Wireless Personal Communications (2021) 117:2225–2251
https://doi.org/10.1007/s11277-020-07969-w

An Ultra-Lightweight Mutual Authentication Protocol Based on LPN Problem with Distance Fraud Resistant

Kazem Mirzadi[1] · **Jamshid B. Mohasefi**[1]

Accepted: 11 November 2020 / Published online: 2 January 2021

Abstract
RFID tags are one of the main enablers of the internet of things. All objects have to be equipped with an electronic product code such as RFID tags. Because of minimizing the price, RFID environments are resource-scarce, then designing ultra-lightweight authentication protocols is of great importance. Many ultra-lightweight authentication protocols such as HB family protocols are proposed. One of the most important threats against HB family protocols is a type of man-in-the-middle attack called GRS. Also, in the real world, IoT requires mutual authentication that traditional HB protocols do not support it. Besides, misconceptions about reader-tag distance could create problems in several applications especially in contactless systems such as access control and electronic payment systems, which could be damaged by distance-based frauds. In the present work, we have proposed a novel distance bounding protocol based on HB family protocols with four major characteristics: (1) it can resist terrorist, mafia, and distance fraud attacks; (2) it is a lightweight mutual authentication protocol capable of being used in low-cost IoT equipment such as NFC and RFID; (3) it employs a hard problem that be post-quantum resistant, and (4) it identifies and solves the weaknesses of HB protocols including GRS attacks. The proposed protocol has also been shown to be able to address the known weaknesses and attacks in distance frauds and HB family protocols.

Keywords Lightweight protocol · Mutual authentication · Key disclosure · Distance fraud · LPN · Distance bounding · RFID

1 Introduction

With the increase in the use of IoT in the past decade, the security issue has attracted the attention of research and industrial centers. Due to the more closeness of IoT to real life, its security is the major challenge. The most critical security issues are privacy, data integrity,

✉ Kazem Mirzadi
kazem_mirzadi@yahoo.com

Jamshid B. Mohasefi
j.bagherzadeh@urmia.ac.ir

[1] Department of Computer Engineering, Urmia University, Urmia, Iran

and authentication. Although it is now possible to implement the concept of IoT, people will not use it until they are sure it is safe.

Despite rapid increase in the number of articles on the security of IoT, there still exist two major challenges for the comprehensive development of security in IoT. First, IoT is inherently complex due to information exchange among several heterogeneous entities in different locations and at different times. This increases the difficulty of the development and implementation of scalable, consistent, and efficient security mechanisms. Second, there are significant resource constraints. Despite the introduction of many security protocols and solutions, most IoT objects, especially passive devices like NFC and passive RFID tags, suffer from limited computational resources; therefore, complex security algorithms and heavy-weight protocols cannot be implemented on these objects. Lee et al. [1] suggested as low as 250–3000 logical gates for security tasks in RFID tags, but these numbers are not sufficient for the implementation of classic encryption methods. For example, at least 5000 logical gates are required for a one-way hash function. Therefore, new approaches have to be introduced to address these requirements.

It should be noted that with the development of quantum computers, number theory based systems will be easily broken. Factorization and discrete logarithm are the two most known and used hard problems. Unfortunately, they can be easily solved on a quantum computer by Shor's algorithm, which was introduced in 1994 [2]. Also, the research area of cryptography demands for crypto-diversity which says that we should offer a range of hard problems for public-key cryptography. If one hard problem proves to be easy, we should be able to provide alternative solutions. Some of the candidates for post-quantum hard problems, i.e. problems which are believed to be hard even on a quantum computer, are the Learning Parity with Noise (LPN), the Learning with Errors (LWE) and the Shortest Vector Problem (SVP) [3]. The hardness of LPN and LWE has proved to be a goldmine in modern cryptography. Both LPN and LWE have given us efficient and plausibly quantum-proof cryptographic constructions [4].

LPN is a hard problem that is attractive, as it is believed to be post-quantum resistant and suitable for lightweight devices. In practice, it has been employed in several encryption schemes and authentication protocols with different levels of practicality and security [3].

Hopper and Blum [5] introduced the first LPN-based authentication protocol known as HB protocol. Most HB family protocols were vulnerable to man-in-the-middle (MIM) attack. Furthermore, in most of them authentication was generally performed unilaterally while mutual authentication is necessary for IoT and consequently RFID systems.

Wrong assumptions of reader-tag distance in RFID systems create serious problems in a variety of applications. A tag could only be activated and recognized when it is close to a reader [6]. One example is access control systems installed at building entrances. These systems should ensure that the person who wants to enter is at a certain distance from the door. Once the user gets close to the building, his contactless smart card (tag) and the reader installed near the door, start a challenge and response protocol. If the protocol is successfully completed, the door is opened.

1.1 Our Contribution

Currently, variants of HB and distance bounding protocols are two popular research topics in wireless network security and RFID systems. The aim of this work is to introduce a lightweight authentication protocol for internet of things, especially RFID systems to overcome the following major challenges:

1. Strict resource constraints, mainly for reducing cost, make the use of asymmetric and symmetric encryption as well as Hash algorithms unreasonable. In addition, they can be easily solved on a quantum computer by Shor's algorithm. therefore, it seems necessary to develop new authentication protocols employing a hard problem that be post-quantum resistant and suitable for lightweight devices. For this reason, we proposed a protocol based on HB family protocol and LPN problem.
2. The developed authentication protocol has to be immune to passive (such as eavesdropping) and active (such as MIM) attacks. GRS attack, a simple MIM attack, is a major threat against HB protocols, which was shown by Gilbert et al. [7]. Hence, the second aim of this work was to address these attacks without employing classical encryption protocols.
3. It is noteworthy that unilateral authentication is performed in most HB family protocols. Tag-reader communication is always considered to be secure and there is only the possibility of RFID tag impersonation. However, IoT requires mutual authentication. Using two independent authentication protocols results in higher risks of relay attack, replay attack, non-synchronization attack, session hijacking, etc. Therefore, our third aim was to develop a mutual authentication protocol capable of resisting these attacks.
4. In the meanwhile, incorrect assumptions on tag-reader distance creates problems in a variety of applications [6]. Major threats of distance bounding protocols are terrorist fraud, mafia fraud, and distance fraud attacks [8]. Identifying these attacks and assuring that the other communicating party is within the authorized distance, is necessary. Distance Bounding Protocols (DBs) cover a wide range of research in RFID systems that aim to prevent relay attacks. Attackers in these attacks impersonate a tag without information exchange or modification [9]. Therefore, our fourth aim in this work was to provide an authentication protocol capable of resisting against distance-related attacks.

To achieve these aims, in this work, we have proposed a lightweight mutual authentication protocol based on HB family protocols which can be used in low-cost IoT equipment like NFC and RFID. It is shown that the proposed protocol is capable of resisting against terrorist fraud, mafia fraud, and distance fraud attacks and simultaneously addresses the main weaknesses of HB protocols including key recovery, GRS, etc.

2 Review of Previous Works

Hopper and Blum developed the first practical and secure human identification protocol, HB protocol, which could operate based on limited memorizing and calculating abilities of human [5]. Specifically, this protocol was developed based on the difficulty of solving LPN problems. One matrix multiplication and some XORs were the only operations performed in this protocol. Since HB protocols are light-weight authentication protocols, they can be implemented in RFID devices (which have limited computational power) [10]. Many modifications were performed on HB protocols and some of them are introduced in this section.

Later et al. [11] found that HB protocol could only resist passive attacks and therefore developed a modified version called HB + which was able to resist active attacks too. However, Gilbert et al. [7] showed that both the original and modified protocols were vulnerable to MIM attacks. Later, Bringer et al. [12] modified HB + protocol and developed a new version of it called HB ++ to make it immune to active attacks. However, this protocol was also vulnerable to some active attacks which are described in Gilbert et al. [7]. To

solve the disadvantages of HB++, Lee et al. [1] proposed the AUHB++protocol in 2008. The new protocol could resist MIM attacks but had high computational costs and lacked privacy protection.

PUFHB is another member of HB family protocols which was developed by Hammouri et al. [13] in 2008 and was a tamper-resistant protocol. However, it was vulnerable to distance fraud, GRS, and MIM attacks, additional hardware was required in RFID tags.

Munilla and Peinado developed a new protocol called HB-MP in 2007 [14]. Later, Leng et al. [15] showed that HB-MP protocol was seriously threatened by a type of MIM attack and introduced a modified version called HB-MP+which was resistant to this attack. As far as we know, HB-MP+could resist MIM attacks but had other security problems. First, authentication was performed in *m* time-consuming rounds. Second, applying a one-way hash function required high computational costs in tags and created practical problems. Although only 250–3000 logical gates are suggested for assigning security tasks in a tag [9], one-way functions require at least 5000 gates.

Gilbert et al. [16] introduced two novel protocols in 2008 called HB# and RANDOM-HB#. RANDOM-HB# needed high memory on tags because it needed to store two random matrices. On the other hand, HB# improved the efficiency of RANDOM-HB# protocol using Toeplitz matrix structure and the authentication process was performed in a single round. Ouaf et al. [17] introduced a general MIM attack against these two protocols which obtained secret key values in 220 to 225 and 228 to 234 authentication steps in HB# and RANDOM-HB#, respectively. HB# protocol was vulnerable to MIM attacks although authentication process was performed in a single round and it did not use hash functions.

Later, Yoon et al. [18] developed HB-MP++protocol by employing linear feedback shift register which had good randomized outputs; however, it had very high computational costs. Madhavan et al. [19] introduced NL-HB protocol using simple non-linear functions. Although they achieved the same security of HB protocol with smaller key size, the modified protocol did not rely on LPN problem and therefore was not suitable for post-quantum cryptography.

Samia et al. [20] used rotation and complementation to develop RC-HB lightweight protocol which was also a more secure version of HB protocol with lower communication costs and shorter keys. Khoureich [21] improved the session key exchange of hHB protocol and called it LhHB. This was a two-step protocol and was more applicable compared to the original version. However, all of these protocols were vulnerable to distance fraud, GRS, and MIM attacks and did not have mutual authentication [22].

