymbol{z}^{\#}_{(j_0L+j_1)q^{\#}+j_2} = \left(\boldsymbol{z}_{(j_0L+j_1)q^{\#}+j_2} || \boldsymbol{\mu}^{\#}_{(j_0L+j_1)q^{\#}+j_2}\right),$$

and $\boldsymbol{\mu}^{\#}_{(j_0L+j_1)q^{\#}+j_2} \in Ber(|I_{\boldsymbol{\Phi}}|, \eta)$. Otherwise, she rewinds adversary $\mathcal{A}^{\#}$ to the beginning of the current query and uses a new pair. If the available pairs are exhausted, guess the message $\boldsymbol{z}^{\#}_{(j_0L+j_1)q^{\#}+j_2}$.

(b) Phase II: $\mathcal{A}$ proceeds to the second, impersonation, phase of the DET-model attack.
   i. Adversary $\mathcal{A}^{\#}$ sends the commitment vector $\boldsymbol{b}'$.
   ii. Adversary $\mathcal{A}$ chooses two challenges $\boldsymbol{a}'_0$ and $\boldsymbol{a}'_1$ with complement values at the s-th bit; i.e. $\boldsymbol{a}'_0(s) \oplus \boldsymbol{a}'_1(s) = 1$.
   iii. Adversary $\mathcal{A}$ transmits $\boldsymbol{a}'_0$ and gets the reply $\boldsymbol{z}'_0$.
   iv. Adversary $\mathcal{A}$ rewinds the attacker $\mathcal{A}^{\#}$ just after the transmission of $\boldsymbol{b}'$ and sends $\boldsymbol{a}'_1$ to get the reply $\boldsymbol{z}'_1$.
   v. Adversary $\mathcal{A}$ computes the sum

$$\boldsymbol{z}' = \boldsymbol{z}'_0 \oplus \boldsymbol{z}'_1 \oplus \boldsymbol{\Phi}\left(\sum_{i=1, i\neq s}^{k_{\boldsymbol{X}}} \boldsymbol{a}'_0(i)\boldsymbol{X}^{\#}(:,i)\right) \oplus \boldsymbol{\Phi}\left(\sum_{i=1, i\neq s}^{k_{\boldsymbol{X}}} \boldsymbol{a}'_1(i)\boldsymbol{X}^{\#}(:,i)\right) \tag{2}$$

   and adds the value of the j_0-th bit to $\boldsymbol{e}(j_0)$, i.e $\boldsymbol{e}(j_0) = \boldsymbol{e}(j_0) + \boldsymbol{z}'(j_0)$.

2. The estimation of the bit $\boldsymbol{z}(j_0)$ is given by majority decision; i.e.

$$\boldsymbol{z}(j_0) = \boldsymbol{e}(j_0)/L \mod 2.$$

We will show that the attacker $\mathcal{A}$ successfully simulates a tag algorithm that uses a $k_{\boldsymbol{Y}} \times m^{\#}$ binary matrix $\boldsymbol{Y}^{\#}$ that has the i-th column $\boldsymbol{Y}^{\#}(:,i)$ all zero, for all $i \in I_{\boldsymbol{\Phi}}$, i.e. from Lemma 2, $\boldsymbol{\Phi}(\boldsymbol{b}_1 \cdot \boldsymbol{Y}^{\#}) \oplus \boldsymbol{\Phi}(\boldsymbol{b}_2 \cdot \boldsymbol{Y}^{\#}) = \boldsymbol{\Phi}((\boldsymbol{b}_1 \oplus \boldsymbol{b}_2) \cdot \boldsymbol{Y}^{\#})$ and the last $|I_{\boldsymbol{\Phi}}|$ of the output of are all zero. Finally, the secret matrix $\boldsymbol{Y}^{\#}$ is such that $\boldsymbol{\Phi}(\boldsymbol{b}_i \cdot \boldsymbol{Y}^{\#}) = \left(\boldsymbol{b}_i \cdot \boldsymbol{Y} \;||\; \boldsymbol{0}_{|I_{\boldsymbol{\Phi}}|}\right)$.

