hHB: a Harder HB$^+$ Protocol

Ka Ahmad Khoureich*

Abstract

In 2005, Juels and Weis proposed HB$^+$, a perfectly adapted authentication protocol for resource-constrained devices such as RFID tags. The HB$^+$ protocol is based on the *Learning Parity with Noise* (LPN) problem and is proven secure against active adversaries. Since a man-in-the-middle attack on HB$^+$ due to Gilbert et *al.* was published, many proposals have been made to improve the HB$^+$ protocol. But none of these was formally proven secure against general man-in-the-middle adversaries.

In this paper we present a solution to make the HB$^+$ protocol resistant to general man-in-the-middle adversaries without exceeding the computational and storage capabilities of the RFID tag.

Keywords. RFID, Authentication, LPN, HB$^+$, Man-In-the-Middle.

1 Introduction

Radio-frequency identification (RFID) belongs to the family of Automatic Identification systems. RFID system consists of tags and readers that communicate wirelessly. The RFID tag attached to an object can be used for access control, product traking, identification, etc. Since the tag is programmable, a malicious person can then create counterfeit tags and benefit from it. Hence the need to secure the protocol run between the tag and the reader.

RFID tags have a low computational and storage capacity. Therefore, it is impossible to use classical cryptographic algorithms to secure the protocol they execute. At Crypto 2005, Juels and Weis proposed HB$^+$ [13], a perfectly adapted authentication protocol for resource-constrained devices such as RFID tags. The protocol consists of a number of rounds of challenge-response authentication. HB$^+$ is based on the *Learning Parity with Noise* (LPN) problem — which is known to be NP-Hard — and is proven secure against active adversaries [13,14]. Since a simple man-in-the-middle attack on HB$^+$ due to Gilbert et *al* [9]. was published, many proposals [4–6,16,18] have been made to improve the HB$^+$ protocol. But none of these was formally proven secure against general man-in-the-middle adversaries [8,10,19].

In this paper we present a solution to make HB$^+$ resistant to general man-in-the-middle adversaries without exceeding the computational and storage capabilities of the RFID tag.

Our paper is organized as follow: (1) we give a definition of the LPN problem, (2) we describe the HB$^+$ protocol, (3) we present our protocol based on HB$^+$ and provide security proofs, (4) we conclude with some observations and future work.

2 The LPN Problem

The LPN problem is a very known one [1–3,11,12,15,20]. Let $\mathsf{hw}(v)$ denote Hamming weight of a binary vector $\boldsymbol{v}$.

*Dept. of Computer Science, Alioune Diop University of Bambey, Senegal. ahmadkhoureich.ka@uadb.edu.sn

Definition 2.1. *Let A be a random $q \times k$ binary vector matrix, let $\boldsymbol{x}$ be a random k-bit vector, let $\boldsymbol{\varepsilon} \in]0, \frac{1}{2}[$ be a constant noise parameter, and let $\boldsymbol{\nu}$ be a random q-bit vector such that $\mathsf{hw}(\nu) < \varepsilon q$. Given A, $\boldsymbol{\varepsilon}$, and $z = (A \cdot x) \oplus \nu$, find a k-bit vector $\boldsymbol{x'}$ such that $\mathsf{hw}(A \cdot x' \oplus z) \leq \varepsilon q$.*

The difficulty of finding $\boldsymbol{x}$ (solving the LPN) comes from the fact that each bit of $A \cdot x$ is flipped independantly with probability $\boldsymbol{\varepsilon}$, thus making hard to get a system of linear correct equations in $\boldsymbol{x}$ which can be easily solved using the Gaussian elimination.

Let Ber_ε denote the Bernoulli distribution with parameter $\boldsymbol{\varepsilon}$, (i.e. $\nu \leftarrow \mathsf{Ber}_\varepsilon$, $\Pr[\nu = 1] = 1 - \Pr[\nu = 0] = \varepsilon$) and let $A_{x,\varepsilon}$ be the distribution define by $\{a \leftarrow \{0,1\}^k; \nu \leftarrow \mathsf{Ber}_\varepsilon : (a, a \cdot x \oplus \nu)\}$. One consequence of the hardness of the LPN with noise parameter $\boldsymbol{\varepsilon}$ is that $A_{x,\varepsilon}$ is indistinguishable from the uniform distribution U_{k+1} on $(k+1)$-bit strings; see [14].