Kiltz et al. [23] introduced novel authentication protocols and even MACs from LPN. In fact, they developed a two-round authentication protocol which was secure against active adversaries. They constructed two efficient MACs and made two-round authentication protocols secure against MIM attacks, using the capabilities of the LPN assumption. The efficiency of the proposed authentication protocol was almost similar to the HB+protocol but with a key length of almost two times longer.

To address challenges regarding distance, Brands and Chaum introduced distance bounding (DB) protocols in 1993 [24]. In these protocols, delays of challenge-response round-trip are measured and the closeness of provers to the verifiers is checked. In the first step, a fast data transfer from the reader to the tag takes place. Then, the tag transmits a response data to the reader and the timer is stopped. The propagation time is calculated in the reader based on this round trip time. After repeating the procedure for n times (n is a security parameter), the reader decides if the tag is located in the trusted zone. In this protocol, public key cryptography is used in the signature stage, and therefore, it is not suitable for devices with weak hardware.

A brief literature review revealed that more than 50 distance bounding protocols have been reported since 1993, but as can be found in [25], among them only a few are still considered as efficient. Ahmadi and Safavi Naini in [26] proposed a privacy-preserving distance-bounding (PDB) protocol which was resistance to terrorist attacks. PDB protocol keeps the prover anonymous by solving the disadvantages of DBPK-log protocol [27]. However, this protocol has difficulty in revoking the validity of certain particular provers. Especially, it cannot be revealed that if a hijacked secret key is used by a session or not because authentication has to be anonymous. Avoine et al. [8] proposed a new method (Tread) to achieve protocols provably resistant to terrorist fraud without long-term secret keys for provers. Instead, attackers can simply replay the information they receive from their accomplices. Therefore, they provide a generic construction to provably secure distance-bounding protocol. This way, Tread protocol can eliminate classical threats, but this is done by the use of a digital signature method and an encryption algorithm.

Munilla et al. [28] improved the protocol by adding void challenges to decrease the success likelihood of attackers. This protocol had some disadvantages that encouraged Kim and Avoine [29] to introduce the concept of mixed challenges in which reader-tag challenges can be divided into two groups of predefined and random challenges. In the former, predefined bits are already known to both reader and tag and in the latter random bits from the reader are used.

Pagnin et al. [30] combined distance bounding ideas and HB + protocol and developed a hybrid HB + DB protocol for lightweight device authentication. Their work showed that the application of the LPN problem as the basis of response function could be effective. The main objective of HB + DB protocol is to provide a modified version of HB + protocol which could resist GRS MIM key-recovery attacks. Instead of preventing attacks with cryptographic response functions, they proposed detecting MIM attackers by accurate timing of prover-verifier challenge-response exchanges like that in the distance-bounding protocol. This places no additional burden on the constrained prover as only the verifier is tasked with taking the round-trip measurement. GRS attacks against HB + DB results in authentication failure since the verifier detects the delay incurred in sampling and flipping bits. Therefore, the authors successfully prevented the attackers from getting information on the prover's key which eliminates the risk of the GRS attack.

There are different strategies in DB protocols. Some protocols use signature(sign) of the transcript. We note that the presence of the (signed) transcript in the verification phase prevents Mafia Fraud attacks. Indeed, it prevents the adversary from using a pre ask strategy to improve his success probability since the verifier aborts the protocol if the challenges do not correspond to the adversary's challenges. But sign use the classical encryption algorithms and thus the computational cost increased. Some other protocols do not use signature of the transcript. In this strategy, the adversary can pre-ask the prover with a random challenge in order to obtain some responses beforehand. Protocols using PRF to pre-compute response for the fast phase are often exposed to a Distance Fraud attack, and protocols using PRF to compute the signature of the transcript are often exposed to a MiM attack which permits to the adversary to impersonate the prover [25].

It is noteworthy that in all methods, classical encryption algorithms (asymmetric, symmetric, hashing or Pseudo-Random Function) are employed, and therefore they cannot be used in passive IoT devices.

Table 1 gives an overview of some HB family protocols that mentioned in this section. It should be noted that the comparison of protocols from a security point of view is given in Sect. 5. The success probabilities of mafia fraud attack and terrorist fraud attack in some of important DB protocols are compared in Table 2. In the present work, we have

Table 1 Overview of HB family protocols

Protocol	Based on	Comments
HB (2001)	LPN	The first protocol tries to provide light weight characteristics
HB + (2005)	HB	Vulnerable from resistance disclosure attack and not 100% secure from passive attack
HB ++ (2006)	HB+	Computation cost is high and prone to side channel attack
HB-MP + (2008)	HB +, Hash	Reduce the communication cost, not provide mutual authentication and computational cost is high
AUHB ++ (2008)	HB ++	Computation cost is high
PUF-HB (2008)	HB +	RFID tags needed more hardware requirements
HB# (2008)	HB+	Use Toeplitz matrices to reduce the communication complexity
HB-MP ++ (2009)	HB-MP+	High computational cost
NL-HB (2010)	LPN	Achieved the same security of HB protocol with smaller key size
RC-HB (2011)	HB	Less communication complexity
IHB (2015)	HB+	Using rotation and complementation with shorter key lengths and low communication cost

Table 2 Comparision of the mentioned distance bounding protocols

Protocol	Model of verification	Mafia fraud resistance	Terrorist fraud resistance
Brands and Chaum [24]	Sign (transcript)	$\left(\frac{1}{2}\right)^n$	1
Capkun et al. (2003)	Sign (transcript)	$\left(\frac{1}{2}\right)^n$	1
Bussard and Bagga [27]	Sign (transcript)	$\left(\frac{1}{2}\right)^n$	1
Hancke and Kuhn (2005)	PRF	$\left(\frac{3}{4}\right)^n$	1
Munilla and Peinado [28]	PRF	$\left(\frac{3}{5}\right)^n$	1
Kim et al. [29]	PRF	$\left(\frac{1}{2}\right)^n$	1
Kim and Avoine (2011)	PRF	$\left(\frac{1}{2}\right)^n$	1
Proposed protocol	PRF	Equation 34	Equation 35

been combined HB family protocols whit distance bounding protocols to overcome some requirements of authentication protocols in the IoT environment. Therefore, the new lightweight mutual authentication protocol capable of being used in low-cost IoT equipment such as NFC and RFID and be post-quantum resistant. Besides, it can resist terrorist, mafia, and distance fraud attacks. The proposed protocol has also been shown to be able to address the known weaknesses and attacks in distance frauds and HB family protocols.

3 The Proposed Protocol

Based on the goals mentioned in our contribution section, here, we have proposed a novel lightweight mutual authentication protocol based on HB protocols. In the rest of this section, the threat model, the notations and parameters used in the proposed protocol, and the detail of the proposed protocol are discussed.

3.1 Threat Model

To authenticate an RFID tag, the tag is queried by the closest RFID reader and the tag's data is obtained through an RF channel. Then, this data is relayed by the RFID reader to the back-end server which verifies the authority of the tag.

Some assumptions were made in the execution of the proposed protocol: (1) authentication session is initiated by the reader (2) back-end server-reader communication channel is secure, (3) reader-tag communication channel is not secure, (4) the data in back-end server were accessible through a secure access control method, (5) authentication parameters stored in back-end server and tags could be updated, and (6) due to resource scarcity in RFID tags, encryption algorithms such as asymmetric, symmetric, and hash algorithms are not applicable.

Some important attacks which were dealt with in the proposed protocol are introduced below.

3.1.1 Distance Bounding Attacks

Location based attacks are among the most important attacks in physical layer. In RFID systems, wrong assumptions regarding reader-tag distance create problems in many applications [21]. Cryptographic protocols which act in application layers are vulnerable to location based attacks. An efficient way to avoid these types of attacks is the application of DB protocols. Generally, three types of location-based attacks, which can be prevented by DB protocols are distance fraud, mafia fraud and terrorist fraud.

3.1.2 Distance Fraud Attack

These attacks are performed by swindling genuine cards instead of external adversaries which operates from out of the range where it supposed to be. This attack, could initially seem not very dangerous, but they should be practically taken into account in systems with variable access rights based on physical locations. An example of this attack is shown in Fig. 1.

3.1.3 Mafia Fraud or Relay Attack

Mafia fraud attack, also known as relay attack, is a kind of MIM attack in which the attacker interacts with a genuine reader and relays its messages to the rough reader which is near the genuine card. The attacker deceives the reader into thinking that the message is directly coming from the genuine card. In this case, the genuine reader and card could be far from each other. In a common type of relay attack, the attacker communicates with both parties and then relays messages between them without manipulating or even reading them. An example of this attack is shown in Fig. 2.

3.1.4 Terrorist Fraud Attack

Terrorist fraud is another type of attack against distance bounding protocols which was first introduced by Desmedt [31]. In terrorist fraud attack, a genuine card collaborates with an adversary giving him the necessary information to impersonate it only a few times. That is to say, adversary is given the necessary information to pass a single round of the protocol, not whenever he wishes. If the adversary could access the long term private key of the

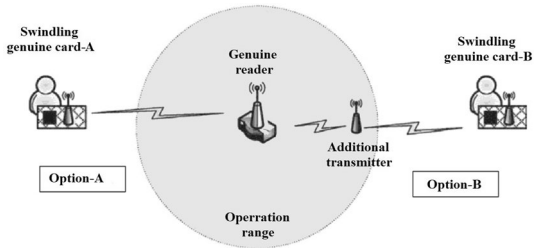

Fig. 1 Sketch of distance fraud attacks [21]

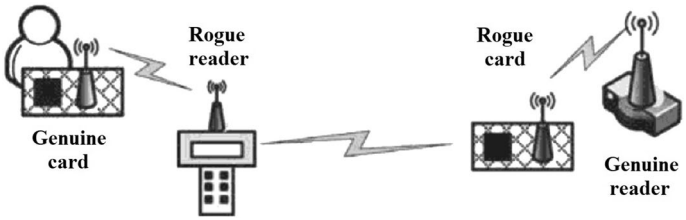

Fig. 2 Sketch of mafia fraud attack [21]

card, it could impersonate the card every time he wishes. From a cryptographic point of view, the adversary and the legitimate card would be the same party in this case.

