Computationally Efficient Mutual Entity Authentication in Wireless Sensor Networks

Zhijun Li and Guang Gong

University of Waterloo, Waterloo ON N2L3G1, Canada
leezj@engmail.uwaterloo.ca, ggong@calliope.uwaterloo.ca
http://comsec.uwaterloo.ca

Abstract. Mutual entity authentication plays an important role in securing wireless sensor networks. In this paper, we present a computationally efficient authentication framework, based on learning parity with noise problem. The authentication only requires the simplest bit-operations, which makes it suitable for resource-restrained wireless sensor networks. The framework not only presents an approach to securely combine two one-way authentication protocols from the HB-family, but also provides significant enhancements in terms of feasibility of storage/communication requirement. It spawns three specific protocols with different trade-offs between communication overload and memory cost. We extensively analyze their performance and security properties. Furthermore, their applications in different wireless sensor network scenarios are discussed in detail.

Keywords: mutual entity authentication, computationally efficient, learning parity with noise, wireless sensor networks, HB protocol, HB-hybrid.

1 Introduction

Wireless sensor networks (WSNs) are innovative ad-hoc wireless networks consisting of a large number of sensor nodes with limited power, computation, storage and communication capabilities [1]. The basic function of wireless sensor networks is to collect information for authorized users. Typically, base stations or users issue various commands of tasks to nodes; then nodes start to work accordingly, gathering data and forwarding to base stations or users. To function properly, base stations and users should be authenticated to be the acclaimed entities by nodes. This is because, without entity authentication, adversaries can easily abuse the sensor networks to collect information maliciously or launch energy-exhaustion denial-of-service attacks by frequently ordering nodes to perform nonsense tasks. On the other hand, entities of nodes should also be authenticated by other nodes, base stations, and users. Otherwise, adversaries can insert invalid nodes into sensor networks to corrupt the result of information collection. Moreover, any further advanced access control mechanisms require entity authentication. In a word, mutual entity authentication plays a significant role in security of wireless sensor networks.

J. Zheng et al. (Eds.): ADHOCNETS 2009, LNICST 28, pp. 192–207, 2010.

Some entity authentication schemes in wireless sensor networks have been proposed. Benenson, Gedicke, and Raivio [2] introduced an entity authentication scheme of WSNs, based on elliptic curve cryptography. Jiang and Xu [3] presented a distributed entity authentication scheme in wireless sensor networks. It is built upon the self-certified keys cryptosystem, which is modified to use elliptic curve cryptography to establish pair-wise keys for use in the entity authentication scheme. Wong *et al.* [4] proposed a dynamic strong-password-based entity authentication scheme for WSNs; then Tseng, Jan, and Wang [5] enhanced Wong *et al.*'s scheme to thwart potential replay and forgery attacks. Tripathy and Nandi [6] used cellular automata based components to achieve entity authentication.

All of the above are based on conventional cryptographic mechanisms, symmetric or public-key. Since sensor networks consist of a large number of sensor nodes, the cost of a single node is very important to justify the overall cost of the network. In many applications of sensor networks, the production cost of nodes would dominate the success of the system. Akyildiz *et al.* [1] argued that the cost of a sensor node should be much less than one dollar in order for sensor networks to be feasible. Under this constraint, sensor nodes on some applications may not be equipped with necessary hardware to perform costly standard cryptographic operations, even symmetric primitives.

In this paper, we propose three computationally efficient mutual entity authentication protocols for sensor networks. The protocols are based on a well-studied hard problem: learning parity with noise (LPN). All they require is bit-operations as well as a random number generator. Almost all entity authentication schemes include one or several rounds of challenge-response interactions, of which randomization is a necessary part. Therefore, the facility of a random number generator is indispensable for each participant in mutual entity authentication. There are many methods for sensor nodes to generate random numbers, such as deriving from environmental and circuital noise. As far as we know, the proposed schemes are the first attempt to design bit-operation-based mutual entity authentication protocols in wireless sensor networks. We do not intend to substitute the existing approaches—our proposal has different application areas from existing schemes.

The remainder of this paper is organized as follows. First we introduce the LPN problem and address the previous LPN-based one-way entity authentication schemes in Section 2. Then we describe our protocols, evaluate and compare their performance in Section 3. Extensive security analysis on the proposed protocols is given in Section 4. Afterward, the application scenarios of the proposed protocols in sensor networks are addressed in Section 5. Finally, Section 6 concludes our work and gives some further research directions.

2 LPN Problem and HB-Family

For convenient discussions, we name the two participants in an authentication procedure Alice and Bob. For one-way authentication, we assume that Alice is

the entity who would like to authenticate herself to the other participant, Bob. The following notations are used throughout this paper.

- $\boldsymbol{b} \circ \boldsymbol{y}$: the binary inner-product of two vectors (or matrices) $\boldsymbol{b}$ and $\boldsymbol{y}$.
- $\boldsymbol{b} \oplus \boldsymbol{y}$: the bitwise exclusive-or operation of two vectors (or matrices) $\boldsymbol{b}$ and $\boldsymbol{y}$.
- Hwt($\boldsymbol{y}$): the Hamming weight of the binary vector (or matrix) $\boldsymbol{y}$, that is, the number of bit '1' in the vector (or matrix).

2.1 LPN Problem

Alice holds a secret binary vector $\boldsymbol{y}$ of length k. Given a sequence of randomly chosen binary vectors $\boldsymbol{b}_1, \cdots, \boldsymbol{b}_q$ along with the values of inner-product $z_i = \boldsymbol{y} \circ \boldsymbol{b}_i$ with $\boldsymbol{y}$, an adversary can easily reconstruct $\boldsymbol{y}$ using Gaussian elimination, as long as q is slightly larger than k such that the set $\{\boldsymbol{b}_i\}$ contains k linearly-independent vectors.