When $\boldsymbol{a}^{\#}_{j_0,j_1,j_2}(r) = 0$, the reply (1) of $\mathcal{A}$ can be written as

$$\begin{aligned}\boldsymbol{z}^{\#}_{j_0,j_1,j_2} &= \boldsymbol{\Phi}(\boldsymbol{a}^{\#}_{j_0,j_1,j_2}\boldsymbol{X}^{\#}) \oplus \left(\boldsymbol{z}_{(j_0L+j_1)q^{\#}+j_2} || \boldsymbol{\mu}^{\#}_{(j_0L+j_1)q^{\#}+j_2}\right) \\ &= \boldsymbol{\Phi}(\boldsymbol{a}^{\#}_{j_0,j_1,j_2}\boldsymbol{X}^{\#}) \oplus \left(\boldsymbol{b}_{(j_0L+j_1)q^{\#}+j_2} \cdot \boldsymbol{Y} || \boldsymbol{0}_{I_{\boldsymbol{\Phi}}}\right) \\ &\oplus \left(\boldsymbol{\nu}_{(j_0L+j_1)q^{\#}+j_2} || \boldsymbol{\mu}^{\#}_{(j_0L+j_1)q^{\#}+j_2}\right)\end{aligned}$$

and since $\boldsymbol{\nu}_{(j_0L+j_1)q^{\#}+j_2} \in Ber(m - |I_{\boldsymbol{\Phi}}|, \eta)$ and $\boldsymbol{\mu}^{\#}_{(j_0L+j_1)q^{\#}+j_2} \in Ber(|I_{\boldsymbol{\Phi}}|, \eta)$, it holds that, $\left(\boldsymbol{\nu}_{(j_0L+j_1)q^{\#}+j_2} || \boldsymbol{\mu}^{\#}_{(j_0L+j_1)q^{\#}+j_2}\right) \in Ber(m, \eta)$.
Due to the specific choice of the matrix $\boldsymbol{X}^{\#}$, the function $\boldsymbol{\Phi}$ is restricted to $K(\boldsymbol{\Phi})$ and behaves like a linear function. Thus, for $\boldsymbol{a}^{\#}_{j_0,j_1,j_2}(r) = 1$, the reply (1) of $\mathcal{A}$ can be written as

$$\begin{aligned}\boldsymbol{z}^{\#}_{j_0,j_1,j_2} &= \boldsymbol{\Phi}\left(\sum_{i=1, i\neq s}^{k_{\boldsymbol{X}}} \boldsymbol{a}^{\#}_{j_0,j_1,j_2}(i)\boldsymbol{X}^{\#}(:,i)\right) \oplus \boldsymbol{\Phi}(\boldsymbol{X}^{\#}(s,:)) \\ &\oplus (\boldsymbol{b} \cdot \boldsymbol{Y} || \boldsymbol{0}_{I_{\boldsymbol{\Phi}}}) \oplus \left(\boldsymbol{b}_{(j_0L+j_1)q^{\#}+j_2} \cdot \boldsymbol{Y} || \boldsymbol{0}_{I_{\boldsymbol{\Phi}}}\right) \oplus \left(\boldsymbol{\nu}_{(j_0L+j_1)q^{\#}+j_2} || \boldsymbol{\mu}^{\#}_{(j_0L+j_1)q^{\#}+j_2}\right)\end{aligned}$$

and, again, it holds that, $\left(\boldsymbol{\nu}_{(j_0L+j_1)q^{\#}+j_2} || \boldsymbol{\mu}^{\#}_{(j_0L+j_1)q^{\#}+j_2}\right) \in Ber(m, \eta)$. From the above, we have that only the first $m - |I_{\boldsymbol{\Phi}}|$ entries of $\boldsymbol{z}'$ given in (2) provide an estimation of the puzzle's answer $\boldsymbol{z}$.

Next, we compute necessary amount of estimations L for the majority strategy to give the correct value of $\boldsymbol{z}$ with significant advantage δ. For the computation of r, L, δ, we follow mainly the approach presented in [8].

For each of the L estimations of a single bit of $\boldsymbol{z}$, the attacker has $r \cdot q^{\#}$ extra pairs, where $r = 1 + log_2 q^{\#}$ and these pairs will be sufficient with probability more than $1/2$.