3.1.5 GRS Attack

In some MIM attacks, trivial variations in challenge sequences can result in key disclosure. Gilbert et al. [7] presented one major disadvantage of HB family protocols, which is their vulnerability to GRS attacks. In these attacks, as can be seen in Fig. 3 against HB+, an adversary (A) modifies the challenges sent to the prover (P) by the verifier (V) and evaluates if this modification causes successful authentication. To be more precise, A selects a constant k bit vector $\mathbf{d}$ and employs it to manipulate the sent challenges $\mathbf{a}$; in other words, $\mathbf{d}$ is XORed with $\mathbf{a}$ in each authentication round. It should be noted that the same $\mathbf{d}$ is applied in all HB+ protocol execution rounds. The success probability of the attack is high when $\mathbf{d} \cdot \mathbf{x} = 0$ and it is more likely to fail when $\mathbf{d} \cdot \mathbf{x} = 1$. Authentication acceptance gives one bit of the secret key $\mathbf{x}$. To obtain the whole secret key $\mathbf{x}$, A should repeat the protocol k times for linearly independent values of $\mathbf{d}$, and solve the resulting system every time. Once $\mathbf{x}$ is obtained, A can simply impersonate P by adjusting $\mathbf{b} = 0$ or employing a strategy similar to the one described for recovering $\mathbf{y}$.

3.2 The Parameters and Notations Used in the Proposed Protocol

The parameters and notations used in the developed protocol are summarized in Table 3. In this article, matrices, vectors, and scaler variables are written in capital bold, small bold, and small italic, respectively. Some matrices are in the form of

Verifier $\mathcal{V}$		Adversary $\mathcal{A}$		Prover $\mathcal{P}$
$(\mathbf{x}, \mathbf{y})$				$(\mathbf{x}, \mathbf{y}, \eta)$
$\mathbf{a} \xleftarrow{R} \{0,1\}^k$	$\xleftarrow{\mathbf{b}}$		$\xleftarrow{\mathbf{b}}$	$\mathbf{b} \xleftarrow{R} \{0,1\}^k$
	$\xrightarrow{\mathbf{a}}$	$\mathbf{d} \xleftarrow{R} \{0,1\}^k$		$\epsilon \in \{0,1 \mid \mathbb{P}[\epsilon = 1] = \eta\}$
		$\mathbf{a}' = \mathbf{a} \oplus \mathbf{d}$	$\xrightarrow{\mathbf{a}'}$	
	$\xleftarrow{z'}$		$\xleftarrow{z'}$	$z' = (\mathbf{a}' \cdot \mathbf{x}) \oplus (\mathbf{b} \cdot \mathbf{y}) \oplus \epsilon$
Accept if it holds $(\mathbf{a} \cdot \mathbf{x}) \oplus (\mathbf{b} \cdot \mathbf{y}) = z'$				

Fig. 3 GRS attack during one round of HB+ protocol [7]

Table 3 The notations and parameters used in the proposed protocol

v1, v2	k bit noise vectors (every bit of these vectors is equal to 1 with probability $\eta \in [0, ½]$)
row[i](**X**)	The ith row of the matrix **X**
col[i](**X**)	The ith column of the matrix **X**
z[i]	The ith index of the vector **z**
Hwt(**x**)	the Hamming weight of the vector **x**
Δt_{max}	The highest acceptable latency for sending a challenge bit and receiving the response
Δt_i	Latency for sending a challenge bit and receiving the response in ith round of the protocol
Operator $\odot$	Vector–Matrix inner product of a vector into a matrix (defined in the text)
Operator ','	Concatenation of vectors or part of vectors, to create a big vector
Operator '.'	Binary inner product of two vectors
Operator $\oplus$	XOR operator (bitwise XOR of two binary vectors results in a binary vector)
Operator $\oplus$	TXOR operator (defined in the text)
Operator '*'	Specific form of multiplying two matric (defined in the text).

Toeplitz matrices. A $k \times m$ binary matrix **M** is considered as a Toeplitz matrix when $\forall i \in [1..k-1], j \in [1.m-1] \Rightarrow \mathbf{M}[i][j] = \mathbf{M}[i+1][j+1]$. Therefore, the matrix can be specified by its first column and first row stored in $k+m-1$ bits vector, $\mathbf{w} = (\mathbf{M}[1][1], \ldots, \mathbf{M}[1][k], \mathbf{M}[2][1], \mathbf{M}[3][1], \ldots, \mathbf{M}[m][1])$, rather than $k \times m$ bits required for a truly random matrix.

Binary inner product of two k-bit vectors **a** and **b** is denoted as $\mathbf{a} \cdot \mathbf{b}$, and is expressed as:

$$\forall \boldsymbol{i} \in [1 \ldots k].c = \left(\sum_{i=1}^{k} \mathbf{a}[i] \times \mathbf{b}[i] \right) \bmod 2.$$

Binary inner product of a $k \times m$ matrix **X** with a k-bit vector **a** is denoted as $\mathbf{a} \odot \mathbf{X}$, and is defined as:

$$\forall \boldsymbol{i} \in [1 \ldots k].\mathbf{b}[i] = acol[i](\mathbf{X}),$$

where '.' is the inner product of two vectors.

If **b** is a k-bit vector and we have a $k \times m$ bit Toeplitz matrix **M** represented by vector $\mathbf{w} = (\mathbf{M}[1][1], \ldots, \mathbf{M}[6][k], \mathbf{M}[2][1], \ldots, \mathbf{M}[m][1])$, then TXOR of **M** and **b**, can be written as: $\mathbf{M} \oplus \mathbf{b} = ((\mathbf{w}[6], \ldots, \mathbf{w}[k]) \oplus \mathbf{b}), ((\mathbf{w}[k], \ldots, \mathbf{w}[k+m-1]) \oplus \mathbf{b})$ where ',' is concatenation operator.

The operator '*' was defined as a specific form of multiplying two matrices. If **X** is $m \times h$ matrix and **B** is a $k \times m$ matrix then $\mathbf{z} = \mathbf{B} * \mathbf{X}$ is stated as:

$\forall \boldsymbol{i}.\mathbf{z}[i] = \text{row}[i](\mathbf{B}) \cdot \text{col}[i](\mathbf{X})$, where '.' is the inner product of two vectors.

3.3 Proposed Protocol

The following initial information is included in authentication entities: (1) there exists a $k \times m$ bit Toeplitz matrix **X** as secret key in each tag, (2) for each tag, two copies of secret matrix $\mathbf{X}$, $\mathbf{X}_{\mathbf{recover}}$ and $\mathbf{X}_{\mathbf{current}}$, are kept in back-end server, (3) the longest acceptable latency threshold for sending challenge bits and receiving responses is set to be Δt_{max}.

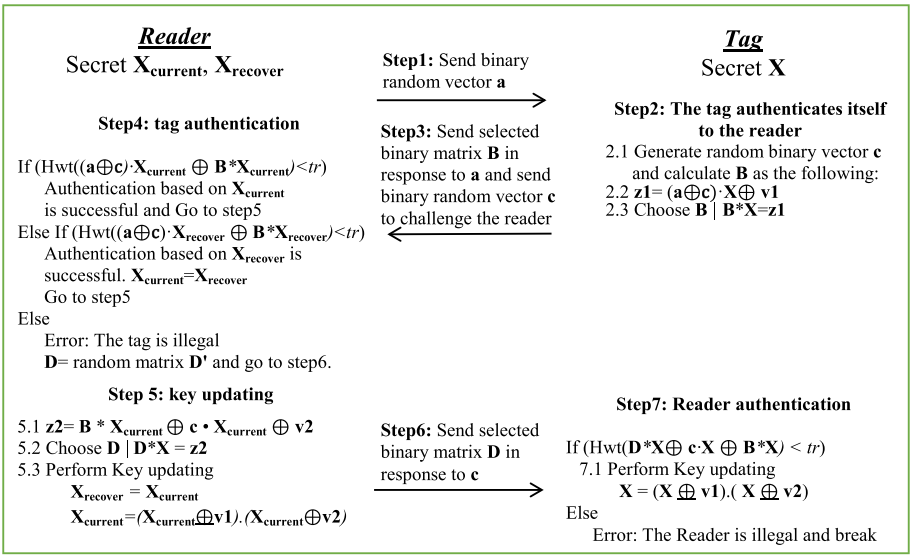

Fig. 4 Authentication steps

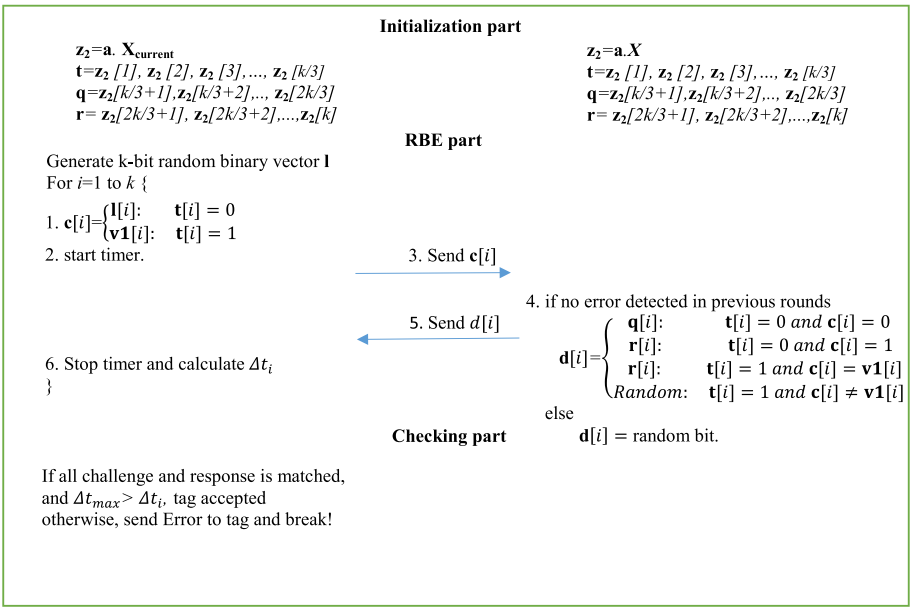

Fig. 5 Distance bounding steps

Every data in the server was assumed to be accessible through secret channels for all readers.