In the presence of noise, however, where each bit z_i is independently exclusive-or'ed (XORed) by a noise bit taking '1' with probability $\eta \in (0, \frac{1}{2})$, determining $\boldsymbol{y}$ becomes much more difficult. This problem is known as Learning Parity with Noise, or the *LPN Problem*. Formally, it is defined as follows:

Definition 1 (LPN Problem). *Let $\mathbb{B}$ be a random $(q \times k)$-binary matrix, let $\boldsymbol{y}$ be a random k-bit vector, let $\eta \in (0, \frac{1}{2})$ be a noise level, and let $\boldsymbol{v}$ be a random q-bit vector such that Hwt($\boldsymbol{v}$) $\leq \eta q$. Given $\mathbb{B}$, η, and $\boldsymbol{z} = (\mathbb{B} \circ \boldsymbol{y}^T) \oplus \boldsymbol{v}^T$, find a k-bit vector $\boldsymbol{y'}$ such that Hwt($(\mathbb{B} \circ \boldsymbol{y'}^T) \oplus \boldsymbol{z}$) $\leq \eta q$.*

The LPN problem is an average-case version of the following problem: given a set of equations over binary finite field $GF(2)$, find a vector $\boldsymbol{y}$ that maximally satisfies the equations. The latter problem was also formalized and referred to as the minimal disagreement parity problem [7], or the problem of finding the closest vector to a random linear error-correcting code; also known as the syndrome decoding problem [8,9]. The latter problem is known to be NP-Hard [8], and has been proven to be hard to even find a vector satisfying more than half of the challenge-response pairs [10]. However, the random instances in the LPN problem do not represent the worst case of the latter problem, and the study of the hardness of the LPN problem is still in progress [11,12,13,14].

2.2 HB-Family Authentication

All protocols discussed in this section are one-way entity authentication. The term *"proven security"* of a protocol means that this protocol, under certain models, can be reduced to the LPN problem, which we believe is hard. In other words, if an adversary can break the protocol under its model, then he can successfully solve the LPN problem.

Hopper and Blum [11] first presented an authentication protocol (HB protocol) based on the LPN problem. In the HB protocol, Alice and Bob have a secret vector $\boldsymbol{y}$ of length k in common. They interact n rounds of two passes for authentication. In each round, Bob generates and sends a random binary vector $\boldsymbol{b}$ as a

challenge; and then Alice responds with the inner-product of the challenge vector and the secret $\boldsymbol{y}$, but with noise of probability η on purpose. After n rounds, Alice is authenticated provided the number of rejected challenge-response pairs is not greater than $n\eta$.

Hopper and Blum [11] proved that the HB protocol is secure against passive eavesdroppers. However, an active adversary [15] can easily overcome the noise and then recover $\boldsymbol{y}$. To defend against this active attack, Juels and Weis [15] proposed HB$^+$ three-pass authentication protocol. HB$^+$ still involves n rounds; but Alice and Bob have to share two secret k-bit vectors $\boldsymbol{x}$ and $\boldsymbol{y}$. In each round, a blinding-factor vector $\boldsymbol{r}_1$ is first randomly generated by Alice and sent to Bob. Then Bob selects a challenge vector $\boldsymbol{r}_2$ at random. After receiving $\boldsymbol{r}_2$, Alice generates a noise bit v which takes '1' with probability η, computes $z = (\boldsymbol{r}_1 \circ \boldsymbol{x}^T) \oplus (\boldsymbol{r}_2 \circ \boldsymbol{y}^T) \oplus v$, and transmits z to Bob. Bob independently computes $z' = (\boldsymbol{r}_1 \circ \boldsymbol{x}^T) \oplus (\boldsymbol{r}_2 \circ \boldsymbol{y}^T)$, and validates Alice's response if $z = z'$. Similar to HB, after n rounds, the authentication succeeds if no more than $n\eta$ responses do not match challenges.

The HB$^+$ protocol is secure under the *detection-based model* [15,16,17]. Once again, despite that the HB$^+$ protocol has those security proofs, a new attack [18] is discovered, since the detection-based model used in the proofs is relatively restrictive. Gilbert, Robshaw, and Sibert [18] have shown that there exists a man-in-the-middle (MIM) active attack (GRS attack) against the HB$^+$ protocol.

GRS Attack: In second pass of every round of one HB$^+$ authentication procedure, an active adversary intercepts the challenge $\boldsymbol{r}_2$, and replaces $\boldsymbol{r}_2$ with $\boldsymbol{r}_2 \oplus \boldsymbol{\delta}$, where $\boldsymbol{\delta}$ is a constant vector for one authentication procedure. Then the adversary can learn the result of $\boldsymbol{\delta} \circ \boldsymbol{y}$ according to acceptance or rejection of this authentication procedure. That is, the adversary can discover one bit of $\boldsymbol{y}$. He simply repeats k times of manipulating authentication procedures, and then fully recovers secret $\boldsymbol{y}$. Holding secret $\boldsymbol{y}$, the adversary can successfully impersonate Alice by setting $\boldsymbol{r}_1 = \{0\}^k$. Alternatively, the same method can be applied to compromising secret $\boldsymbol{x}$.

Since the discovery of the GRS attack, a variety of protocols built upon HB$^+$, such as HB^{++} [19], HB* [20], HB-MP [21], have been proposed, intending to thwart the GRS attack. However, Gilbert, Robshaw, and Sibert [22] showed that these three variants can be attacked using corresponding techniques in the linear time. The further modification version of HB^{++} proposed by Piramuthu [23] and HB-MP$^+$ [24] are insecure because of their flawed basis. The PUF-HB protocol [25] and the Trust-HB protocol [26] make use of a physically unclonable circuit and a lightweight hash function family respectively, intending to resist man-in-the-middle attacks against HB$^+$. The introduction of such elements into HB$^+$ might not fully meet the motivation of designing computationally efficient authentication protocol.

Gilbert, Robshaw, and Seurin [27] presented the Random-HB$^\#$ protocol, which can resist the GRS attack and is proved secure under the *GRS-MIM model*. Surprisingly, the Random-HB$^\#$ protocol only needs one round. Instead of secret vectors, Alice and Bob share two secret matrices: $(k_X \times n)$-binary matrix $\mathbb{X}$ and

$(k_Y \times n)$-binary matrix $\mathbb{Y}$. First, Alice generates a random blinding-factor vector $\boldsymbol{r}_1$ of k_X bits and sends it to Bob; Bob generates a random challenge vector $\boldsymbol{r}_2$ of k_Y bits and transmits it to Alice. Then Alice randomly chooses a n-bit noise vector $\boldsymbol{v}$ with respect to noise level $\eta \in (0, \frac{1}{2})$, computes $\boldsymbol{z} = (\boldsymbol{r}_1 \circ \mathbb{X}) \oplus (\boldsymbol{r}_2 \circ \mathbb{Y}) \oplus \boldsymbol{v}$, and transmits vector $\boldsymbol{z}$ to Bob. Bob accepts the authentication if and only if $\mathrm{Hwt}((\boldsymbol{r}_1 \circ \mathbb{X}) \oplus (\boldsymbol{r}_2 \circ \mathbb{Y}) \oplus \boldsymbol{z}) \leq \tau$, where τ is the designated pass-threshold, which is an integer less than $\frac{n}{2}$ and greater than ηn.