The guess of $\boldsymbol{z}$ is correct if either both $\boldsymbol{z}'_0$ and $\boldsymbol{z}'_1$ are correct or if both are false. From [12], the probability this to happen is greater than $p = \frac{1}{2} + \frac{\epsilon^3}{2} - \frac{\epsilon^3+1}{k_{\boldsymbol{X}}}$, where $\epsilon = \frac{\delta^{\#}}{2}(\frac{1}{2} - \frac{\tau}{m})$. Since, the probability of guessing the message $\boldsymbol{z}^{\#}_{(j_0 L + j_1)q^{\#} + j_2}$ is less than $1/2$, the probability of correct guessing one of the m bits of $\boldsymbol{z}$ is lower bounded by $1/4 + p/2 \geq 1/2 + \epsilon'$, where $\epsilon' = \frac{\epsilon^3}{4} - \frac{\epsilon^3+1}{2k_{\boldsymbol{X}}}$.

Finally, from Chernoff bound on the majority of the L experiments, the guess of all m bits is lower bounded by $p^{MHB} \geq \left(1 - e^{-\frac{L\epsilon'^2}{2}}\right)^m$. Thus, for the probability p^{MHB} to be greater than $1/2$, the number of experiments must be at least

$$L \geq \frac{2}{\epsilon'^2} \ln\left(\frac{1}{1 - e^{-\frac{\ln(2)}{m}}}\right).$$

□

From Theorem 1, any efficient adversary achieving a noticeable advantage $\delta^{\#}$ against the $GHB^{\#}$ protocol in the DET-model can be turned into an efficient solver of the MHB puzzle with a success probability greater than $\frac{1}{2^m}$ by $\frac{\delta^{\#}}{4}$, and, from Lemma 1, this contradicts the hardness assumption of the LPN problem.

Lemma 3. *Let $\boldsymbol{\Phi} \in \mathcal{G}(m, d)$ and let $\mathcal{X}$ and $\mathcal{Y}$ be two sets of randomly selected binary vectors of length m with cardinality $|\mathcal{X}| = 2^{k_{\boldsymbol{X}}}$ and $|\mathcal{Y}| = 2^{k_{\boldsymbol{Y}}}$, respectively, and $k_{\boldsymbol{X}} \leq k_{\boldsymbol{Y}}$. Then, for given $(\bar{\boldsymbol{b}}, \bar{\boldsymbol{a}}, \bar{\boldsymbol{z}}) \in \mathbb{F}_2^m \times \mathbb{F}_2^m \times \mathbb{F}_2^m$, the probability*

$$p(d) = Pr\left[\mathrm{wt}(D_{\bar{\boldsymbol{a}}}\boldsymbol{\Phi}(\boldsymbol{x}) \oplus D_{\bar{\boldsymbol{b}}}\boldsymbol{\Phi}(\boldsymbol{y}) \oplus \bar{\boldsymbol{z}}) \leq d\right],\ 1 \leq d \leq n$$

where $\boldsymbol{x} \in \mathcal{X}$ and $\boldsymbol{y} \in \mathcal{Y}$, is upper bounded by

$$p(d) \leq 2^{-\min(k_{\boldsymbol{X}}, m) + 2 + mH(\frac{d}{m})}.$$

$H(s) = s \cdot \log_2(\frac{1}{s}) - (1-s) \cdot \log_2(\frac{1}{1-s})$ is the entropy function.