Since GRS attacks may initiate within the trusted zone and may not significantly increase Δt_i, using DB protocol without digital signature may not avoid GRS. Furthermore,

due to limited resources of RFID tags, using digital signature may not be realistic. We claim and prove that combining DB mechanism with mutual authentication will solve these weaknesses. Thus, the proposed protocol is described in two phases: mutual authentication and distance bounding shown in Figs. 4 and 5, respectively. These two phases are described in more detail in the remainder of this section. It is noteworthy that, distance bounding can be ignored from the main protocol in some applications, because it is an extension.

3.3.1 Mutual Authentication

In most HB protocols, it is assumed that tag impersonations are possible but reader impersonations are not. However, this assumption is wrong and mutual authentication is essential for IoT. Also, using two independent authentications can increase the risk of desynchronization, relay, and replay attacks as well as session hijacking. Therefore, we used mutual authentication method.

In HB family protocols both readers and tags should have shared keys for secure communication. The main threat against HB family protocols is the GRS attack as discussed before. The main idea to deal with the GRS attack is to generate a response packet in each round using a new key, until the relation between challenges, keys and responses could not be detected. This can be done in two ways:

1. The primary shared key remains unchanged throughout the execution of the protocol and a new key $\mathbf{x}_s$ is generated at the beginning of each round based on the shared key. In this way, only one key is stored in tags, but the authentication time increases because of the new key generation.
2. Round keys required in the next authentication session are synchronously updated or generated at both sides. Since at the beginning of the protocol all keys are known, protocol efficiency is increased and authentication is performed in one round, but memory space is increased for storing all keys. This method may provide desynchronization problem.

The proposed protocol employs the second method and exchanges secret key **x** from a k-bits binary vector to a $k \times m$-bits binary matrix **X** where each column denotes a vector **x** in HB family protocol (m columns). Thus, a challenge vector is sent and authentication is executed in one round instead of n rounds with different random challenges. After successful authentication, matrix **X** is updated in both reader and tag according to random vectors **v1** and **v2**, the previous value of **X,** and responses sent in previous steps of authentication. A Toeplitz matrix was used to solve memory usage problem for storing matrix **X**; resolving the desynchronization problem is described at the rest of this section.

Execution steps of mutual authentication phase are described below:

Step 1 A random k-bit vector **a** is generated by the reader and sent to the tag as a challenge.

Step 2 The tag authenticates itself to the reader. For this, first the tag generates a random binary vector **c** which is used as a challenge vector by reader (step4). Then, it calculates vector **z1**. Each bit of **z1** calculated based on the random noise vector **v1**, generated challenge random vector **c**, received challenge vector **a**, and one column of matrix **X**. (similar to computing z in HB +).

For making the problem even more complex, matrix **B** is generated such that $\mathbf{B}^*\mathbf{X} = \mathbf{z1}$ and is sent to V instead of vector **z1**. That is to say, row[i](**B**) is sent instead of the bit **z1**[i].

Vector row $[i](\mathbf{B})$ is calculated such that $\mathbf{z}1[i] = \text{row}[i](\mathbf{B}).\text{Col}[i](\mathbf{X})$. Each row in matrix **B** is obtained using the algorithm reported in [10]. Munilla and Peinado [9] showed that finding **X** given **B** is at least as difficult as solving LPN problem.

Step 3 in response to the challenge vector **a,** binary matrix **B** is generated and sent; a new challenge vector **c** is also sent to reader.

Step 4 The reader authenticates the tag by verifying its response (**B**). As mentioned, two copies of secret matrix **X** ($\mathbf{X}_{\mathbf{recover}}$ and $\mathbf{X}_{\mathbf{current}}$) are kept in the reader about each tag. To avoid message loss or desynchronization attack in step 6, $\mathbf{X}_{\mathbf{current}}$ contains the new updated key and $\mathbf{X}_{\mathbf{recover}}$ contains the last synced and valid key. After the completion of previous mutual authentication, the keys on both sides are synchronously changed. Therefore, $\mathbf{X}_{\mathbf{current}}$ on the reader side corresponds to **X** on tag and the following authentication step is performed according to $\mathbf{X}_{\mathbf{current}}$. It is noteworthy that if a desynchronization attack happens or authentication of the reader fails, the key inside the tag remains unchanged. This way, the value of **X** in tag is equal to the last valid key on the server ($\mathbf{X}_{\mathbf{recover}}$). Therefore, authentication can be done based on $\mathbf{X}_{\mathbf{recover}}$.

At this stage, authentication is performed through the comparison of vectors $(\mathbf{a} \oplus \mathbf{c}) \cdot \mathbf{X}$ and $\mathbf{B} * X_{recover}$ or $\mathbf{B} * X_{current}$, and if there are fewer errors than the threshold value of $tr = u \times \text{m}$, where $u \in \left]\eta, 1/2\right[$, the tag is considered as legitimate.

If tag authentication is not done using $\mathbf{X}_{\mathbf{current}}$ or $\mathbf{X}_{\mathbf{recover}}$, it is considered as a fraudulent tag. In this case, the matrix **D** is generated based on the random matrix **D**' and the step 6 is executed. This way, the adversary does not know whether the authentication is rejected or accepted and it avoids MIM attack.

If authentication is performed based on $\mathbf{X}_{\mathbf{recover}}$, the value of $\mathbf{X}_{\mathbf{current}}$ is updated according to the value of $\mathbf{X}_{\mathbf{recover}}$; otherwise, it is kept unchanged. Therefore, after a successful tag authentication $\mathbf{X}_{\mathbf{current}}$ is ensured to be equal to **X**.

Step 5 After successful authentication of the tag in step 4, the reader generates matrix **D** as described in step 2 for matrix **B**. In addition, the reader generates and updates matrix $\mathbf{X}_{\mathbf{current}}$ based on **v1** and **v2**. To do so, a new operator $\bigoplus$ was defined as described in Sect. 3.1 and the tag updated its key as $\mathbf{X}_{\mathbf{current}} = (\mathbf{X}_{\mathbf{current}} \bigoplus \mathbf{v1}).(\mathbf{X}_{\mathbf{current}} \bigoplus \mathbf{v2})$.

Step 6 **D** is sent by the reader as a response to received challenge **c**.

Step 7 The reader is authenticated by the tag by verifying the received matrix **D**. If authentication is successful, the key is updated as $\mathbf{X} = (\mathbf{X} \bigoplus \mathbf{v1}).(\mathbf{X} \bigoplus \mathbf{v2})$. In this case **X** is equal to $\mathbf{X}_{\mathbf{current}}$.

3.3.2 Distance Bounding Phase

Figure 5 shows distance bounding phase. In situations where distance fraud is important, distance bounding phase is executed after mutual authentication. At the beginning of this phase, instead of symmetric cryptography, a lightweight function is employed. This is performed using the key **X**. First, shared vectors, **z2**, are generated in tag and reader as $\mathbf{z2} = \mathbf{a}.\mathbf{X}$ and $\mathbf{z2} = \mathbf{a}.\mathbf{X}_{\mathbf{current}}$, respectively.

Then, rapid bit exchange (RBE) is executed. Here, mixed challenges idea [29] is used. The challenges sent from reader in RBE are divided into two groups of random and predefined challenges. In this work, **z2** is divided into three n/3-bits vectors **t**, **q** and **r**. Furthermore, A k-bit random binary vector **l** is generated by the reader as a random challenge. Then, RBE is iterated k times.

The random challenges are identified by sequence **t** created in the previous step: if $\mathbf{t}[i]=1$, reader sends the random bit sequence $\mathbf{v1}[i]$ to tag and if $\mathbf{t}[i]=0$, it sends the random value $\mathbf{l}[i]$. In case of an adversary, all $\mathbf{c}[i]$ bits are randomly sent, such that predefined and random challenges cannot be distinguish. Sending predefined bit can also help the tag identify the random challenge of attackers during rapid exchange phase. The tag sends appropriate proper response to reader based on vectors **t**, **q**, and **r** and received bit. In case of an error at any step, the following responses are randomized. This way, both reader and tag fight adversary.

Furthermore, after sending a challenge in round i, the reader starts a timer to calculate Δt_i after receiving the response. Finally, on checking part, the matching of response and challenge bits is evaluated. If they match and $\Delta t_{max} > \Delta t_i$ at all rounds, tag is accepted and no confirmation is sent which increases efficiency.

4 Security Proof of the Proposed Protocol

In order to demonstrate the completeness of the proposed protocol using GNY logic, we first need to describe the protocol based on this logic and its goals. In the propositions of this section, T represents a tag and R represents a reader. In the authentication process of the proposed protocol, the following data is exchanged on the communication channel, in which **a** and **c** are the challenge vectors and **B** and **D** are the matrices.

Step 1. (**a**): $R \rightarrow T$
Step 3. (**B**): $T \rightarrow R$
Step 3. (**c**): $T \rightarrow R$
Step 4. (**D**): $R \rightarrow T$

In the following, we describe the objectives to demonstrate the completeness of the proposed protocol, which consists of three different aspects.

4.1 Goals List for the Proposed Protocol

4.1.1 Message Content Authentication

First, according to the operator * in the protocol, we define the operator $\otimes$ to find the value of **A** in such a way that $\mathbf{A}*\mathbf{X}=\mathbf{B}$.