The trade-off of Random-HB$^{\#}$ protocol is the high memory consumption: $n(k_X + k_Y)$ bits. To improve the storage performance, Gilbert, Robshaw, and Seurin suggested using the *Toeplitz* matrix to encode matrices $\mathbb{X}$ and $\mathbb{Y}$, which leads to their final proposed version—HB$^{\#}$ protocols. A Toeplitz matrix is a matrix in which the elements on every upper-left to lower-right diagonal have the same value, and is stipulated by the top row and the first column. Therefore, a $(k \times n)$-binary Toeplitz matrix can be stored in $k + n - 1$ bits rather than kn bits. The HB$^{\#}$ protocol is conjectured to be secure under the *GRS-MIM model*, and the authors give some supportive arguments about the conjecture.

3 Proposed Mutual Authentication Protocols

Now we introduce our three HB-hybrid mutual authentication protocols. The parameter set used in our protocols is (k, η, τ, l, m), where k is the *secret length*, η is the *noise level*, τ is the *pass-threshold*, l is the *secret-expansion*, and m is the *interaction-expansion*. The product of l and m acts as the round number n in the HB^{+} protocol or the secret matrix column dimension n in the HB$^{\#}$ protocol, whereas different combinations of l and m could provide the desired balance between storage requirement and communication overload.

3.1 Protocols Description

Our three protocols use the same HB-hybrid mutual authentication framework. The main difference among them is how to encode secret matrices and challenge matrices.

In the HB-hybrid mutual authentication framework, Alice and Bob share two $(k \times l)$-binary matrices $\mathbb{X}$ and $\mathbb{Y}$ as secrets. First, Alice randomly generates an $(m \times k)$ challenge matrix $\mathbb{R}_1$ and sends it to Bob; Bob generates an $(m \times k)$ challenge matrix $\mathbb{R}_2$ at random and sends it to Alice. Then Alice chooses an $(m \times l)$ noise matrix $\mathbb{V}$, in which each element takes value '1' with probability $\eta \in (0, \frac{1}{2})$, and calculates the $(m \times l)$ response matrix $\mathbb{Z}_1 = (\mathbb{R}_1 \circ \mathbb{X}) \oplus (\mathbb{R}_2 \circ \mathbb{Y}) \oplus \mathbb{V}$. Alice also generates a third $(m \times k)$ challenge matrix $\mathbb{R}_3$, and transmits $\mathbb{Z}_1, \mathbb{R}_3$ to Bob. Bob first verifies if $\mathrm{Hwt}((\mathbb{R}_1 \circ \mathbb{X}) \oplus (\mathbb{R}_2 \circ \mathbb{Y}) \oplus \mathbb{Z}_1) \leq \tau$, where integer pass-threshold $\tau \in (\eta ml, \frac{ml}{2})$. If the check fails, the authentication procedure is terminated. If it passes, Bob goes on to choose his $(m \times l)$ noise matrix $\mathbb{W}$, computes $(m \times l)$ response matrix $\mathbb{Z}_2 = (\mathbb{R}_2 \circ \mathbb{X}) \oplus (\mathbb{R}_3 \circ \mathbb{Y}) \oplus \mathbb{W}$, and send $\mathbb{Z}_2$ to Alice. Likewise, Alice accepts Bob's authentication if and only if $\mathrm{Hwt}((\mathbb{R}_2 \circ \mathbb{X}) \oplus (\mathbb{R}_3 \circ \mathbb{Y}) \oplus \mathbb{Z}_2) \leq \tau$. The HB-hybrid authentication framework is depicted in Fig. 1.

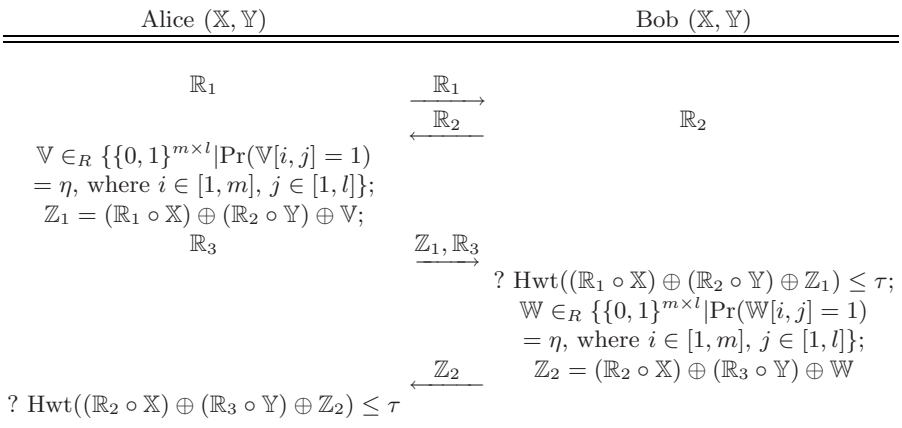

Fig. 1. HB-hybrid mutual authentication framework, where secrets $\mathbb{X}$ and $\mathbb{Y}$ are $(k \times l)$-binary matrices; challenges $\mathbb{R}_1$, $\mathbb{R}_2$ and $\mathbb{R}_3$ are $(m \times k)$-binary matrices; $\mathbb{V}$, $\mathbb{W}$, $\mathbb{Z}_1$ and $\mathbb{Z}_2$ are $(m \times l)$-binary matrices; noise level $\eta \in (0, \frac{1}{2})$, integer pass-threshold $\tau \in (\eta ml, \frac{ml}{2})$