Although many algorithms solving the LPN problem has been published [3,7,17], the current most efficient one due to Blum, Kalai, and Wasserman [3] has a runtime of $2^{O(\frac{k}{\log k})}$.

3 The HB$^+$ Protocol

HB$^+$ is an authentication protocol based on the LPN problem and designed for low-cost devices like RFID tags. The protocol consists of $r = r(k)$ challenge-response authentication rounds between the reader and the tag who share two random secrets keys $\boldsymbol{x}$ and $\boldsymbol{y}$ of length k. A round consists of the following steps (see fig 1 for a graphical representation):

1. the tag randomly chooses and sends a vector $b \leftarrow \{0,1\}^k$ called "blinding factor" to the reader,
2. the reader randomly choose and sends $a \leftarrow \{0,1\}^k$ a challenge vector to the tag,
3. the tag gets a bit ν following Ber_ε and responses to the reader by sending a bit $z = a \cdot x \oplus b \cdot y \oplus \nu$,
4. the reader accepts the authentication round if $a \cdot x \oplus b \cdot y = z$.

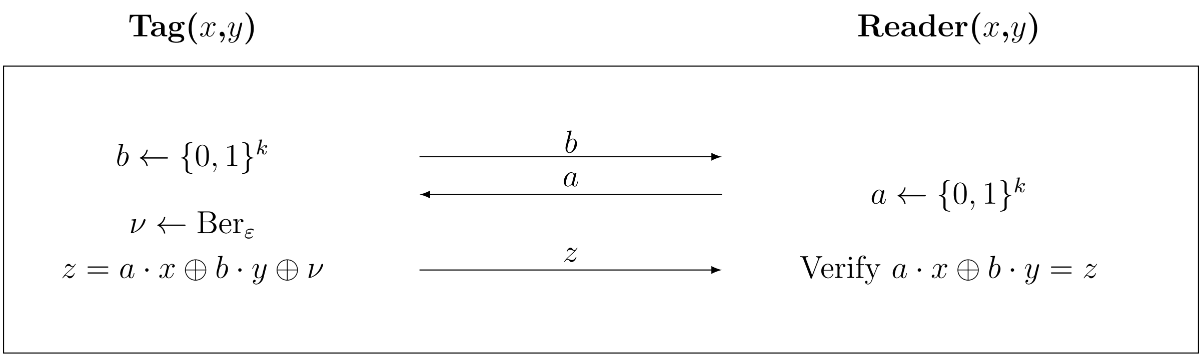

Figure 1: A round of the HB$^+$ Protocol.

The parameters of HB$^+$ are: the shared secrets $\boldsymbol{x}$ and $\boldsymbol{y}$ each of lenght k, the number of rounds $r = r(k)$, the Bernoulli parameter $\boldsymbol{\varepsilon}$ and the threshold $\mathsf{u} = \mathsf{u}(k)$. The threshold u is such that it is greater than $\varepsilon \cdot r$ so the reader accepts the tag if the number of rounds for which Verify $a \cdot x \oplus b \cdot y = z$ returns false is less than u. Because of $\boldsymbol{\nu}$ in the response $\boldsymbol{z}$ of the tag, the probability that an authentication round be unsuccessful even for the honest tag is not null. Therefore the event called false rejection that the reader rejects a honest tag happens with probability

$$P_{FR} = \sum_{i=\mathsf{u}+1}^{r} \binom{r}{i} \varepsilon^i (1-\varepsilon)^{r-i}.$$

At the same time an adversary sending random responses $\boldsymbol{z}$ to the reader can be accepted with probability

$$P_{FA} = \frac{1}{2^r} \sum_{i=0}^{u} \binom{r}{i}.$$

This event is called false acceptance. Fortunately these probabilities (P_{FR} and P_{FA}) are negligible in k because $r = r(k)$ (the use of Chernoff bound helps to see it).