Goal 1: R believes the message sent from T in step3 of the protocol is recognizable.

$$R| \equiv \phi(\mathbf{B} = \mathbf{z1} \otimes \mathbf{X}, \mathbf{c}) \tag{1}$$

Goal 2: T believes the message sent from R in the sixth run of the protocol is recognizable.

$$T| \equiv \phi(\mathbf{D} = \mathbf{z2} \otimes \mathbf{X}) \tag{2}$$

4.1.2 Message Origin Authentication

Goal 3: R believes that T conveyed the message in the third run of the protocol.

$$R|\equiv T|\sim(\mathbf{B},\mathbf{c}) \tag{3}$$

Goal 4: T believes that R conveyed the message in the sixth run of the protocol.

$$T|\equiv R|\sim(\mathbf{D}) \tag{4}$$

4.1.3 Key Material Establishment for Authentication

Goal 5: R believes that **v1** is a secret shared between T and R.

$$R|\equiv T\overset{\mathbf{v1}}{\longleftrightarrow}R \tag{5}$$

Goal 6: R believes that T possesses **v1**.

$$R|\equiv T\ \ni\ \mathbf{v1} \tag{6}$$

Goal 7: T believes that **v2** is a secret shared between T and R.

$$T|\equiv T\overset{\mathbf{V}_2}{\longleftrightarrow}R \tag{7}$$

Goal 8: T believes that R possesses **v2**.

$$T|\equiv R\ \ni\ \mathbf{v2} \tag{8}$$

4.2 Assumption List for Proposed Protocol

In our proposed protocol, some assumptions are made based on GNY logic as follows:

1. Since R keeps the matrix $\mathbf{X}$, it believes that $\mathbf{X}$ is recognizable.

$$R|\equiv\phi(\mathbf{X}) \tag{9}$$

2. In our protocol, the random vector **v1** and the random binary vector **c** are generated by T, so T possesses **v1** and **c**, and believes that **v1** and **c** are fresh.

$$T\ \ni\ \mathbf{v_1},T|\equiv\#\left(v_{\mathbf{1}}\right) \tag{10}$$

$$T\ \ni\ \mathbf{c},T|\equiv\#(\mathbf{c}) \tag{11}$$

3. The random vector **v1** generated by T is a temporal data required for authentication in the current run, so we assume that T believes that **v1** is a suitable secret between itself and R.

$$\mathbf{T}|\equiv T\overset{\mathbf{v_1}}{\leftrightarrow}R \tag{12}$$

4. R believes that T is an authority on generating a suitable key material **v1** shared between T and R as a primary sequence for authentication.

$$R|\equiv\ T|\Rightarrow R\overset{\mathbf{v}_1}{\leftrightarrow}T \tag{13}$$

5. The random noise vector **v2** and the random binary vector **a** are generated by R, so R possesses **v2** and **a**, and believes that **v2** and **a** are fresh.

$$R \ni \mathbf{v2}, R| \equiv \#(\mathbf{v2}) \tag{14}$$

$$R \ni \mathbf{a}, R| \equiv \#(\mathbf{a}) \tag{15}$$

6. The random noise vector **v2** generated by R is a temporal data required for authentication in the current run, so we assume that R believes that **v2** is a suitable secret between itself and T.

$$R| \equiv R \overset{\mathbf{v1}}{\leftrightarrow} T \tag{16}$$

7. T believes that R is an authority on generating a suitable key material **v2** shared between T and R as a primary sequence for authentication.

$$T| \equiv R| \Rightarrow T \overset{\mathbf{v1}}{\leftrightarrow} R \tag{17}$$

4.3 Authentication Proof Using GNY Logic

In this section, we adopt the GNY logic to analyze our proposed protocol in terms of defined goals and assumptions.

(Goal 1):

The third step of the protocol, If R believes that **X** is recognizable and R possesses the **X**, then R is entitled to believe that the formula ($\mathbf{B}*\mathbf{X} = (\mathbf{a} \oplus \mathbf{c}).\ \mathbf{X} \oplus \mathbf{v1}$) is also recognizable. Thus, R can recognize the message content, especially **v1**:

$$\frac{R| \equiv \phi(\mathbf{X}),\ R \ni \mathbf{X}}{R| \equiv \phi(\mathbf{B} * \mathbf{X} = (\mathbf{a} \oplus \mathbf{c}).X \oplus \mathbf{v1})} \tag{18}$$

(Goal 2):

The sixth step of the protocol, T possesses the **X**, **c** and **B**. Thus, it is entitled to believe that the formula ($\mathbf{D}*\mathbf{X} = \mathbf{B}*\mathbf{X} \oplus \mathbf{c}.\mathbf{X} \oplus \mathbf{v2}$) is recognizable. Therefore, T can recognize the message content, especially **v2**:

$$\frac{T \ni \mathbf{X},\ T \ni \mathbf{c},\ T \ni \mathbf{B}}{T| \equiv \phi(\mathbf{D} * \mathbf{X} = \mathbf{B} * \mathbf{X}\text{Å}\mathbf{c}.\mathbf{X}\text{Å}\mathbf{v2})} \tag{19}$$

(Goal 3):

If the following four conditions hold:

1. R receives the formula ($\mathbf{B} = \mathbf{z1} \otimes \mathbf{X}$) encrypted with the random vector **v1**;
2. R believes that **v1** is fresh;
3. R believes that **v1** is a suitable secret for itself and T;
4. R believes that the formula ($\mathbf{B} = \mathbf{z1} \otimes \mathbf{X}$) is recognizable;

Then R is entitled to believe that:

1. T possesses **v1** (that is, it has it).
2. T once conveyed ($\mathbf{B} = \mathbf{z1} \otimes \mathbf{X}$) encrypted with **v1**:

$$\frac{R \lhd (\mathbf{B} = \mathbf{z1} \Im \mathbf{X}),\ R|\equiv\ \#(\mathbf{v1}),\ R\left|\equiv R \overset{\mathbf{v1}}{\leftrightarrow} T,\ R\right| \equiv \phi(\mathbf{B} = \mathbf{z1} \otimes \mathbf{X})}{R|\equiv\ T \ni \mathbf{v1}, R|\equiv T| \sim (\mathbf{B} = \mathbf{z1} \otimes \mathbf{X})} \tag{20}$$

(Goal 4):

Also, if the following four conditions hold:

1. T receives the formula ($\mathbf{D} = \mathbf{z2} \otimes \mathbf{X}$) encrypted with the random vector **v2**;
2. T believes that **v2** is fresh;
3. T believes that **v2** is a suitable secret for itself and R;
4. T believes that the formula ($\mathbf{D} = \mathbf{z2} \otimes \mathbf{X}$) is recognizable;

Then T is entitled to believe that:

1. R possesses **v2** (that is, it has it).
2. R once conveyed ($\mathbf{D} = \mathbf{z2} \otimes \mathbf{X}$) encrypted with **v2**:

$$\frac{T \lhd (\mathbf{D} = \mathbf{z2} \otimes \mathbf{X}),\ T|\equiv\ \#(\mathbf{v2}),\ T\left|\equiv R \overset{\mathbf{v2}}{\leftrightarrow} T,\ T\right| \equiv \phi(\mathbf{D} = \mathbf{z2} \otimes \mathbf{X})}{T|\equiv\ T \ni \mathbf{v2}, T|\equiv R| \sim (\mathbf{D} = \mathbf{z2} \otimes \mathbf{X})} \tag{21}$$

(Goal 5):

If R believes that T is an authority on the statement $T \overset{\mathbf{v1}}{\longleftrightarrow} R$ and R believes that T believe in $T \overset{\mathbf{v1}}{\longleftrightarrow} R$, then R ought to believe in $R \overset{\mathbf{v1}}{\longleftrightarrow} T$ as well. So, R believes that **v1** is a suitable secret between itself and T:

$$\frac{R| \equiv T| \Rightarrow T \overset{\mathbf{v1}}{\longleftrightarrow} R,\ R|\equiv T| \equiv T \overset{\mathbf{v1}}{\longleftrightarrow} R}{R| \equiv R \overset{\mathbf{v1}}{\longleftrightarrow} T} \tag{22}$$

(Goal 6):

If R believes that T once conveyed the formula ($\mathbf{B} = \mathbf{z1} \otimes \mathbf{X}$), then it is entitled to believe that

$$\frac{R| \equiv T| \sim (\mathbf{B} = \mathbf{z1} \otimes \mathbf{X}), R| \equiv \#(\mathbf{v1})}{R| \equiv T| \sim \mathbf{v1}, R| \equiv T \ni \mathbf{v1}} \tag{23}$$

T once conveyed **v1**. And if R also believes that **v1** is fresh, then it is entitled to believe that T possesses **v1**. Therefore, R believes that **v1** is possessed by T:

(Goal 7):

If T believes that R is an authority on the statement $R \overset{\mathbf{v}_2}{\longleftrightarrow} T$ and T believes that R believe in $R \overset{\mathbf{v}_2}{\longleftrightarrow} T$, then T ought to believe in $T \overset{\mathbf{v}_2}{\longleftrightarrow} R$ as well. So, T believes that **v2** is a suitable secret between itself and R:

$$\frac{T| \equiv R| \Rightarrow R \overset{\mathbf{v2}}{\longleftrightarrow} T,\ T|\equiv R| \equiv R \overset{\mathbf{v2}}{\longleftrightarrow} T}{T| \equiv T \overset{\mathbf{v2}}{\longleftrightarrow} R} \tag{24}$$

(Goal 8):

$$\frac{T| \equiv R| \sim (\mathbf{D} = \mathbf{z2} \otimes \mathbf{X}), T| \equiv \#(\mathbf{v2})}{T| \equiv R| \sim \mathbf{v2}, T| \equiv R \ni \mathbf{v2}} \quad (25)$$

If T believes that R once conveyed the formula ($\mathbf{D}=\mathbf{z2} \otimes \mathbf{X}$), then it is entitled to believe that R once conveyed **v2**. And if T also believes that **v2** is fresh, then it is entitled to believe that R possesses **v2**. Therefore, T believes that **v2** is possessed by R:

As it can be seen, all eight defined goals have been achieved to demonstrate the completeness of the proposed protocol and we obtained our desired result. So, this protocol can be used as a lightweight protocol for small/cheap RFID tags.