Proof. From the Definition 4 of APN functions, for given $\bar{\boldsymbol{a}}$ there is a subset $\mathcal{S}_{\bar{\boldsymbol{a}}} \subseteq \mathbb{F}_2^m$, such that for every $\boldsymbol{c} \in \mathcal{S}_{\bar{\boldsymbol{a}}}$ there is $\boldsymbol{x} \in \mathcal{X}$ satisfying $D_{\bar{\boldsymbol{a}}}\boldsymbol{\Phi}(\boldsymbol{x}) = \boldsymbol{c}$. Since each $\boldsymbol{c} \in \mathcal{S}_{\bar{\boldsymbol{a}}}$ can appear at most twice, it holds that $\min(2^{k_{\boldsymbol{X}}-1}, 2^{m-1}) \leq |\mathcal{S}_{\bar{\boldsymbol{a}}}| \leq \min(2^{k_{\boldsymbol{X}}}, 2^{m-1})$. Similarly, we define $\mathcal{S}_{\bar{\boldsymbol{b}}} \subseteq \mathbb{F}_2^m$, such that for every $\boldsymbol{c} \in \mathcal{S}_{\bar{\boldsymbol{b}}}$ there is $\boldsymbol{y} \in \mathcal{Y}$ such that $D_{\bar{\boldsymbol{b}}}\boldsymbol{\Phi}(\boldsymbol{y}) = \boldsymbol{c}$, with $\boldsymbol{c} \in \mathcal{S}_{\bar{\boldsymbol{b}}}$ and $\min(2^{k_{\boldsymbol{Y}}-1}, 2^{m-1}) \leq |\mathcal{S}_{\bar{\boldsymbol{b}}}| \leq \min(2^{k_{\boldsymbol{Y}}}, 2^{m-1})$.

All the sums $\boldsymbol{c} = \boldsymbol{c}_1 \oplus \boldsymbol{c}_2$ of a given vector $\boldsymbol{c}_1 \in \mathcal{S}_{\bar{\boldsymbol{a}}}$ with any $\boldsymbol{c}_2 \in \mathcal{S}_{\bar{\boldsymbol{b}}}$, are different. Thus, the sum $D_{\bar{\boldsymbol{a}}}\boldsymbol{\Phi}(\boldsymbol{x}) \oplus D_{\bar{\boldsymbol{b}}}\boldsymbol{\Phi}(\boldsymbol{y}) = \boldsymbol{c}$, with $\boldsymbol{x} \in \mathcal{X}$ and $\boldsymbol{y} \in \mathcal{Y}$ can take the same value $\boldsymbol{c}$ with probability at most

$$\frac{2}{2^{\min(k_{\boldsymbol{X}}, m)}} \frac{2}{2^{\min(k_{\boldsymbol{Y}}, m)}} 2^{\max(|\mathcal{S}_{\bar{\boldsymbol{a}}}|, |\mathcal{S}_{\bar{\boldsymbol{b}}}|)} \leq \frac{1}{2^{\min(k_{\boldsymbol{X}}, m) - 2}}.$$

Given that the number of binary vectors of length m and Hamming weight less than d is $\sum_{i=0}^{d} \binom{m}{i} \leq 2^{mH(\frac{d}{m})}$, the probability $D_{\bar{\boldsymbol{a}}}\boldsymbol{\Phi}(\boldsymbol{x}) \oplus D_{\bar{\boldsymbol{b}}}\boldsymbol{\Phi}(\boldsymbol{y}) = \boldsymbol{c}$ to have Hamming weight less than $d \leq n$ is upper bounded by $2^{-\min(k_{\boldsymbol{X}},m)+2+mH(\frac{d}{m})}$. To conclude, clearly, the constant value $\bar{\boldsymbol{z}}$ does not influence this probability. □

Theorem 2. *(Security in the MIM-model) If there is an adversary $\mathcal{A}^{\#}$ that can attack the $GHB^{\#}$ protocol with parameters $(k_{\boldsymbol{X}}, k_{\boldsymbol{Y}}, m^{\#}, \eta, \tau)$ in the MIM-model by modifying $q^{\#}$ protocol executions between an honest tag and the reader, with running time $T^{\#}$ and achieving advantage at least $\delta^{\#}$, then, there is an adversary $\mathcal{A}$ that can attack the $GHB^{\#}$ protocol in the DET-model with the same parameters by interrogating an honest tag $q^{\#}$ times, with running time at most $T^{\#}$ and with advantage at least $\delta \geq \delta^{\#} - (P_{FA} + \delta^{\#})q^{\#}p_r$, where p_r is a negligible function.*

Proof. The attacker $\mathcal{A}$ has a legitimate tag at her disposal that she can interrogate. We will show how $\mathcal{A}$ can attack $GHB^{\#}$ protocol in the DET-model using the algorithm that the adversary $\mathcal{A}^{\#}$ executes.