If the two secret matrices $\mathbb{X}$, $\mathbb{Y}$ and the three challenge matrices $\mathbb{R}_1$, $\mathbb{R}_2$, $\mathbb{R}_3$ are all random matrices, then the protocol is referred to as HB-hybrid 1. If only the three challenge matrices $\mathbb{R}_1$, $\mathbb{R}_2$, and $\mathbb{R}_3$ are random matrices while the two secret matrices $\mathbb{X}$ and $\mathbb{Y}$ are Toeplitz matrices, then it is called HB-hybrid 2 protocol. If we allow the three challenge matrices $\mathbb{R}_1$, $\mathbb{R}_2$, $\mathbb{R}_3$ to be Toeplitz matrices and keep the two secret matrices $\mathbb{X}$, $\mathbb{Y}$ to be random, then it is named HB-hybrid 3 protocol. Furthermore, the HB-hybrid mutual authentication framework can degenerate to one-way authentication, which consists of three-pass, including $\mathbb{R}_1$, $\mathbb{R}_2$, $\mathbb{V}$, and $\mathbb{Z}_1$. Correspondingly, we have three one-way authentication protocols: HB-hybrid-OW 1, HB-hybrid-OW 2, and HB-hybrid-OW 3.

3.2 Protocol Parameters

We denote the desired security level of authentication protocols by d. Generally speaking, we desire to achieve at least 80-bit security. We select $d = 80$ in the following discussion.

Secret Length and Noise Level

The two parameters k (secret length) and η (noise level) dominate the security level d of LPN-based authentication protocols. The lower η is, the easier an adversary can overcome noise. On the other hand, the false negative rate of LPN-based schemes would be too high to be acceptable if η is approaching 0.5. Consequently, 0.25 might be an optimal option for η. Many proposals choose 0.125 as their experimental noise level. However, a probabilistic passive attack, recently proposed by Carrijo, Tonicelliy, and Imai [28], showed that the low noise level would undermine the security dramatically, while $\eta = 0.25$ resists this attack well. Therefore, we recommend $\eta = 0.25$ in the LPN-based protocols.

The desirable value of k is a reflection of the running time of current best algorithm solving LPN instances [12,13,14]. Blum, Kalai, and Wasserman [12] proposed a sub-exponential algorithm (BKW algorithm) of running time $2^{O(\frac{k}{\log k})}$, which is widely cited in the literatures. According to the BKM algorithm, the parameter set $(\eta = 0.25, k = 224)$ can provide 80-bit security. However, the LF algorithm, presented by Levieil and Fouque [14] as an enhancement of the BKM algorithm, shows that 2^{53} operations are sufficient for this parameter set. To reach the intended security level, k should be 512. To be precise, according to the LF algorithm, the secret length $k = 512$ provides 89-bit security and $k = 768$ provides 131-bit security under the condition $\eta = 0.25$. With performance improvements and advances on algorithm design in the future, it is likely that the secret length of LPN-based schemes will be increased. At present, we choose 512 as the secret length for $d = 80$ in our protocols.

The two secret matrices in HB$^{\#}$ have different secret lengths, as can secret vectors in HB^{+}. This argument comes from remark 1 in [14], which concludes that only the length of the blinding-factor vector matters to security of the LPN problem, whereas the challenge vector is used for mask. Therefore, the parameter set $(\eta = 0.25, k_X = 80, k_Y = 512)$ is sufficient to guarantee 80-bit security for HB^{+} and Random-HB$^{\#}$. This conclusion still holds for our HB-hybrid one-way protocols. However, for mutual authentication protocols, the two secret matrices have to take the same secret length, for example $k = 512$ for 80-bit security. Otherwise an adversary who impersonates either entity and interacts with the other can get the calculation result in a mutual authentication environment.

False Negative and False Positive

Since the HB-family protocols, including our HB-hybrid protocols, are probabilistic approaches, there exist two types of authentication errors. A *false negative*, that is, the authentication of a legitimate entity being rejected, takes place when the number of incorrect responses exceeds the pass-threshold τ. By contrast, a *false positive* is defined that the number of unmatched responses out of random bits is less than the pass-threshold τ. The probabilities of a false negative, P_{FN}, and a false positive, P_{FP}, can be computed by

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

where $n = ml$. P_{FN} and P_{FP} are also referred to as the false negative rate and the false positive rate respectively.

For an authentication protocol to achieve 80-bit security, the false positive rate P_{FP} should be less than 2^{-80}. Considering users' satisfaction, we argue that the false negative rate P_{FN} should be less than 2^{-40}. Notice that the error rates have nothing to do with the secret length k.

Secret-Expansion and Interaction-Expansion

The product of l and m in proposed protocols is determined by the desired false negative rate P_{FN} and false positive rate P_{FP}. Furthermore, the secret-expansion

Table 1. Memory cost and communication cost of the proposed protocols

Protocol	Memory Cost	Communication Cost
HB-hybrid 1	$2kl$	$3mk + 2ml$
HB-hybrid 2	$2k + 2l - 2$	$3mk + 2ml$
HB-hybrid 3	$2kl$	$3m + 3k - 3 + 2ml$
HB-hybrid-OW 1	$kl + dl$	$mk + md + ml$
HB-hybrid-OW 2	$k + d + 2l - 2$	$mk + md + ml$
HB-hybrid-OW 3	$kl + dl$	$2m + k + d - 2 + ml$

Table 2. The range of memory cost (bits) and communication overload (bits) of the proposed protocols, under the conditions: $d = 80$, $P_{\mathrm{FN}} < 2^{-40}$, $P_{\mathrm{FP}} < 2^{-80}$, $k = 512$, and $ml \geq 1090$

Protocol	l		m		Memory		Communication	
	Min	Max	Min	Max	Min	Max	Min	Max
HB-hybrid 1	80	1090	1	14	81920	1116160	3716	23744
HB-hybrid 2	80	1090	1	14	1182	3202	3716	23744
HB-hybrid 3	1	1090	1	1090	1024	1116160	3716	6983
HB-hybrid-OW 1	80	1090	1	14	47360	645280	1682	9408
HB-hybrid-OW 2	80	1090	1	14	750	2770	1682	9408
HB-hybrid-OW 3	1	1090	1	1090	592	645280	1682	3860

l has its own security effect. To thwarts the GRS attack, the Random-HB$^{\#}$ protocol employs $(k \times n)$ matrices. On the other hand, in order to achieve reasonable error rates, n has to be very large, such as 1090, which may lead to significant memory requirement. We observe that the secret-expansion $l = 80$ in the HB-hybrid 1/2 protocols can guarantee 80-bit security, because the probability of GRS attack being successful is 2^{-l} [27]. The bigger l cannot improve the security level as the secret length $k = 512$ limits the security level to be 80-bit. For HB-hybrid 3, $l = 80$ can also guarantee 80-bit resistance against the GRS attack, but the fact that the challenges are encoded by Toeplitz matrices can alleviate the GRS attack too. Therefore we allow l to be smaller than 80 for the HB-hybrid 3 protocol. We will analyze it in detail afterward.