5 Security Analysis

5.1 Resistance to GRS Attacks

To prove the resistance of the proposed protocol to GRS attacks, an attacker was assumed who could interrupt reader-tag communication and impersonate the reader and challenge the tag. He is also able to modify and interrupt reader-tag messages.

Figure 6 depicts a typical MIM attack against the proposed protocol. If the attacker receives challenge **a** from a legal reader (verifier) and sends it to a legal tag (prover) on behalf of the reader, he will receive matrix **B** in response. Assume that the attacker adds $\boldsymbol{\delta}_i$ to row $\mathbf{B}_i$ at each row of matrix **B** and sends matrix $\mathbf{B}'$ to the reader. Under these conditions, the tag is authenticated by the reader based on the values of **a**, $\mathbf{B}'$ and key matrix. If this process is successful, matrix **D** and if it is failed, matrix $\mathbf{D}'$ is sent by the reader and it is not known which rows of the matrix, $\boldsymbol{\delta i} * \mathbf{X}_i$ are 0 or 1. Therefore, no specific result is obtained under this condition. If the attacker correctly selects all $\boldsymbol{\delta}_i$s in the previous step, and authentication is successful, he obtained some part of **X**, because of $\boldsymbol{\delta} * \mathbf{X} = 0$. However, even under these conditions, all keys change and become invalid for the next authentication.

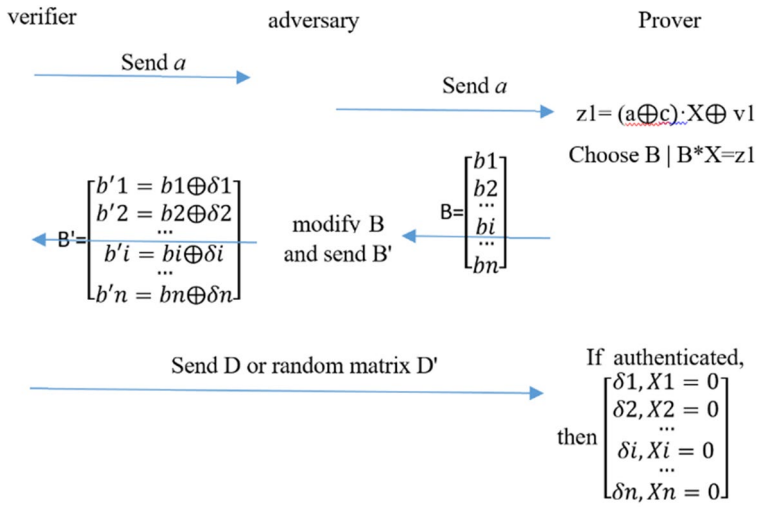

Fig. 6 A typical MIM attack to the proposed protocol

5.2 Protection Against Eavesdropping

As can be seen from the definition of the protocol, the first modification is performed on the format of the keys and responses. The basic operations employed here are AND and XOR binary operations similar to those in HB and HB+. We assumed an attacker eavesdropping reader-tag messages. The keys are not directly sent over the communication channel and the attacker will access only **a** and **B**. Munilla and Peinado [14] showed that finding **X** based on **B** is at least as hard as solving LPN problem Consequently, the practical security of our protocol against eavesdropping is confirmed.

5.3 Message Loss and Desynchronization Attacks

In this attack, the attacker is assumed to be able to modify the shared data to desynchronize the communication of the parties. In our proposed protocol, however, parties do not directly exchange shared data and after successful authentication, keys are independently updated on both sides. More precisely, as described above, two key matrices are stored for each tag in the reader. If any messages in steps 1 or 3 are lost, authentication fails and keys are not changed. When the reader authenticates the tag, the key is changed and when other messages in step 6 are lost, the connection is recovered due to access to the key matrix of the previous step ($\mathbf{X_{recover}}$). Therefore, our proposed protocol is immune to desynchronization attacks.

5.4 Replay Attack

In our protocol, the reader and the tag generate k-bit random binary vectors **a** and **c**, respectively, and responses (**B** and **D**) are generated and verified according to these vectors. Therefore, the adversary cannot repeatedly send challenges and receive the same response. Assume the adversary changes the value of **a** to **a′**. Since the tag generates a response using the random sequence **c** in combination with **a′**, and the reader employs the vector **a** in its calculations, then no valuable response is obtained by the adversary and thus replay attack is impossible.

5.5 Forward Secrecy

Secret matrices of the reader and the tag change according to the random vectors **v1** and **v2**. These values are never sent over the channel and are not accessible by adversary. To obtain the values, the adversary will have to solve the LPN problem which is NP-Hard. Furthermore, random challenge vectors **a** and **c** are independently generated in each session and the keys changed based on them. Consequently, secret information will not be accessible for attackers by storing the data exchanged on the communication channel.

5.6 Security Analysis of Distance Bounding and Solutions to Defeat Distance Fraud Attacks

Two major attacks of mafia and terrorist fraud threaten distance bounding protocols and the resistance of proposed protocol against them has to be confirmed.

5.6.1 Mafia Fraud or Relay Attack

In a mafia fraud, an active MIM adversary interacts with a verifier and a prover in several sessions tries to gain authentication. However, he cannot purely relay information between prover and verifier in time-critical phases [32].

The adversary trying to impersonate a tag should correctly response all challenges. He has two major attack strategies. He can simply estimate responses to challenges (no-ask strategy) or ask the tag some random challenges taking the risk of being detected by the tag (pre-ask strategy). The success probability of the adversary without asking is taken as P_{no-ask} and that with asking as $P_{pre-ask}$.

If pre ask strategy is chosen and random responses are generated, $P_{pre-ask}$ in each round is $\left(\frac{1}{2}\right)$. Similarly, P_{no-ask} is always $\left(\frac{1}{2}\right)^n$. To obtain $P_{pre-ask}$, the adversary is assumed to be able to query the tag in advance between the initialization and RBE phases with some arbitrary c_i^*. This way, the adversary achieves n bits of registers. If challenge c_i^* from the adversary to tag is similar to challenge c_i from reader the tag, he can send the response received from the tag to the reader. If $c_i^* \neq c_i$, he can send a random response to reader. Even if the tag detects the attack of the adversary, the reader may still not detect the attack. To evaluate this possibility, the following events are defined:

$\overline{a_i}$: when the attack is not detected at the ith round by the reader,
$\underline{b_i}$: when the attack is detected at the ith round by the tag,
$\overline{b_i}$: when the attack is not detected at the ith round by the tag,
A_i: when the attack is not detected until the ith round by the reader,
$\underline{B_i}$: when the attack is detected at the ith round by the tag for the first time,
$\overline{B_i}$: when the attack is not detected until the ith round by the tag.
P_d and P_r are defined as probability of a challenge being a predefined or random, respectively; therefore, $P_d + P_r = 1$.
$P\left(\overline{A_i}\right)$, which is the probability of remaining undetected by the reader until ith round, depends on whether the tag detects the attack in the previous rounds. Therefore:

$$P\left(\overline{A_i}\right) = P\left(\overline{A_i}|\overline{B_i}\right)P\left(\overline{B_i}\right) + \sum_{k=1}^{i} P\left(\overline{A_i}|B_k\right)P\left(B_k\right) \tag{26}$$

$$P\left(B_i\right) = \left(1 - \frac{P_d}{2}\right)^{i-1} \cdot \frac{P_d}{2} \tag{27}$$

and the probability of remaining undetected by the tag until ith round is:

$$P\left(\overline{B_i}\right) = \left(1 - \frac{P_d}{2}\right)^{i} \tag{28}$$

We can express

$$P\left(\overline{A_i}|B_k\right) = \prod_{j=1}^{k-1} P\left(\overline{a_j}|\overline{b_j}\right) \cdot \prod_{j=k}^{i} P\left(\overline{a_j}|b_k\right) \tag{29}$$

Remaining undetected by the reader in j th round, depends on the possibility of being detected by the tag in similar round. Random values are given by the tag once an error is detected and the adversary cannot verify the authority of the data; therefore:

$$\begin{aligned} P\left(\overline{a_j}|\overline{b_j}\right) &= \mathrm{P}_{\text{random ch.}} \quad \& \quad \textit{not detected} + \mathrm{P}_{\text{predefined ch.}} \quad \& \quad \textit{not detected} \\ &= p_r \cdot \left(P\left(c_j^* = c_j \quad \& \quad \textit{not det. by Reader}\right) + P\left(c_j^* \neq c_j \quad \& \quad \textit{not det. by Reader}\right)\right) \\ &\quad + \mathrm{p}_r \cdot \left(\mathrm{P}\left(\mathrm{c}_j^* = c_j \quad \& \quad \text{not det.by Reader}\right) + \mathrm{P}\left(\mathrm{c}_j^* \neq c_j \quad \& \quad \text{not det.by Reader}\right)\right) \\ &= \mathrm{P}_r\left(\frac{1}{2} \cdot 1 + \frac{1}{2} \cdot \frac{1}{2}\right) + P_d\left(\frac{1}{2} \cdot 1 + \frac{1}{2} \cdot \frac{1}{2}\right) = \frac{3}{4}P_r + \frac{3}{4}P_d = \frac{3}{4}. \end{aligned} \tag{30}$$

If $\mathrm{c}_j^* = c_j$ and the challenge is random, the adversary can correctly answer the response. However, if $\mathrm{c}_j^* \neq c_j$ and the challenge is random, the adversary has a $\frac{1}{2}$ chance to provide a correct response. Also, if $\mathrm{c}_j^* = c_j$ and the challenge is predefined, the attacker can provide a correct answer to response. Finally, If $\mathrm{c}_j^* \neq c_j$ and the challenge is predefined, the attacker has a $\frac{1}{2}$ chance to give a correct response to reader despite the fact that he is always detected by the tag.