3.1 Attacks on HB$^+$

HB$^+$ is in fact an improvment of an earlier protocol named HB [12] which is secure against passive attack but vulnerable to active ones. In an active attack the adversary plays the role of a reader and tries to get the secrets from a honest tag. HB$^+$ is proven secure against this type of attack [13, 14] but is defenceless against more powerful adversaries like man-in-the-middle (MIM). In such attacks the adversary stays between the reader and the tag and have the abilities to tamper with messages.

In [9] Gilbert, Robshaw, and Silbert present a MIM-attack against HB$^+$ called GRS attack. The attack is depicted in fig 2. In the GRS attack, in order to reveal the secret $\boldsymbol{x}$, the adversary does not need to modify all the messages exchanged between the tag and the reader but only the challenge vector $\boldsymbol{a}$. The adversary adds a perturbation δ on the challenge vector $\boldsymbol{a}$ and looks if the whole authentication process will be successful or not. The reader will verify if $a' \cdot x \oplus b \cdot y = z$ that is if $\delta \cdot x \oplus \nu = 0$. If the honest tag continues to be authenticated normaly with negligible fails (P_{FR}) then the whole authentication process is not disturbed and it means that $\delta \cdot x = 0$ otherwise $\delta \cdot x = 1$. By using $\delta = e_i$ the vector with 1 at position i and 0s elsewhere, the adversary gets the bit x_i of x. By repeating the attack k times with different δ the adversary gets the whole secret $\boldsymbol{x}$.

Tag(x,y)		**Reader**(x,y)
$b \leftarrow \{0,1\}^k$	$\xrightarrow{\quad b \quad}$	
	$\xleftarrow{a' = a \oplus \delta} \quad \xleftarrow{\; a \;}$	$a \leftarrow \{0,1\}^k$
$\nu \leftarrow \mathrm{Ber}_\varepsilon$		
$z = a' \cdot x \oplus b \cdot y \oplus \nu$	$\xrightarrow{\quad z \quad}$	Verify $a \cdot x \oplus b \cdot y = z$

Figure 2: The GRS attack. The adversary adds a perturbation on the challenge vector $\boldsymbol{a}$ and looks if the whole authentication process will be disturbed or not.

Much work [4–6, 16, 18] has been done in order to propose a protocol based on the LPN problem and resistant to the GRS attack. But none of these has been formally proven secure against general man-in-the-middle attacks [8, 10, 19].

4 Our proposal

Intuitively we believe that the weakness of HB$^+$ to the man-in-the-middle attack is due to the fact that the secret $\boldsymbol{x}$ does not change. This intuition is reinforced by our observation of RANDOM-HB$^{\#}$ — partially resistant to this type of attack (GRS attack) — which can be viewed as an HB$^+$ protocol where the secret $\boldsymbol{x}$ varies in each round (although in fact parallel) but remains fixed for each instance of the protocol.

The main idea is to let the reader choose a random k-bit secret $\boldsymbol{x}$ and then sends it to the tag in a secure way. Our protocol denoted hHB for harder HB consists of two stages. The first stage is the random selection of the secret $\boldsymbol{x}$ and its transmission to the tag and the second stage is identical to HB$^+$. The secret $\boldsymbol{x}$ is transmitted bit by bit from the reader to the tag. The reader randomly selects three bits (τ, ξ_0, ξ_1) and sets the value x_i (a bit of x) to ξ_τ. After that the order of the three bits is randomly changed by a function f_s (see Algorithm 1 and 2) and securely communicated to the tag using the shared secret $\boldsymbol{s}$. This operation is repeated $|x|$ times. The hHB protocol is outlined in figure 3. The second stage of hHB witch is identical to a round of HB$^+$ is run r times. An authentication round is successful if $\mathsf{Verify}\ a \cdot x \oplus b \cdot y = z$ returns true. The reader accepts the tag if the number of unsuccessful rounds is less than a threshold u.