During the MIM attack, $\mathcal{A}^{\#}$ is modifying all messages between the legitimate tag and reader. While, the adversary $\mathcal{A}$ has access to an honest tag, she has to simulate the behavior of the reader. More precisely, her strategy goes as follows.

1. $\mathcal{A}$ receives from the honest tag $\mathcal{T}_{\boldsymbol{X},\boldsymbol{Y},\eta}$ a blinding vector $\boldsymbol{b}$ and sends this vector to the $\mathcal{A}^{\#}$.
2. $\mathcal{A}^{\#}$ produces a new blinding vector $\hat{\boldsymbol{b}} = \boldsymbol{b} \oplus \bar{\boldsymbol{b}}$ and sends this vector to the simulated reader; i.e. to the adversary $\mathcal{A}$.
3. $\mathcal{A}$ produces a random challenge vector $\boldsymbol{a}$, on behalf of the reader and sends it to $\mathcal{A}^{\#}$.
4. $\mathcal{A}^{\#}$ produces a new challenge vector $\hat{\boldsymbol{a}} = \boldsymbol{a} \oplus \bar{\boldsymbol{a}}$ and sends this vector to the honest tag, via $\mathcal{A}$.
5. the tag responds with $\boldsymbol{z}$ and $\mathcal{A}$ sends the response to $\mathcal{A}^{\#}$.
6. $\mathcal{A}^{\#}$ produces a new response vector $\hat{\boldsymbol{z}} = \boldsymbol{z} \oplus \bar{\boldsymbol{z}}$ and sends this vector to the simulated reader; i.e. to adversary $\hat{\mathcal{A}}$.
7. If the triplet $(\bar{\boldsymbol{b}}, \bar{\boldsymbol{a}}, \bar{\boldsymbol{z}})$ is all-zero, the simulated reader; i.e. $\mathcal{A}$, notifies adversary $\mathcal{A}^{\#}$ that the tag has been accepted. Otherwise, it is rejected.

The previous steps are repeated $q^{\#}$ times. The adversary $\mathcal{A}$ impersonates the tag to an honest reader in the DET-attack, by using the second phase of $\mathcal{A}^{\#}$.

The probability of successfully simulating a reader's behavior depends on the ability of the adversary to simulate the last step; i.e. the acceptance or rejection of the tag. Let p_{auth} be this probability, then the overall probability of the attack is given by

$$p_{MIM} = p_{auth} \cdot (P_{FA} + \delta). \tag{3}$$

We will compute p_{MIM}. In order for the attack to be successful, the adversary $\mathcal{A}$ must be able to simulate the reader's behavior for $q^{\#}$ consecutive executions of the protocol. Let p_r be the probability to fail in one execution. Then,

$$p_{auth} = (1 - q^{\#} \cdot p_r). \tag{4}$$

The probability of false rejecting a tag when $(\bar{\boldsymbol{b}}, \bar{\boldsymbol{a}}, \bar{\boldsymbol{z}})$ is all zero is P_{FR}. That is, $p_r \geq P_{FR} = 0$, since we have assumed that the Hamming weight of ν is checked. The value p_r is also defined by the probability that the condition $\mathrm{wt}(\hat{\boldsymbol{z}} \oplus \boldsymbol{\Phi}(\boldsymbol{a} \cdot \boldsymbol{X}) \oplus \boldsymbol{\Phi}(\hat{\boldsymbol{b}} \cdot \boldsymbol{Y})) \leq \tau$ is satisfied when $(\bar{\boldsymbol{b}}, \bar{\boldsymbol{a}}, \bar{\boldsymbol{z}}) \neq (\mathbf{0}_{k_{\boldsymbol{Y}}}, \mathbf{0}_{k_{\boldsymbol{X}}}, \mathbf{0}_m)$. The sum can be written as