3.3 Performance

The computation cost of our protocols is the same: $4mkl + 4ml$ bit-operations for each participant. The main overload in the proposed protocols is the memory cost and the communication cost. Table 1 shows the formulas of these two metrics of our protocols.

We consider the performance of the HB-hybrid 1, 2, and 3 protocols under the conditions of $d = 80$, $P_{\mathrm{FN}} < 2^{-40}$, and $P_{\mathrm{FP}} < 2^{-80}$. We choose $\eta = 0.25$ and $k = 512$ to satisfy $d = 80$. In order to achieve $P_{\mathrm{FN}} < 2^{-40}$ and $P_{\mathrm{FP}} < 2^{-80}$, we determine the minimum n' such that there exists a valid pass-threshold solution

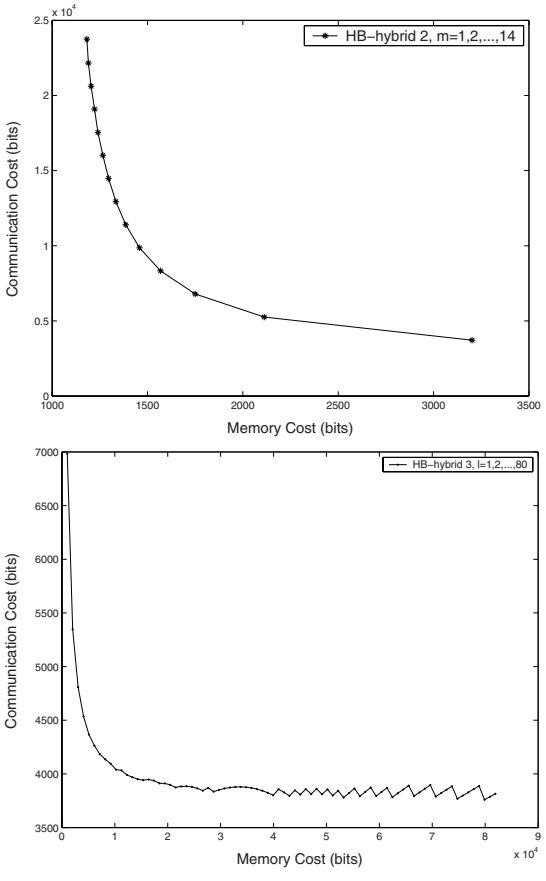

Fig. 2. The trade-off between memory and communication in HB-hybrid 2 and HB-hybrid 3, where $d = 80, k = 512$, and $ml \geq 1090$

τ for all $ml \geq n'$. Interestingly, $(n = 1085, \tau = 375)$ and $(n = 1088, \tau = 376)$ are valid solutions for the designated error rates, but there is no valid τ for $n \in \{1086, 1087, 1089\}$. We can find proper τ for all $n \geq 1090$, therefore $n' = 1090$. The range on the memory requirement and communication overload of the proposed protocols is given in Table 2. The trade-off between memory and communication in HB-hybrid 2 and HB-hybrid 3 is depicted in Fig. 2.

4 Security Analysis

4.1 Threat Model

We consider three kinds of realistic adversaries. First, adversary DET disguises himself as Bob and interacts with Alice, aiming at recovering the secret matrices. Second, adversary GRS manipulates the authentication interaction between

the genuine Alice and Bob, and is allowed to learn the authentication results, intending to recover the secret matrices. Third, adversary RFL claims himself as Bob and interacts with Alice, trying to successfully complete the authentication.

4.2 Security Properties

Property 1. Secret matrices and challenge matrices cannot be Toeplitz matrices simultaneously. Otherwise, adversary DET can recover $\min(k+l-1, 2l-2)$ bits of secret $\mathbb{X}$.

Justification. For adversaries DET and GRS, the main relation among interactions is $\mathbb{Z}_1 = (\mathbb{R}_1 \circ \mathbb{X}) \oplus (\mathbb{R}_2 \circ \mathbb{Y}) \oplus \mathbb{V}$. Since adversary DET is free to choose $\mathbb{R}_2 = \{0\}^{m\times k}$, he can get a challenge-response pair $(\mathbb{R}_1, \mathbb{Z}_1)$ satisfying $\mathbb{Z}_1 = (\mathbb{R}_1 \circ \mathbb{X}) \oplus \mathbb{V}$ in one authentication procedure. Let $\boldsymbol{r}$ be a vector of length $k+m$-1, and let $\boldsymbol{x}$ be a vector of length $l+k$-1. For simplicity, let $\hat{\boldsymbol{r}}(i)$ denote the row vector consisting of k elements from $\boldsymbol{r}[i]$ to $\boldsymbol{r}[i+k\text{-}1]$, where $i = 0, \ldots, m-1$; let $\hat{\boldsymbol{x}}(j)$ denote a column vector consisting of k elements from $\boldsymbol{x}[j]$ to $\boldsymbol{x}[j+k\text{-}1]$, where $j = 0, \ldots, l-1$. Suppose Toeplitz matrices $\mathbb{R}_1$ and $\mathbb{X}$ are encoded by $\boldsymbol{r}$ and $\boldsymbol{x}$ respectively, say, $\mathbb{R}_1$ consisting of m row vectors $\hat{\boldsymbol{r}}(0), \ldots, \hat{\boldsymbol{r}}(m-1)$ and $\mathbb{X}$ consisting of l column vectors $\hat{\boldsymbol{x}}(0), \ldots, \hat{\boldsymbol{x}}(l-1)$ (the encoding method used here is not exactly identical with but essentially equivalent to that in the Toeplitz matrix). Then each element in the response matrix $\mathbb{Z}_1$ can be computed by

$$\mathbb{Z}_1[i,j] = (\hat{\boldsymbol{r}}(i) \circ \hat{\boldsymbol{x}}(j)) \oplus \mathbb{V}[i,j] \ . \tag{2}$$