Tag(s,y)		**Reader**(s,y)
		$\tau \leftarrow \{0,1\}\ \xi_0 \leftarrow \{0,1\}\ \xi_1 \leftarrow \{0,1\}$
		$x_i = \xi_\tau$
	$\xleftarrow{(\alpha,\beta,\gamma)}$	$(\alpha,\beta,\gamma) = f_s(\tau,\xi_0,\xi_1)$
$(\tau,\xi_0,\xi_1) = f_s^{-1}(\alpha,\beta,\gamma)$		
$x_i = \xi_\tau$		
$x = x_1x_2 \ldots x_k$		$x = x_1x_2 \ldots x_k$
$b \leftarrow \{0,1\}^k$	$\xrightarrow{b}$	
	$\xleftarrow{a}$	$a \leftarrow \{0,1\}^k$
$\nu \leftarrow \mathrm{Ber}_\varepsilon$		
$z = a \cdot x \oplus b \cdot y \oplus \nu$	$\xrightarrow{z}$	$\mathsf{Verify}\ a \cdot x \oplus b \cdot y = z$

Figure 3: The hHB authentication protocol. The first stage is run for $i = 1$ to $|x|$ to obtain and share the secret $\boldsymbol{x}$. The second stage is identical to HB$^+$.

5 Security Proofs

5.1 Notation and Security definitions

We call negl any negligible function, that is which tends to zero faster than any inverse polynomial. That is, for any polynomial $p(\cdot)$ there exist an N such that for all integer n greater than N we have $\mathsf{negl}(n) < \frac{1}{p(n)}$.

The parameters of hHB are: the shared secrets $\boldsymbol{s}$ and $\boldsymbol{y}$ each of lenght k, the number of rounds $r = r(k)$ of its second part, the Bernoulli parameter $\boldsymbol{\varepsilon}$ and the threshold $\mathsf{u} = \mathsf{u}(k)$. The parameters $\boldsymbol{\varepsilon}$, $\boldsymbol{r}$ and u are the same as for the HB$^+$ protocol.

Let $\mathcal{T}_{s,y,\varepsilon,r}$ and $\mathcal{R}_{s,y,\varepsilon,\mathsf{u},r}$ denote the algorithms respectively run by the honest tag and the honest reader in the hHB protocol. Let k denote the security parameter. An active attack is by definition performed in two stages: first the adversary interacts $q(k)$ times with the tag, second she tries to authenticate to the reader. Man-in-the-middle attacks requires more power

Algorithm 1 Function f_s that changes the order of elements in a triplet $(\lambda_1,\ \lambda_2,\ \lambda_3)$

function $f_s(\lambda_1,\ \lambda_2,\ \lambda_3)$
  $c_1 \leftarrow \{0,1\}^k \quad t_1 = c_1 \cdot s \oplus \lambda_1$
  $c_2 \leftarrow \{0,1\}^k \quad t_2 = c_2 \cdot s \oplus \lambda_2$
  $c_3 \leftarrow \{0,1\}^k \quad t_3 = c_3 \cdot s \oplus \lambda_3$

  if $\lambda_1 \wedge \lambda_2 \wedge \lambda_3 = \lambda_1 \vee \lambda_2 \vee \lambda_3$ **then**
    return $((c_1, t_1), (c_2, t_2), (c_3, t_3))$
  end if
  if $\lambda_1 \oplus \lambda_2 \oplus \lambda_3 = 0$ **then**
    return $((c_3, t_3), (c_1, t_1), (c_2, t_2))$
  end if
  if $\lambda_1 \oplus \lambda_2 \oplus \lambda_3 = 1$ **then**
    return $((c_2, t_2), (c_3, t_3), (c_1, t_1))$
  end if
end function

than active attacks. There the adversary can tamper with all messages going from the reader to the tag and vice versa for $q(k)$ executions of the protocol, and after that tries to authenticate to the reader. The adversary's advantage according to the model of attack can be defined as follow

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{active}}(\varepsilon, \mathsf{u}, \mathsf{r}) \stackrel{\text{def}}{=} \mathsf{Pr}\Big[s \leftarrow \{0,1\}^k; y \leftarrow \{0,1\}^k; \mathcal{A}^{\mathcal{T}_{s,y,\varepsilon,r}}(1^k) : \big\langle \mathcal{A}, \mathcal{R}_{s,y,\varepsilon,\mathsf{u},r} \big\rangle = \mathsf{accept}\Big],$$
$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{mim}}(\varepsilon, \mathsf{u}, \mathsf{r}) \stackrel{\text{def}}{=} \mathsf{Pr}\Big[s \leftarrow \{0,1\}^k; y \leftarrow \{0,1\}^k; \mathcal{A}^{\mathcal{T}_{s,y,\varepsilon,r}, \mathcal{R}_{s,y,\varepsilon,\mathsf{u},r}}(1^k) : \big\langle \mathcal{A}, \mathcal{R}_{s,y,\varepsilon,\mathsf{u},r} \big\rangle = \mathsf{accept}\Big],$$

where $\big\langle \mathcal{A}, \mathcal{R}_{s,y,\varepsilon,\mathsf{u},r} \big\rangle$ denote an attempt of $\mathcal{A}$ to authenticate to the reader.