The probability remaining undetected by the reader in jth round when tag detects the attack in k th round with $k \leq j$, $P\left(\overline{a_j}|b_k\right)$, is:

$$\begin{aligned} P\left(\overline{a_j}|b_k\right) &= \mathrm{P}_{\text{random ch.}} \quad \& \quad \textit{not detected} + \mathrm{P}_{\text{predefined ch.}} \quad \& \quad \textit{not detected} \\ &= \mathrm{p}_r \cdot \left(\mathrm{P}\left(\mathrm{c}_j^* = c_j \quad \& \quad \text{not det.by Reader}\right) + \mathrm{P}\left(\mathrm{c}_j^* \neq c_j \quad \& \quad \text{not det.by Reader}\right)\right) \\ &\quad + \mathrm{p}_r \cdot \left(\mathrm{P}\left(\mathrm{c}_j^* = c_j \quad \& \quad \text{not det.by Reader}\right) + \mathrm{P}\left(\mathrm{c}_j^* \neq c_j \quad \& \quad \text{not det.by Reader}\right)\right) \\ &= \mathrm{P}_r\left(\frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2}\right) + P_d\left(\frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2}\right) = \frac{1}{2}P_r + \frac{1}{2}P_d = \frac{1}{2}. \end{aligned} \tag{31}$$

If $\mathrm{c}_j^* = c_j$ and the challenge is random, the attacker, not knowing that he was detected in previous round, sends the same response with tag. Because the tag has detected an error in the previous round, its response is a random value. Therefore, the attacker has a $\frac{1}{2}$ chance of remaining undetected by the reader. Also, when $\mathrm{c}_j^* \neq c_j$ and the challenge is random, the chance of the attacker to correct response is $\frac{1}{2}$ since he has to choose a random response. If $\mathrm{c}_j^* = c_j$and the challenge is predefined, the attacker again sends the same response as the tag and he has a $\frac{1}{2}$ chance of remaining undetected by the reader. When $\mathrm{c}_j^* \neq c_j$and the challenge is predefined, his chance to give the correct answer is again $\frac{1}{2}$ because he chooses a random response.

From (20), (30) and (31), we have:

$$P\left(\overline{A_i}|B_k\right) = \prod_{j=1}^{k-1} P\left(\overline{a_j}|\overline{b_j}\right) \cdot \prod_{j=k}^{i} P\left(\overline{a_j}|b_k\right) = \prod_{j=1}^{k-1} \frac{3}{4} \cdot \prod_{j=k}^{i} \frac{1}{2} = \left(\frac{3}{4}\right)^{k-1} \cdot \left(\frac{1}{2}\right)^{i-k+1} \tag{32}$$

Similarly:

$$P\left(\overline{A_i}|\overline{B_i}\right) = \prod_{j=1}^{i} P\left(\overline{a_j}|\overline{b_j}\right) = \left(\frac{3}{4}\right)^{i} \tag{33}$$

From Eqs. (26), (27), (28), (32) and (33), we can finally obtain the probability of remaining undetected by the reader until i th round as:

$$
\begin{aligned}
P\left(\overline{A_i}\right) &= P\left(\overline{A_i}|\overline{B_i}\right)P\left(\overline{B_i}\right) + \sum_{k=1}^{i} P\left(\overline{A_i}|B_k\right)P\left(B_k\right) \\
&= \left(\frac{3}{4}\right)^i \cdot \left(1 - \frac{P_d}{2}\right)^i + \frac{P_d}{2} \cdot \sum_{k=1}^{i} \left(\frac{3}{4}\right)^{k-1} \cdot \left(1 - \frac{P_d}{2}\right)^{k-1} \cdot \left(\frac{1}{2}\right)^{i-k+1}
\end{aligned} \tag{34}
$$

$$
\Rightarrow \begin{cases} P_d = 0 \text{(random challenges always)} & P\left(\overline{A_i}\right) = \left(\frac{3}{4}\right)^i \\ P_d = 1 \text{(predefined challenges always)} & P\left(\overline{A_i}\right) = \left(\frac{3}{8}\right)^i + \left(\frac{1}{2}\right)^{i+1} \cdot \sum_{k=1}^{i} \left(\frac{3}{4}\right)^{k-1}. \end{cases}
$$

In the proposed protocol, the answer of the tag is always a random bit once an error is detected. It is noteworthy that once fast bit exchanges are finished, no confirmation message is used which improves computation and communication efficiency of the protocol. Furthermore, instead of hash functions, we used lightweight binary functions in initialization phase.

5.7 Terrorist Fraud

Until now, we have assumed that the tag was honest and the adversary tried to perform Mafia fraud attack. Here, we consider a dishonest tag. A terrorist fraud is a man-in-the-middle technique between a dishonest tag outside the neighborhood and the reader, such that the former actively helps the attacker maximize his success probability Terrorist out giving him an advantage for future attacks. A typical scenario for contactless authentication devices is a public transport system in which users authenticate to access buses or subway stations through their NFC-enabled smartphones. The transportation company must deploy controls to prevent misuse of its system. A legitimate user might want to help a friend to use his credentials illegally for a single trip while he is not using them, which is known as a terrorist fraud (TF). Nevertheless, this user would not accept that his friend uses them at will afterward as the original user may get caught and accountable. In other words, TF assumes that a malicious prover does not sufficiently trust his accomplice to directly authenticate him using potential long-term secret keys [31].

Suppose that the predefined challenge $\mathbf{t}$ is known by the illegal tag before starting fast bit exchange. Hence, the attacker may try to deceive the reader using terrorist fraud attack. Assuming that the legal tag is far away, for each challenge bit $\mathbf{c}[i]$, the legal tag will have to guess it in advance and pass the corresponding response to adversary, or pass right $\mathbf{z2}[i]$ and therefore $\mathbf{X}$ which is not acceptable. The success probability of terrorist fraud attack by a dishonest tag for a round can be expressed as:

$$
\begin{aligned}
\mathrm{P}_{\text{Terrorist Fraud attack}} &= \mathrm{P}_{\text{random ch. \& } \textit{deceive}} + \mathrm{P}_{\text{predefined ch. \& } \textit{deceive}} \\
&= \mathrm{P}_r \cdot (\mathrm{P}(\mathbf{q}[i] = \mathbf{r}[i] \quad \& \quad \text{deceive}) + \mathrm{P}(\mathbf{q}[i] \neq \mathbf{r}[i] \quad \& \quad \text{deceive})) + \mathrm{P}_d \\
&= \mathrm{P}_r\left(\frac{1}{2} \cdot 1 + \frac{1}{2} \cdot \frac{1}{2}\right) + P_d = \frac{3}{4}P_r + P_d = 1 - \frac{1}{4}P_r.
\end{aligned} \tag{35}
$$

If the ith bit of $\mathbf{q}[i]$and$\mathbf{r}[i]$ are equal and the challenge is a random ($\mathbf{t}[i]=0$), the tag can quickly send its response. If ith bit of $\mathbf{r}[i]$ and $\mathbf{q}[i]$ are not equal at $\mathbf{t}[i]=1$, the tag randomly chooses the response. Finally, if the challenge is predefined ($\mathbf{t}[i]=1$), it can quickly send its

response. Therefore, the overall probability of n round would be $\left(1-\frac{1}{4}P_r\right)^n$. If $P_r=\frac{1}{2}$, which is the average case when we use a pseudorandom function to generate a bit string, then we have $\left(\frac{7}{8}\right)^n$.

5.8 Error Rate

When the number of incorrect answers in the authentication phase exceeds a threshold level *tr,* false rejection (when a reader mistakenly rejects a legitimate tag) takes place. Also, when the number of random responses of the fraudulent tag is smaller than or equal to *tr,* false acceptance (when a reader mistakenly accepts a fraudulent tag) happens. In the proposed protocol, considerable flexibility is observed in setting acceptance threshold *tr*. It should be kept in mind that false rejection rate depends on η, *tr*, and m and false acceptance rate is determined based on *tr* and m only. Since the authentication is performed mutually and the process looks almost similar for both sides, the probabilities of false acceptance (P_{FA}) and false rejection (P_{FR}) are calculated as:

$$\begin{aligned} P_{FR} &= \sum_{i=tr+1}^{r}\binom{r}{i}\eta^i(1-\eta)^{r-i}\times\sum_{i=tr+1}^{r}\binom{r}{i}\eta^i(1-\eta)^{r-i} \\ P_{FA} &= \sum_{i=0}^{t}\binom{r}{i}2^{-r}\times\sum_{i=0}^{t}\binom{r}{i}2^{-r} \end{aligned} \tag{36}$$

In authentication phase, the proposed protocol takes the advantages of both HB# and HB-MP + protocols. To evaluate the costs and resources required for the execution of these protocols, such as communication and memory cost as well as computational power, acceptable error rate is observed to be similar. Levieil and Fouque [33] provided some example parameters for different noise levels η in HB + and obtained very reasonable error rates of $P_{FR}<2^{-40}$ and $P_{FA}<2^{-80}$. Table 4 summarizes the practical specifications of these protocols *to* obtain $P_{FA}=2.1\times10^{-25}$and $P_{FR}=8.7\times10^{-13}$.Before comparing the protocol proposed in this work with previously reported ones, one should note that in HB family protocols, authentication is carried out unilaterally and distance frauds cannot be defeated [30]. Therefore, the required resources must be compared in authentication phases.

