Practical Attacks on HB and HB+ Protocols

Zbigniew Gołębiewski[1], Krzysztof Majcher[2], Filip Zagórski[3]⋆, and Marcin Zawada[3]

[1] Institute of Computer Science, Wrocław University
[2] Mathematical Institute, Wrocław University
[3] Institute of Mathematics and Computer Science, Wrocław University of Technology
zbigniew.golebiewski@ii.uni.wroc.pl,
krzysztof.majcher@math.uni.wroc.pl,
{filipz,zawada}@im.pwr.wroc.pl

Version submitted to Crypto 2008 on 18 Feb 2008

Abstract. HB and HB+ are a shared-key authentication protocol designed for low-cost devices such as RFID tags. It was proposed by Juels and Weis at Crypto 2005. The security of the protocol relies on the "learning parity with noise" (LPN) problem, which was proved to be NP-hard.
The best known attack [4] on LPN requires exponential number of samples and exponential number of operations to be performed. This makes this attack impractical because it is infeasible to collect exponentially-many observations of the protocol execution.
We present a passive attack on HB protocol which requires only linear (to the length of the secret key) number of samples. Number of performed operations is still exponential, but attack is efficient for some real-life values of the parameters, i. e. noise $\frac{1}{8}$ and key length 144-bits.
Keywords: lightweight cryptography, RFID, authentication, HB, HB+, passive attack

1 Introduction

The HB Scheme HB and HB+ schemes are based on the "learning parity with noise" (LPN) problem, which was proved to be NP-hard. HB/HB+ are designed to be reliable authentication protocols for low-cost devices with small computational power.

Previous Attacks Over the last years, there have been proposed several attacks on the LPN problem. Most of them are tune-ups of the 2003 BKW algorithm (Blum, Kalai, Wasserman) [1].

The BKW algorithm takes an exponential (in the size of the secret key) number of samples and then tries to find a secret key by adding up sample vectors to receive vectors from canonical basis of vector space.

⋆ contact author

HB+ is vulnerable to man-in-the-middle attack proposed by Gilbert, Robshaw, and Silbert [3]. It is quite simple but it is questionable if it is possible to perform in the real-life.

Our Results In this paper we present completely different approach to attack LPN problem.

Our attack is "practical" – we need only to collect two successful executions of the HB protocol in order to start an attack. We assume that executions of the HB protocols use at least parameters computed in [4], i. e. number of bits being sent during single execution of the protocol is $O(n^2)$, where n is the key length. Number of bits required by the best known algorithm is $\Omega(2^n)$ while we need only to collect $O(n^3)$ bits.

Our first implementation of the algorithm breaks 80-bit HB with noise parameter $\frac{1}{4}$. We estimate that algorithm presented is able to practically break HB for noise parameter $\frac{1}{4}$ for keys of the length up to $n = 96$.

2 HB & HB+ protocols

The HB Protocol. The Tag and the Reader share public values: n, ϵ, η and a secret key x. In order to be authenticated by a Reader, the Tag and Reader repeat the following round of the protocol r times:

HB protocol

Public parameters: n, ϵ, η, r
Secret key: $x \in \{0,1\}^n$

Tag		Reader
	$\xleftarrow{a}$	choose $a \in \{0,1\}^n$
choose ν according to Ber_ϵ		
$l = <a\|x> \oplus \nu$	$\xrightarrow{l}$	check $l \stackrel{?}{=} <a\|x>$

The Reader authenticates the Tag if the number of accepted labels is at least $(1 - \eta) \cdot r$.

Efficiency of the HB As one can see, the efficiency of the HB protocol depends on three values: n, ϵ, r (in fact $r = r(\eta)$). The number of bits being sent during an authentication process by the reader is equal to $N_r(n, \eta) = n \cdot r(\eta)$, tag responds with $N_t = r(\eta)$ bits. Unfortunately, the simplicity in the hardware design badly influences on the protocol efficiency. According to [4], the number of bits being send during a reliable authentication are in the table below (all values in KB, $1KB = 8192b$).