Algorithm 2 Function f_s^{-1}

function $f_s^{-1}((c_1, t_1), (c_2, t_2), (c_3, t_3))$
  $\lambda_1 = c_1 \cdot s \oplus t_1$
  $\lambda_2 = c_2 \cdot s \oplus t_2$
  $\lambda_3 = c_3 \cdot s \oplus t_3$

  if $\lambda_1 \wedge \lambda_2 \wedge \lambda_3 = \lambda_1 \vee \lambda_2 \vee \lambda_3$ **then**
    return $(\lambda_1, \lambda_2, \lambda_3)$
  end if
  if $\lambda_1 \oplus \lambda_2 \oplus \lambda_3 = 0$ **then**
    return $(\lambda_2, \lambda_3, \lambda_1)$
  end if
  if $\lambda_1 \oplus \lambda_2 \oplus \lambda_3 = 1$ **then**
    return $(\lambda_3, \lambda_1, \lambda_2)$
  end if
end function

5.2 Security of the hHB Protocol against Active Attacks

Theorem 5.1. *If HB$^+$ with parameters $0 < \varepsilon < \frac{1}{2}$, $r = r(k)$ and $\mathsf{u} > \varepsilon \cdot r$ is secure against active attacks then hHB with the same settings of parameters is secure against active attacks.*

Proof. Let $\mathcal{A}$ be a probabilistic polynomial-time adversary interacting with the tag in at most q executions of hHB protocol and achieving $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{active}}(\varepsilon, \mathsf{u}, \mathsf{r}) = \delta$.

We construct a probabilistic polynomial-time adversary $\mathcal{A}'$ who performs an active attacks on HB$^+$ and uses $\mathcal{A}$ as a sub-routine. For the first phase of the attack, $\mathcal{A}'$ simulates for $\mathcal{A}$ the hHB tag for q times as follows:

1. $\mathcal{A}'$ receives the triplets $(\alpha_i, \beta_i, \gamma_i)$ for $i = 1..k$ sent by $\mathcal{A}$.
2. $\mathcal{A}'$ forwards $\boldsymbol{b}$ sent by the honest HB$^+$ tag to $\mathcal{A}$,
3. $\mathcal{A}$ replies to $\mathcal{A}'$ by sending a challenge vector $\boldsymbol{a}$ which is then fowarded by $\mathcal{A}'$ to the honest HB$^+$ tag,
4. $\mathcal{A}'$ forwards z sent by the honest tag HB$^+$ to $\mathcal{A}$,

Steps 2., 3. and 4. are run r times. For the second phase of the attack, $\mathcal{A}'$ simulates for $\mathcal{A}$ the hHB reader as follows:

5. $\mathcal{A}'$ generates k triplets $(\alpha_i, \beta_i, \gamma_i)$ and sends it to $\mathcal{A}$,
6. $\mathcal{A}$ sends $\boldsymbol{b}$ to $\mathcal{A}'$ which it forwards to the honest HB$^+$ reader,
7. $\mathcal{A}'$ sends to $\mathcal{A}$ the challenge vector $\boldsymbol{a}$ which it received from the honest HB$^+$ reader,
8. $\mathcal{A}$ sends z to $\mathcal{A}'$ which it forwards to the honest HB$^+$ reader,

Steps 6., 7. and 8. are run r times. It is not difficult to see that the view of $\mathcal{A}$ when run as a sub-routine by $\mathcal{A}'$ is distributed identically to the view of $\mathcal{A}$ when performing an active attack on hHB (Because even if $\mathcal{A}$ has carefully chosen the triplets $(\alpha_i, \beta_i, \gamma_i)$ it sent in step 1, the blinding vector b prevents it to distinguish the effects of its choises in the value of z). So,