Therefore, for each element pair $(\mathbb{Z}_1[i,j], \mathbb{Z}_1[i+1,j+1])$, where $i \in \{0, \cdots, m-2\}$ and $j \in \{0, \cdots, l-2\}$, the following relation holds:

$$\begin{aligned}\mathbb{Z}_1[i,j] \oplus \mathbb{Z}_1[i+1,j+1] = (\boldsymbol{r}[i] \cdot \boldsymbol{x}[j]) \oplus (\boldsymbol{r}[i+k] \cdot \boldsymbol{x}[j+k]) \\ \oplus \mathbb{V}[i,j] \oplus \mathbb{V}[i+1,j+1] \ .\end{aligned}$$

Adversary DET collects a great number of challenge-response pairs $(\mathbb{R}_1, \mathbb{Z}_1)$, choosing two sets based on $(\boldsymbol{r}[i] = 1$ and $\boldsymbol{r}[i+k] = 0)$ or $(\boldsymbol{r}[i] = 0$ and $\boldsymbol{r}[i+k] = 1)$. Then adversary DET, according to the equation above, can overcome noises and recover secret $\boldsymbol{x}[j]$ and $\boldsymbol{x}[j+k]$ with overwhelming probability, where $j \in \{0, \cdots, l-2\}$. Therefore, the head and tail portion of secret $\mathbb{X}$ ($\boldsymbol{x}[0], \cdots, \boldsymbol{x}[l\text{-}2]$ and $\boldsymbol{x}[k], \cdots, \boldsymbol{x}[k+l\text{-}2]$) are compromised.

Property 2. If only one class of matrices in the secrets and challenges are Toeplitz matrices, then it does not provide any non-negligible advantage to adversary DET. Therefore, HB-hybrid 2 and HB-hybrid 3 are as secure as HB-hybrid 1 against adversary DET.

Justification. The HB-hybrid 1 protocol is provably secure under adversary DET, combining the proofs in [16,17,27]. In the HB-hybrid framework, adversary DET, at his best, can get a challenge-response pair $(\mathbb{R}_1, \mathbb{Z}_1)$ satisfying $\mathbb{Z}_1 = (\mathbb{R}_1 \circ \mathbb{X}) \oplus \mathbb{V}$ in one authentication procedure. If only secrets are Toeplitz matrices, suppose $\mathbb{X}$ is encoded by vector $\boldsymbol{x}$ of length $l+k$-1, then each bit $\boldsymbol{x}$[j] is multiplied

with each element in random matrix $\mathbb{R}_1$ at most once under $\mathbb{Z}_1 = (\mathbb{R}_1 \circ \mathbb{X}) \oplus \mathbb{V}$. Consequently, the Toeplitz matrix cannot be distinguished with a random matrix for adversary DET, and it would not give adversary DET, who tries to learn the correlation between challenge-response, any non-negligible advantage for recovering the secrets. The same argument can be applied to the Toeplitz challenge matrices.

Property 3. If the secrets are random matrices, encoding challenges by Toeplitz matrices can improve resilience against adversary GRS, in addition to reducing communication cost.

Justification. We demonstrate it by giving an example of boundary case $l = 1$, the secret matrix $\mathbb{X}$ degenerates to vector $\boldsymbol{x}$. In this case, the HB-hybrid 1 protocol will be vulnerable to the GRS attack, and adversary GRS can fully recover secrets. In contrast, the direct GRS attack has little impact on the HB-hybrid 3 protocol. For the direct GRS attack to work fully functionally, adversary GRS is required to have the ability to exclusive-or the same $\boldsymbol{\delta}$ with all row vectors in a challenge matrix. Since adversary GRS cannot freely manipulate the Toeplitz challenge this way, the direct GRS attack fails with one exception $\boldsymbol{\delta} = \{1\}^{m+k-1}$. However, we notice that an extended GRS attack based on probability statistics can be applied to the case of Toeplitz challenge matrices and would give adversary GRS some advantage. This attack is described as follows.

(1) In the HB-family protocols, including HB-hybrid, the successful authentication probability, when there are q reverse bits in the binary response matrix $\mathbb{Z}_1$, can be computed by

$$P_q = \sum_{i=0}^{\tau} \sum_{j=0}^{i} \binom{n-q}{j} \binom{q}{i-j} \eta^{q-i+2j} (1-\eta)^{n-q+i-2j} \quad , \tag{3}$$

where $n = ml$ for HB-hybrid.

Adversary GRS generates the probability table for all $q \in \{0, 1, \cdots, n\}$. For $n = 1090$ and $\tau = 377$, this probability under $q \leq \frac{n}{3}$ is distinguishable with 2^{30} samples.

(2) Adversary GRS selects $\boldsymbol{\delta}$ with a certain period, such as $2, 3, \ldots$, repeats using it to interfere with the challenge, and observes the overall successful probability $P_{\boldsymbol{\delta}}$ of manipulated authentications.

(3) Adversary GRS locates $P_{\boldsymbol{\delta}}$ to the closest item in the probability table, and tries to obtain some useful information about secret $\boldsymbol{x}$.

For instance, adversary GRS chooses $\boldsymbol{\delta} = \widetilde{1001}$ of period 4. This $\boldsymbol{\delta}$ works like four disturbing vectors $\boldsymbol{\delta}_1 = \widetilde{1001}, \boldsymbol{\delta}_2 = \widetilde{0011}, \boldsymbol{\delta}_3 = \widetilde{0110}$ and $\boldsymbol{\delta}_4 = \widetilde{1100}$ under the Toeplitz challenge. Therefore, the probability of successful authentication will reveal if all $\boldsymbol{\delta}_i \circ \boldsymbol{x}$ is equal to 0 or if only one of $\boldsymbol{\delta}_i \circ \boldsymbol{x}$ is equal to 1, but cannot distinguish other cases. Anyway, it still shows that the Toeplitz challenges are better than the random challenges in terms of resilience again GRS attack.