$$\begin{aligned}\hat{\boldsymbol{z}} \oplus \boldsymbol{\Phi}(\boldsymbol{a} \cdot \boldsymbol{X}) \oplus \boldsymbol{\Phi}(\hat{\boldsymbol{b}} \cdot \boldsymbol{Y}) &= \boldsymbol{\Phi}(\hat{\boldsymbol{a}} \cdot \boldsymbol{X}) \oplus \boldsymbol{\Phi}(\boldsymbol{b} \cdot \boldsymbol{Y}) \oplus \boldsymbol{\nu} \oplus \bar{\boldsymbol{z}} \oplus \boldsymbol{\Phi}(\boldsymbol{a} \cdot \boldsymbol{X}) \oplus \boldsymbol{\Phi}(\hat{\boldsymbol{b}} \cdot \boldsymbol{Y}) \\ &= D_{\bar{\boldsymbol{a}}}\boldsymbol{\Phi}(\boldsymbol{a} \cdot \boldsymbol{X}) \oplus D_{\bar{\boldsymbol{b}}}\boldsymbol{\Phi}(\boldsymbol{b} \cdot \boldsymbol{Y}) \oplus \boldsymbol{\nu} \oplus \bar{\boldsymbol{z}}.\end{aligned}$$

Let $\boldsymbol{y}_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}} = D_{\bar{\boldsymbol{a}}}\boldsymbol{\Phi}(\boldsymbol{a} \cdot \boldsymbol{X}) \oplus D_{\bar{\boldsymbol{b}}}\boldsymbol{\Phi}(\boldsymbol{b} \cdot \boldsymbol{Y}) \oplus \bar{\boldsymbol{z}}$ and let $\beta_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}}$ be the Hamming weight of $\boldsymbol{y}_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}}$. Then, $m - \beta_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}}$ bits of $\boldsymbol{y}_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}} \oplus \boldsymbol{\nu}$ follow a Bernoulli distribution of parameter η and the rest $\beta_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}}$ bits follow a Bernoulli distribution of parameter $1-\eta$. That is, the Hamming weight $\mathrm{wt}(\boldsymbol{y}_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}} \oplus \boldsymbol{\nu})$ follows a binomial distribution of expected value $\mu = (m - \beta_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}})\eta + (1 - \beta_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}}\eta)$ and variance $\sigma^2 = m\eta(1-\eta)$. Since, the expected value is a function of $\beta_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}}$ we can easily verify that for $\beta_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}} \geq 1 + \lfloor \frac{\tau - \eta m}{1-2\eta} \rfloor$, it holds that $\mu > \tau$.

When, $\mu > \tau$; i.e. $\beta_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}} \geq 1 + \lfloor \frac{\tau - \eta m}{1-2\eta} \rfloor$ from the Chernoff bound we have that $\mathrm{wt}(\boldsymbol{y}_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}} \oplus \boldsymbol{\nu}) < \tau$ with probability $p_1 < e^{-\frac{(\mu-\tau)^2}{2\mu}}$. When, $\mu \leq \tau$; i.e. $\beta_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}} < 1 + \lfloor \frac{\tau - \eta m}{1-2\eta} \rfloor$, trivially we have that $\mathrm{wt}(\boldsymbol{y}_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}} \oplus \boldsymbol{\nu}) < \tau$ with probability p_2. By combining the two cases, $\mathrm{wt}(\boldsymbol{y}_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}} \oplus \boldsymbol{\nu}) \leq \tau$ with probability

$$\hat{p}_r = p_1 \cdot Pr[\mu > \tau] + p_2 \cdot Pr[\mu \leq \tau] \leq e^{-\frac{(\mu-\tau)^2}{2\mu}} \cdot Pr[\mu > \tau] + P[\mu \leq \tau].$$

From Lemma 3, we have that $P\left[\beta_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}} \leq 1 + \lfloor \frac{\tau - \eta m}{1-2\eta} \rfloor\right]$ is upper bounded by

$$Pr\left[\beta_{\bar{\boldsymbol{a}},\bar{\boldsymbol{b}},\bar{\boldsymbol{z}}} \leq 1 + \lfloor \frac{\tau - \eta m}{1-2\eta} \rfloor\right] \leq 2^{mH(\frac{1+\lfloor \frac{\tau-\eta m}{1-2\eta} \rfloor}{m}) + 2 - \min(k_{\boldsymbol{X}}, m)}.$$