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{active}}(\varepsilon, \mathsf{u}, \mathsf{r}) = \delta = \mathsf{Adv}_{\mathcal{A}', \mathsf{HB}^+}^{\mathsf{active}}(\varepsilon, \mathsf{u}, \mathsf{r}).$$

Because HB$^+$ is secure against active attack, there is a negligible function negl such that

$$\mathsf{Adv}_{\mathcal{A}', \mathsf{HB}^+}^{\mathsf{active}}(\varepsilon, \mathsf{u}, \mathsf{r}) \leq \mathsf{negl}(k).$$

This implies that δ is negligible in k and completes the proof. □

5.3 Security of the hHB Protocol against MIM Attacks

We prove here that hHB is secure against man-in-the-middle attacks.

Theorem 5.2. *Assume the* LPN_ε *problem is hard, where* $0 < \varepsilon < \frac{1}{2}$*. Then the hHB protocol with parameters* $r = r(k)$ *and* $\mathsf{u} > \varepsilon \cdot r$ *is secure against man-in-the-middle attacks.*

Proof. Let $\mathcal{A}$ be a probabilistic polynomial-time adversary tempering with messages between the tag and the reader in at most q executions of hHB protocol and achieving $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{MIM}}(\varepsilon, \mathsf{u}, \mathsf{r}) = \delta$.

In the first phase of its attack, $\mathcal{A}$ eavesdrops and modifies messages at will in order to gain informations on secrets by correlating its actions with the decision of the reader (acceptance or rejection).

For the second phase of the attack, we say for simplicity that $\mathcal{A}$ uses values $b = 0$. $\mathcal{A}$ has the probability δ of being authenticate by the reader. This means with probability δ, $\mathcal{A}$ does a good guess of the value of z in at least $r - \mathsf{u}$ rounds in the second part of hHB protocol. Therefore the probability that $\mathcal{A}$ gets a correct equation in the secret $\boldsymbol{x}$ (received from the reader) thus a correct equation in the secret $\boldsymbol{s}$ is at least $\delta(1 - \frac{u}{r})$. (This is because each bit x_i of x comes from an element of the triplet $(\alpha_i, \beta_i, \gamma_i)$ and each element of that triplet yields an equation in the secret s). But in order to get a correct equation in $\boldsymbol{s}$ one must know the element of $(\alpha_i, \beta_i, \gamma_i)$ which contains the value of x_i. Because of the way the reader transmits x to the tag which is an instance of the LPN and the application of f_s, the probability that $\mathcal{A}$ knows the value of x_i is at most $\frac{1}{3} + \frac{1}{(2^{k+1})^2}$, where $\frac{1}{(2^{k+1})^2}$ is the probability of having all the elements of the triplet equal. This implies that $\delta(1 - \frac{u}{r}) \leq (\frac{1}{3} + \frac{1}{(2^{k+1})^2})^k$. Since $(\frac{1}{3} + \frac{1}{(2^{k+1})^2})^k$ in negligible in k then δ itself is negligible. This completes the proof. □

5.4 *h*HB security settings

We respectively denote by k_s, k_x and k_y the length of the secrets s, x and y. The first phase of hHB consists of the secure transmission of the secret x which relies on the LPN problem with secret s and $\varepsilon \in [0.49, 0.5[$. Taking into account the recommendations of Levieil et *al* [17], we can use $k_s = 256$ to achieve at least 88 bits security. For the second phase of the hHB protocol corresponding to an execution of the HB^+ with $\varepsilon = 0.25$ the same recommendations from [17] can be applied, that is $k_x = 80$ and $k_y = 512$ to achieve 80 bits security. Using $r = 1164$ and $u = 0.348$, the probability of false acceptance and false rejection are respectively 2^{-80} and 2^{-40}.

The transmission cost of x is $3k_x(k_s + 1)$. For hHB that transmission cost is added to that of HB^+. When $k_x = 80$, the transmission cost of x is equal to 61680 bits which is substantially high. Nevertheless, the storage and computation cost of hHB remain low thus suited for low-cost hardware implementation.