Property 4. The role of $\mathbb{R}_3$ cannot be replaced by $\mathbb{R}_1$. Otherwise, adversary RFL can succeed in authenticating himself as Bob.

Justification. If a participant is required to conduct only one authentication at a time, under no circumstances should adversary RFL succeed. If we substitute the role of $\mathbb{R}_3$ by $\mathbb{R}_1$ for saving communication cost, the negative result happens. After receiving $\mathbb{R}_1$ from Alice, adversary RFL responds with $\mathbb{R}_2 = \mathbb{R}_1$. After getting $\mathbb{Z}_1$, adversary RFL answers $\mathbb{Z}_2 = \mathbb{Z}_1$. Then Alice would accept adversary RFL's authentication.

Property 5. If an entity is allowed to conduct many authentication procedures at a time, then the entity should take one constant role as Alice or Bob in all the procedures. Otherwise, adversary RFL can successfully impersonate other entities.

Justification. It is noticed that Alice and Bob are different roles in our mutual authentication protocols. Suppose an entity C initializes an authentication procedure with adversary RFL who claims himself to be entity D. After receiving the challenge $\mathbb{R}_1$, adversary RFL starts other authentication procedure with entity C, and sends $\mathbb{R}_1$ back to entity C as his first challenge. Adversary RFL acts as a receiving-forwarding transfer in the two authentication procedures; then finally he can pass the verification of entity C and successfully impersonate entity D. If we oblige one entity to take one role as Alice or Bob at a time, we can withstand this reflection attack. In this regard, our mutual entity authentication is better than a simple combination of two independent one-way authentications in terms of security, in addition to communication saving.

5 Application Scenarios

Energy is the most valuable resource for wireless sensor networks. The main energy consumption in our protocols lies on challenge-response transmission. Therefore, we should keep the communication cost as low as possible. In this regard, the HB-hybrid 3 protocol is a good candidate for wireless sensor networks. For example, the parameter set $(k = 512, l = 80, m = 14)$ in HB-hybrid 3 will provide 80-bit security, require 10 KB storage for secrets, with the communication cost of 3815 bits for one mutual authentication procedure.

We define the secret matrices $\mathbb{X}$ and $\mathbb{Y}$ as an authentication key. According to different applications, there are four scenarios for authentication key predistribution.

Scenario 1: Single common key shared by all nodes
This is the simplest but very useful scenario for sensor networks. Upon deployment, every node explores its adjacent nodes and trusts all neighbors which are discovered in a short time T_{min}. We assume that the adversary cannot launch any attack during the time slot T_{min}. This initial-trust model has been addressed in many WSN security proposals, such as initial-trust in [29] and smart trust in

[30]. Due to the random employment characteristic of wireless sensor networks, this mode is relatively practical in some applications. After time T_{min}, any new node that wants to join the network has to authenticate itself to other previous nodes. Any user who wants to issue task commands to the sensor network also should authenticate himself to sensor nodes first. The nodes within the trusted network always take the role of Bob in the mutual authentication, while new nodes and users should act as Alice. According to Property 5, adversary RFL cannot pass authentication through receiving-forwarding transfer in two authentication procedures with different nodes. The main disadvantage of this case is that an adversary who captures any node and extracts the global authentication key will compromise the security of authentication. Tamper-proof memory for the global key is a solution to this attack.

Scenario 2: Distinct key that each node shares with base stations
This scenario is useful for authentication of users who want to query individual sensor's data. As a matter of fact, many proposed entity authentication schemes in wireless sensor networks mainly consider this scenario. For example, a nurse tries to query patients' physical information from body sensor networks using handset devices, which are defined as users of the sensor networks. The users, usually being high-capability devices with enough memory to hold all keys with nodes, move around to issue task commands to the sensor networks and collect information. The authentication of users guarantees the patients' privacy, and the authentication of sensor nodes makes sure that users retrieve data from legitimate nodes. This scenario can be combined with the previous one to provide overall authentication for the whole sensor network.

Scenario 3: Pairwise key in every two nodes
If the initial-trust assumption does not hold in some applications and the tamper-proof memory is impractical for sensor nodes, we may ask every two nodes to be preloaded with a distinct pairwise key. The storage requirement in a single node is proportional to the number of all nodes in a network. Consequently, this approach is only suitable for small scale sensor networks.

Scenario 4: Random predistribution keys in nodes
Random key predistribution approaches [31] are one of the most prevalent techniques for key establishment in sensor networks. This technique can be directly applied to authentication key distribution between nodes, in order to increase resilience against physically compromising sensor nodes. Interestingly, we can only count on the basic EG protocol [31] since the motivation of our proposal is to design bit-operation-based-only approaches. In this application scenario, an offline server first generates an authentication key pool of a large number of keys. Then every node is randomly preloaded with some authentication keys out of the key pool before deployment. If two adjacent nodes happen to share at least one common authentication key, they can mutually authenticate each other directly. If they do not, they can rely on other nodes to facilitate their authentication.

According to random graph theory [31], two nodes will almost be able to find an authentication path through multi-hop links if any two nodes share at least one common authentication key with certain probability. Then these two nodes, finding this authentication path with overwhelming probability, authenticate each other via one-by-one authentication relay in the path.

6 Conclusion and Further Work

In this paper, we describe the HB-hybrid mutual authentication framework, and give the extensive performance evaluation and security analysis for our protocols. This kind of computationally efficient approaches are quite appreciated in resource-restrained sensor networks. The mutual entity authentication is the first step in this research direction. If we can design effective and relatively efficient, bit-operation-based message authentication protocols, the combination of effective identification and message authentication would greatly increase security applicability in such application areas. On the other hand, our protocols still suffer from some reflection attacks and the most effective way to resist the reflection attacks is to embed entities' ID into challenges. How to achieve this objective securely will be included in our future research.

Acknowledgment

The research is supported by NSERC Strategic Project Grants.