Thus,

$$\begin{aligned}p_r &\leq e^{-\frac{(\mu-\tau)^2}{2\mu}} \cdot (1 - 2^{mH(\frac{1+\lfloor \frac{\tau-\eta m}{1-2\eta} \rfloor}{m}) + 2 - \min(k_{\boldsymbol{X}}, m)}) + 2^{mH(\frac{1+\lfloor \frac{\tau-\eta m}{1-2\eta} \rfloor}{m}) + 2 - \min(k_{\boldsymbol{X}}, m)} \\ &\leq e^{-\frac{(\mu-\tau)^2}{2\mu}} + 2^{mH(\frac{1+\lfloor \frac{\tau-\eta m}{1-2\eta} \rfloor}{m}) + 2 - \min(k_{\boldsymbol{X}}, m)}.\end{aligned}$$

The exponent of the second term is negative for practical values of the parameters and a decreasing function of d. Also, similarly to [8], in order to ascertain that the first term is negligible, we define $\hat{d}$ the least integer such that $\mu(\hat{d}) > (1+c)\tau$ for some $c > 0$ and for all $d \geq \hat{d}$, $e^{-\frac{(\mu-\tau)^2}{2\mu}} \leq e^{-\frac{(c\tau)^2}{2(c+1)}}$. From (3) and (4), the overall probability of the attack is lower bounded by

$$(1 - q^{\#} \cdot (e^{-\frac{(\mu-\tau)^2}{2\mu}} + 2^{mH(\frac{1+\lfloor \frac{\tau-\eta m}{1-2\eta} \rfloor}{m}) + 2 - \min(k_{\boldsymbol{X}}, m)})) \cdot (P_{FA} + \delta) \leq p_{MIM}.$$

□

From Theorem 2, any efficient attacker achieving a noticeable advantage $\delta^{\#}$ against the $GHB^{\#}$ protocol in the MIM-model can be turned into an efficient attacker against the same protocol in the DET-model. However, from Theorem 1 this contradicts the conjectured hardness assumption of the LPN problem.

6 Conclusions

The design of lightweight protocols for RFID tag authentication is a challenging task. In this paper, we introduced a new secure authentication protocol, the $GHB^{\#}$, that it is supported by a security proof based on the conjectured hardness of the LPN problem. The new protocol belongs to the family of *HB*-like protocols that have been extensively analysed in the last few years. The $GHB^{\#}$ protocol is shown to be secure against all the attacks that have been proposed so far against LPN-based authentication protocols, including the MIM attacks in which the attacker is able to modify all messages exchanged between an honest tag and the reader. These MIM attacks has been the Achilles heel of almost all the HB-like protocols with only two very recent exceptions ([15], [4]).

As further research, it is interesting to investigate the relation between the $GHB^{\#}$ protocol and other recently proposed LPN based protocols ([15], [4]), that can resist MIM attacks.

Acknowledgement. The authors would like to thank the anonymous reviewers for their valuable comments and suggestions.

References

1. Avoine, G.: RFID Security and Privacy Lounge, The list of papers is available at `http://www.avoine.net/rfid/download/bib/bibliography-rfid.pdf`
2. Bringer, J., Chabanne, H., Dottax, E.: HB^{++}: a Lightweight Authentication Protocol Secure against Some Attacks. In: Proceedings of the IEEE Int. Conference on Pervasive Sevices, Workshop - SecPerU (2006)
3. Bringer, J., Chabanne, H.: *Trusted-HB*: A Low-Cost Version of HB Secure Against Man-in-the-Middle AttackHB^{++}. IEEE Transactions on Information Theory 54, 4339–4342 (2008)
4. Bosley, C., Haralambiev, K., Nicolosi, A.: HB^{N}: An HB-like protocol secure against man-in-the-middle attacks. Cryptology ePrint Archive, Report 2011/350 (2011), `http://eprint.iacr.org`
5. Carlet, C.: Vectorial Boolean Functions for Cryptography. In: Crama, Y., Hammer, P.L. (eds.) Boolean Models and Methods in Mathematics, Computer Science, and Engineering, pp. 398–469. Cambridge University Press (2010)
6. Duc, D.N., Kim, K.: Securing HB^{+} against GRS Man-in-the-Middle Attack. In: Proceedings of the Symp. on Cryptography and Information Security (2007)
7. Gilbert, H., Robshaw, M., Silbert, H.: An Active Attack against HB^{+}-a Provable Secure Lightweighted Authentication Protocol. Cryptology ePrint Archive, Report 2005/237 (2005), `http://eprint.iacr.org`