In the proposed protocol, one challenge vector is sent and the response is transmitted as a matrix (instead of many vectors as in HB protocols) which decreases data overhead and accelerates transmission process. This allows the other party to process and compute the response in a shorter time. On the other hand, the proposed protocol suffers from higher transmission cost than HB# but it should be kept in the mind that HB# is vulnerable to GRS attack. The proposed protocol needs $(k+r-1)$ bits of memory which is lower than

Table 4 Comparison of parameters of protocols in terms of error rate

	k	***r***	***η***	***tr***	Memory	Transmitted data (bits)	Number of data transmission
Proposed protocol	512	142	0.1	31	653	452,608	4
Hb#	512	441	0.125	113	1904	2066	4
HB-MP+	512	441	0.125	113	512	903,168	1764

Table 5 Security analysis of HB protocols and the proposed protocol

Protocol	Light weight	Distance bounding	Passive attack	GRS attack	Mutual authentication	Confidentially	Integrity
HB (2001)	Yes	NO	Not secure	Not secure	No	No	No
HB + (2005)	Yes	NO	Not secure	Not secure	No	No	No
HB ++ (2006)	Yes	NO	Secure	Not secure	No	Yes	No
HB-MP + (2008)	NO	NO	Secure	secure	No	Yes	Yes
AUHB ++ (2008)	NO	NO	Secure	Not secure	No	Yes	Yes
PUF-HB (2008)	NO	NO	Secure	Not secure	No	Yes	No
HB# (2008)	Yes	NO	Secure	Not secure	No	Yes	No
HB-MP ++ (2009)	NO	NO	Secure	Not secure	No	Yes	No
NL-HB (2010)	Yes	NO	Not secure	Not secure	No	No	No
RC-HB (2011)	Yes	NO	Not secure	Not secure	No	No	No
IHB (2015)	Yes	NO	Secure	Not secure	No	Yes	No
HB + DB (2017)	Yes	Yes	Secure	Not secure	Yes	Yes	Yes
Proposed protocol	Yes	Yes	Secure	Secure	Yes	Yes	Yes

$k_X + k_Y + 2\,m - 2$ bits required by HB# and higher than k bits needed by HB-MP + . Table 5 compares the most significant attacks to HB and our proposed protocol.

6 Conclusion

Considering the increasing application of IoT in today's life, its security and privacy requirements and the vulnerabilities of existing authentication protocols are very important issues. In this paper, we proposed a novel Ultra-Lightweight Mutual Authentication Protocol based on LPN Problem with three major properties: it can be applied as a secure and lightweight protocol in low cost resource-limited devices like NFC and RFID; Due to the property of LPN problem, it is also applicable in post-quantum cryptography; it resolves weaknesses of HB protocols such as GRS attack; and it is resistant to terrorist fraud, mafia fraud, and distance fraud attacks without using final digital signature. It was proved that the proposed protocol was able to address known weaknesses and attacks of HB protocols.

Funding None.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Informed consent None.

References

1. He, L., Gan, Y., Li, N.-N., & Zhang, T. (2008). An improved HB++ protocol against man-in-middle attack in RFID system. In *2008 4th international conference on wireless communications, networking and mobile computing* (pp. 1–4).
2. Bogos, S., & Vaudenay, S. (2016). Optimization of LPN solving algorithms. In *International conference on the theory and application of cryptology and information security* (pp. 703–728).
3. Bogos, S. M. (2017). LPN in cryptography: An algorithmic study. *Ecole Polytechnique Fédérale de Lausanne.*
4. Brakerski, Z., Lyubashevsky, V., Vaikuntanathan, V., & Wichs, D. (2019). Worst-case hardness for LPN and cryptographic hashing via code smoothing. In *Annual international conference on the theory and applications of cryptographic techniques* (pp. 619–635).
5. Hopper, N. J., & Blum, M. (2000). A secure human–computer authentication scheme. Carnegie-Mellon Univ Pittsburgh Pa School Of Computer Science.
6. Kitsos, P. (2016). *Security in RFID and sensor networks*. Boca Raton: CRC Press.
7. Gilbert, H., Robshaw, M., & Sibert, H. (2005). Active attack against HB/sup+: a provably secure lightweight authentication protocol. *Electronics Letters, 41,* 1169–1170.
8. Avoine, G., Bultel, X., Gambs, S., Gerault, D., Lafourcade, P., Onete, C., et al. (2017). A terrorist-fraud resistant and extractor-free anonymous distance-bounding protocol. In: *Proceedings of the 2017 ACM on Asia conference on computer and communications security* (pp. 800–814).
9. Boureanu, I., Mitrokotsa, A., & Vaudenay, S. (2015). Practical and provably secure distance-bounding. *Journal of Computer Security, 23,* 229–257.
10. Karrothu, A., Scholar, R., & Norman, J. (2017). An analysis of LPN based HB protocols. In *2016 eighth international conference on advanced computing (ICoAC)* (pp. 138–145).
11. Juels, A., & Weis, S. A. (2005). Authenticating pervasive devices with human protocols. In *Annual international cryptology conference* (pp. 293–308).
12. Bringer, J., Chabanne, H., & Dottax, E. (2006). HB^+^+: a lightweight authentication protocol secure against some attacks. In Second international workshop on security, privacy and trust in pervasive and ubiquitous computing (SecPerU'06) (pp. 28–33).
13. Hammouri, G., Sunar, B. (2008). PUF-HB: a tamper-resilient HB based authentication protocol. In *International conference on applied cryptography and network security* (pp. 346–365).
14. Munilla, J., & Peinado, A. (2007). HB-MP: a further step in the HB-family of lightweight authentication protocols. *Computer Networks, 51,* 2262–2267.
15. Leng, X., Mayes, K., & Markantonakis, K. (2008). HB-MP+protocol: An improvement on the HB-MP protocol. In *2008 IEEE international conference on RFID* (pp. 118–124).
16. Gilbert, H., Robshaw, M., & Seurin, Y. (2008). HB#: Increasing the security and efficiency of HB+. In *Proceedings of international conference the theory and applications of cryptographic techniques advances in cryptology (EUROCRYPT)*
17. Ouafi, K., Overbeck, R., & Vaudenay, S.: On the security of HB# against a man-in-the-middle attack. In *International conference on the theory and application of cryptology and information security* (pp. 108–124).
18. Yoon, B., Sung, M. Y., Yeon, S., Oh, H. S., Kwon, Y., et al. (2009). HB-MP++protocol: an ultra light-weight authentication protocol for RFID system. In *2009 IEEE international conference on RFID* (pp. 186–191).
19. Madhavan, M., Thangaraj, A., Viswanathan, K., & Sankarasubramaniam, Y. (2010). NLHB: a light-weight, provably-secure variant of the HB protocol using simple non-linear functions. In *2010 national conference on communications (NCC)* (pp. 1–5).
20. S. A. Ali, R. M. Mohamed, and M. H. Fahim, "RCHB: Light-weight, provably-secure variants of the HB protocol using rotation and complementation," in 2011 5th International Conference on Network and System Security, 2011, pp. 244-248.
21. K. A. Khoureich, "Light-hHB: A new version of hHB with improved session key exchange," Cryptology ePrint Archive, Report 2015/713, 2015.

22. Lin, Z., & Song, J. S. (2013). An improvement in HB-family lightweight authentication protocols for practical use of RFID system. *Journal of Advances in Computer Networks, 1,* 61–65.
23. Kiltz, E., Pietrzak, K., Cash, D., Jain, A., & Venturi, D., (2011). Efficient authentication from hard learning problems. In *Annual international conference on the theory and applications of cryptographic techniques* (pp. 7–26).
24. Brands, S., & Chaum, D. (1993). Distance-bounding protocols. In *Workshop on the theory and application of of cryptographic techniques* (pp. 344–359).
25. Brelurut, A., Gerault, D., & Lafourcade, P. (2015). Survey of distance bounding protocols and threats. In *International symposium on foundations and practice of security* (pp. 29–49).
26. Ahmadi, A., Safavi-Naini, R. (2014). Privacy-preserving distance-bounding proof-of-knowledge. In *International conference on information and communications security* (pp. 74–88).
27. Bussard, L., & Bagga, W. (2005). Distance-bounding proof of knowledge to avoid real-time attacks. In *IFIP international information security conference* (pp. 223–238).
28. Munilla, J., Ortiz, A., & Peinado, A. (2006). Distance bounding protocols with void-challenges for RFID. In *Printed handout at the workshop on RFID security—RFIDSec*.
29. Kim, C. H. & Avoine, G.: RFID distance bounding protocol with mixed challenges to prevent relay attacks. In *International conference on cryptology and network security* (pp. 119–133)
30. Pagnin, E., Yang, A., Hu, Q., Hancke, G., & Mitrokotsa, A. (2018). HB + DB: Distance bounding meets human based authentication. *Future Generation Computer Systems, 80,* 627–639.
31. Desmedt, Y. (1988). Major security problems with the 'unforgeable'(Feige)-Fiat-Shamir proofs of identity and how to overcome them. In *Proceedings of SECURICOM* (pp. 15–17).
32. Fischlin, M., & Onete, C. (2013). Terrorism in distance bounding: modeling terrorist-fraud resistance. In *International conference on applied cryptography and network security* (pp 414–431).
33. Levieil, É., & Fouque, P.-A. (2006). An improved LPN algorithm. In *International conference on security and cryptography for networks* (pp. 348–359).

Kazem Mirzadi earned his Bachelor of Engineering degree in software Engineering from Shahid Bahonar University in 2009. He received his Master of Science degree in information technology from Kerman Graduate University of Technology in 2012. Currently, he is a Ph.D. student in the information technology faculty at Urmia university. Previously, he has been working as an IT Manger and software engineer in various technological companies. His research interests focus on the network security, RFID, IoT and its applications. He is specially involved in the research field of security protocols for the IoT.

Jamshid B. Mohasefi received his B.S. in computer engineering from Sharif University of Technology in Iran at 1996 and M.S. in computer engineering from Tarbiat Modares University in Iran at 1999. He got his PhD in computer engineering from Indian Institute of Technology Delhi (IITD) in India at 2006. He is currently associate professor and dean of computer and electrical engineering faculty at Urmia University, Iran. He is also head of Computer Emergency Response Team (CERT) center at Urmia University, Iran. His current research interests include network security, machine learning, and internet of things.