6 Conclusion

In this paper we have presented a new protocol hHB which is a solution to thwart the man-in-the-middle attack against HB^+. The transmission cost of our protocol is quite high. But the hHB tag remains a tag as it is not overloaded (the storage and computation cost are substantially the same as for HB^+). Does securing HB^+ worth that transmission cost ? We say yes, but it would be very interesting to find a way to lower it while keeping the same level of security.

References

[1] E. R. Berlekamp, R. J. McEliece, and H. C. Van Tilborg. On the inherent intractability of certain coding problems. *IEEE Transactions on Information Theory*, 24(3):384–386, 1978.

[2] A. Blum, M. Furst, M. Kearns, and R. J. Lipton. Cryptographic primitives based on hard learning problems. In *Advances in cryptology—CRYPTO'93*, pages 278–291. Springer, 1994.

[3] A. Blum, A. Kalai, and H. Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *Journal of the ACM (JACM)*, 50(4):506–519, 2003.

[4] J. Bringer and H. Chabanne. Trusted-HB: a low-cost version of HB^+ secure against man-in-the-middle attacks. *arXiv preprint arXiv:0802.0603*, 2008.

[5] J. Bringer, H. Chabanne, and E. Dottax. HB^{++}: a lightweight authentication protocol secure against some attacks. In *Security, Privacy and Trust in Pervasive and Ubiquitous Computing, 2006. SecPerU 2006. Second International Workshop on*, pages 28–33. IEEE, 2006.

[6] D. N. Duc and K. Kim. Securing HB$^+$ against GRS man-in-the-middle attack. In *Institute of Electronics, Information and Communication Engineers, Symposium on Cryptography and Information Security*, 2007.

[7] M. P. Fossorier, M. J. Mihaljević, H. Imai, Y. Cui, and K. Matsuura. An algorithm for solving the LPN problem and its application to security evaluation of the HB protocols for RFID authentication. In *Progress in Cryptology-INDOCRYPT 2006*, pages 48–62. Springer, 2006.

[8] D. Frumkin and A. Shamir. Un-trusted-HB: Security vulnerabilities of trusted-HB. *IACR Cryptology ePrint Archive*, 2009:44, 2009.

[9] H. Gilbert, M. Robshaw, and H. Sibert. Active attack against HB$^+$: a provably secure lightweight authentication protocol. *Electronics Letters*, 41(21):1169–1170, 2005.

[10] H. Gilbert, M. J. Robshaw, and Y. Seurin. Good variants of HB$^+$ are hard to find. In *Financial Cryptography and Data Security*, pages 156–170. Springer, 2008.

[11] N. J. Hopper and M. Blum. A secure human-computer authentication scheme. In *Technical Report CMU-CS-00-139*. Carnegie Mellon University, 2000.

[12] N. J. Hopper and M. Blum. Secure human identification protocols. In *Advances in cryptology—ASIACRYPT 2001*, pages 52–66. Springer, 2001.

[13] A. Juels and S. A. Weis. Authenticating pervasive devices with human protocols. In *Advances in Cryptology–CRYPTO 2005*, pages 293–308. Springer, 2005.

[14] J. Katz and J. S. Shin. Parallel and concurrent security of the HB and HB$^+$ protocols. In *Advances in Cryptology-EUROCRYPT 2006*, pages 73–87. Springer, 2006.

[15] M. Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM (JACM)*, 45(6):983–1006, 1998.

[16] X. Leng, K. Mayes, and K. Markantonakis. HB-MP$^+$ protocol: An improvement on the HB-MP protocol. In *RFID, 2008 IEEE International Conference on*, pages 118–124. IEEE, 2008.

[17] É. Levieil and P. A. Fouque. An improved LPN algorithm. In *Security and Cryptography for Networks*, pages 348–359. Springer, 2006.

[18] J. Munilla and A. Peinado. HB-MP: A further step in the HB-family of lightweight authentication protocols. *Computer Networks*, 51(9):2262–2267, 2007.

[19] K. Ouafi, R. Overbeck, and S. Vaudenay. On the security of HB$^\#$ against a man-in-the-middle attack. In *Advances in Cryptology-ASIACRYPT 2008*, pages 108–124. Springer, 2008.

[20] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):34, 2009.