References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A survey on sensor networks. IEEE Communications Magazine 40(8), 102–114 (2002)
2. Benenson, Z., Gedicke, N., Raivio, O.: Realizing robust user authentication in sensor networks. In: Real-World Wireless Sensor Networks, REALWSN (2005)
3. Jiang, C., Li, B., Xu, H.: An Efficient Scheme for User Authentication in Wireless Sensor Networks. In: 21st International Conference on Advanced Information Networking and Applications Workshops, pp. 438–442 (2007)
4. Wong, K.H., Zheng, Y., Cao, J., Wang, S.: A Dynamic User Authentication Scheme for Wireless Sensor Networks. In: IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC 2006), pp. 244–251 (2006)
5. Tseng, H.R., Jan, R.H., Yang, W.: An Improved Dynamic User Authentication Scheme for Wireless Sensor Networks. In: IEEE Global Telecommunications Conference (GLOBECOM 2007), pp. 986–990 (2007)
6. Tripathy, S., Nandi, S.: Defense against outside attacks in wireless sensor networks. Computer Communications 31(4), 818–826 (2008)
7. Crawford, J.M., Kearns, M.J.: The Minimal Disagreement Parity Problem as a Hard Satisfiability Problem. Computational Intelligence Research Laboratory and AT&T Bell Labs (1995), `http://www.cs.cornell.edu/selman/docs/crawford-parity.pdf`

8. Berlekamp, E.R., McEliece, R.J., van Tilborg, H.C.: On the inherent intractability of certain coding problems. IEEE Transactions on Information Theory 24(3), 384–386 (1978)
9. MacWilliams, F., Sloane, N.: The Theory of Error-Correcting Codes. North-Holland, Amsterdam (1977)
10. Håstad, J.: Some optimal inapproximability results. In: Proceedings of the twenty-ninth annual ACM symposium on Theory of computing, El Paso, Texas, United States (1997)
11. Hopper, N., Blum, M.: Secure Human Identification Protocols. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 52–66. Springer, Heidelberg (2001)
12. Blum, A., Kalai, A., Wasserman, H.: Noise-tolerant learning, the parity problem, and the statistical query model. Journal of the ACM (J. ACM) 50(4), 506–519 (2003)
13. Fossorier, M.P.C., Mihaljević, M.J., Imai, H., Cui, Y., Matsuura, K.: An Algorithm for Solving the LPN Problem and Its Application to Security Evaluation of the HB Protocols for RFID Authentication. In: Barua, R., Lange, T. (eds.) INDOCRYPT 2006. LNCS, vol. 4329, pp. 48–62. Springer, Heidelberg (2006)
14. Levieil, E., Fouque, P.A.: An Improved LPN Algorithm. In: De Prisco, R., Yung, M. (eds.) SCN 2006. LNCS, vol. 4116, pp. 348–359. Springer, Heidelberg (2006)
15. 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), http://www.rsasecurity.com/rsalabs/staff/bios/ajuels/publications/pdfs/lpn.pdf
16. Katz, J., Shin, J.: 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)
17. Katz, J., Smith, A.: Analyzing the HB and HB$^+$ Protocols in the "Large Error" Case. Technical report, Cryptology ePrint Archive, Report 2006/326 (2006)
18. Gilbert, H., Robshaw, M., Sibert, H.: An Active Attack Against HB$^+$ - A Provably Secure Lightweight Authentication Protocol. Technical report, Cryptology ePrint Archive: Report 2005/237 (2005)
19. Bringer, J., Chabanne, H., Dottax, E.: HB++: a Lightweight Authentication Protocol Secure against Some Attacks. In: Chabanne, H. (ed.) Second International Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing (SecPerU 2006), pp. 28–33 (2006)
20. Duc, D.N., Kim, K.: Securing HB+ Against GRS Man-in-the-Middle Attack. In: Proceedings of Symposium on Cryptography and Information Security (SCIS 2007), Sasebo, Japan (2007)
21. Munilla, J., Peinado, A.: HB-MP: A further step in the HB-family of lightweight authentication protocols. Comput. Networks 51(9), 2262–2267 (2007)
22. Gilbert, H., Robshaw, M.J., 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)
23. Piramuthu, S.: HB and Related Lightweight Authentication Protocols for Secure RFID Tag/Reader Authentication. In: CollECTeR Europe Conference (2006)
24. Leng, X., Mayes, K., Markantonakis, K.: HB-MP+ Protocol: An Improvement on the HB-MP Protocol. In: IEEE International Conference on RFID, pp. 118–124 (2008)
25. Hammouri, G., Sunar, B.: PUF-HB: A Tamper-Resilient HB Based Authentication Protocol. In: Bellovin, S.M., Gennaro, R., Keromytis, A.D., Yung, M. (eds.) ACNS 2008. LNCS, vol. 5037, pp. 346–365. Springer, Heidelberg (2008)

26. Bringer, J., Chabanne, H.: Trusted-HB: A Low-Cost Version of HB+ Secure Against Man-in-the-Middle Attacks. IEEE Transactions on Information Theory 54(9), 4339–4342 (2008)
27. Gilbert, H., Robshaw, M.J., Seurin, Y.: HB$^{\#}$: Increasing the Security and Efficiency of HB^{+}. In: Advances in Cryptology EUROCRYPTO 2008 (2008), Full version available at: Cryptology ePrint Archive: Report 2008/028 (2008)
28. Carrijo, J., Tonicelli, R., Imai, H., Nascimento, A.C.A.: A Novel Probabilistic Passive Attack on the Protocols HB and HB^{+}. Technical report, Cryptology ePrint Archive: Report 2008/231 (2008)
29. Zhu, S., Setia, S., Jajodia, S.: LEAP: efficient security mechanisms for large-scale distributed sensor networks. In: Proceedings of the 10th ACM conference on Computer and Communication Security (CCS 2003), Washington, DC, pp. 62–72. ACM, New York (2003)
30. Anderson, R., Chan, H., Perrig, A.: Key Infection: Smart Trust for Smart Dust. In: Proceedings of the 12th IEEE International Conference on Network Protocols (ICNP 2004), pp. 206–215 (2004)
31. Eschenauer, L., Gligor, V.D.: A key-management scheme for distributed sensor networks. In: Proceedings of the 9th ACM conference on Computer and Communications Security, Washington, DC, USA, pp. 41–47 (2002)