8. Gilbert, H., Robshaw, M., Seurin, Y.: *HB*$^{\#}$: Increasing the Security and Efficiency of *HB*$^{+}$. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 361–378. Springer, Heidelberg (2008)
9. Gilbert, H., Robshaw, M., Seurin, Y.: Good Variants of HB^{+} Are Hard to Find. In: Tsudik, G. (ed.) FC 2008. LNCS, vol. 5143, pp. 156–170. Springer, Heidelberg (2008)
10. Gold, R.: Maximal recursive sequences with 3-valued recursive crosscorrelation functions. IEEE Transactions on Information Theory 14, 154–156 (1968)
11. Hopper, N.J., Blum, M.: Secure Human Identification Protocols. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 52–66. Springer, Heidelberg (2001)
12. Juels, A., Weis, S.A.: Authenticating Pervasive Devices with Human Protocols. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 293–308. Springer, Heidelberg (2005)
13. Katz, J., Shin, J.S.: Parallel and Concurrent Security of the HB and HB^{+} Protocols. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 73–87. Springer, Heidelberg (2006)
14. Katz, J., Shin, J.: Analyzing the *HB* and *HB*$^{+}$ Protocols in the Large Error Case. Cryptology ePrint Archive, Report 2006/326 (2006), `http://eprint.iacr.org/`
15. Kiltz, E., Pietrzak, K., Cash, D., Jain, A., Venturi, D.: Efficient Authentication from Hard Learning Problems. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 7–26. Springer, Heidelberg (2011)
16. Leng, X., Mayes, K., Markantonakis, K.: *HP-MP*$^{+}$: An Improvement on the *HB-MP* Protocol. In: Proceedings of the IEEE Int. Conference on RFID 2008, pp. 118–124. IEEE Press (2008)
17. Lidl, R., Niederreiter, H.: Introduction to Finite Fields and Their Applications. Cambridge University Press (1994)
18. Madhavan, M., Thangaraj, A., Sankarasubramaniam, Y., Viswanathan, K.: *NLHB*: A Non-Linear Hopper Blum Protocol. In: Proceedings of IEEE National Conference on Communications, NCC (2010), CoRR abs/1001.2140:2010.
19. Massey, J.L., Omura, J.K.: Computational Method and Apparatus for Finite Field Arithmetic. US Patent No. 4,587,627 (1986)
20. Munilla, J., Peinado, A.: *HP-MP*: A Further Step in the *HB*-family of Lightweight authentication protocols. Computer Networks 51, 2262–2267 (2007)
21. Ouafi, K., Overbeck, R., Vaudenay, S.: On the Security of HB$^{\#}$ against a Man-in-the-Middle Attack. In: Pieprzyk, J. (ed.) ASIACRYPT 2008. LNCS, vol. 5350, pp. 108–124. Springer, Heidelberg (2008)
22. Piramuthu, S.: *HB* and Related Lightweight Authentication Protocols for Secure RFID Tag/Reader Authentication. In: Proceedings of CollECTeR Europe Conference, Basel, Switzerland, June 9-10 (2006)
23. Reza, M., Abyaneh, S., On, S.: the Security of Non-Linear *HB* (*NLHB*) Protocol Against Passive Attack. Cryptology ePrint Archive, Report 2010/402 (2010), `http://eprint.iacr.org/`
24. Rizomiliotis, P.: *HB-MAC*: Improving the Random - *HB*$^{\#}$ Authentication Protocol. In: Fischer-Hübner, S., Lambrinoudakis, C., Pernul, G. (eds.) TrustBus 2009. LNCS, vol. 5695, pp. 159–168. Springer, Heidelberg (2009)
25. Yoon, B., Sung, M.Y., Yeon, S.H., Oh, S., Kwon, Y.: Kim, Ch., Kim, K.-H.: *HB-MP*$^{++}$ protocol: An ultra light-weight authentication protocol for RFID system. In: Proceedings of the IEEE Int. Conference on RFID, pp. 186–191 (2009)