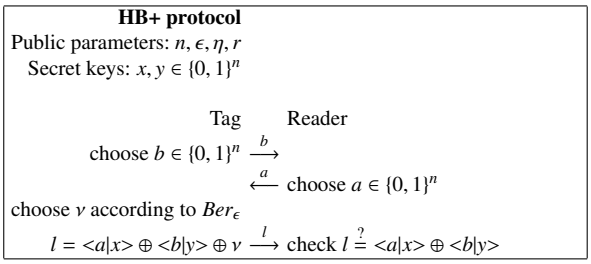
		η		
		$\frac{1}{20}$	$\frac{1}{8}$	$\frac{1}{4}$
n	128	4	7	18
	512	16	28	73

So, for some parameters of the HB/HB+ protocol, it may take seconds to authenticate even an expensive tag. The meaning of the "high-speed data rate" for RFIDs depends on the manufacturer and varies usually from $20KB/s$ to $40KB/s$. Low-end RFIDs are even 10-times slower.

This leads to the observation that for cheap RFIDs key length and number of rounds and thus noise parameter ϵ should be adjusted at the relatively low level.

The HB+ Protocol. The HB+ had been proposed as a protocol robust against active attacks (while HB is immune against passive attacks). Use of the blinding factor y turns an active attack on HB+ into a passive attack on HB.

In the HB+ scheme the Tag and the Reader share public values: n, ϵ, η, r and secret keys x, y. In order to be authenticated by a Reader, the Tag and Reader repeat the following round of the protocol r times:



Let us notice that if an attacker wants to break actively the HB+ Tag i. e. by sending appropriate values of a, she has to be able to passively break HB.

3 The Attack

Let $B = \{a_i \mid i = 1, \ldots, n\}$ be a basis of the linear space $V = \{0,1\}^n$. For a vector $v \in V$ let $l_{v,x}$ denote a correct answer (without noise) for the challenge v and secret key x for the HB. Let $L(B, x)$ denote a set of labels (correct answers) corresponding to each vector of the basis B.

Let $B(v)$ be a representation of a vector v in the basis B. Let $l_{B(v),x}$ be a linear combination of the labels of basis vectors corresponding to a representation of v in the basis B. Then, of course, for all $v \in V\, l_{B(v),x} = l_{v,x}$.

We use notation introduced above for the samples from the real observations, so now labels $l_{v,x}, l_{w,x}$ collected during eavesdropping the HB can differ, even for $v = w$ (when one of the answers of the Tag is with noise and the other one not).

Good Base Property Let us assume that we have collected n pairs challenge-response of the HB protocol - $B_0 = \{(a_i, l_{a_i,x}) \,|\, i = 1, \ldots, n\}$. Let us also assume that for each i during eavesdropping the noise value ν was always equal to 0. In other words, let us assume that we have collected correct answers (without noise), i. e. eavesdropped $l_{a_i,x} = <a_i|x>$. Then for every set of accepted authentications of a Tag t: $O = (\mathcal{O}, \mathcal{L}) = \{(a_j, l_{a_j,x}) \,|\, j = 1, \ldots, r\}$ the following inequality holds: $\left|\{j \,|\, l_{a_j,x} \neq l_{B(a_j)}\}\right| < \eta r$.

Definition 1. *We define a* 0*-Basis for a HB protocol instance* $(\epsilon, \eta, r, n, x)$ *as a set* $B_0 = \{(a_i, l_{a_i,x}) \,|\, i = 1, \ldots, n\}$ *for which:*

- $B = \{a_i \,|\, i = 1, \ldots n\}$ *is a basis of the linear space* $V = \{0, 1\}^n$*,*
- *for every* $O = \{(a_j, l_{a_j,x}) \,|\, j = 1, \ldots, r\}$ *where* $a_j \in V$*,* $|O| = r \geq n$ *and* $\left|\{j \,|\, l_{a_j,x} \neq <a_j|x>\}\right| < \eta r$ *it holds* $\left|\{j \,|\, l_{a_j,x} \neq l_{B(a_j)}\}\right| < \eta r$*.*

Let us remark that finding 0-Basis is equivalent to finding a secret key x. This is because, one can easily find a transition matrix from a basis B to a canonical basis and then multiplying labels of the B_0 by the transition matrix.

Brute-Force Attack Brute-Force Attack goes as follows:

1. collect a set of samples $O = (\mathcal{O}, \mathcal{L}) = \{(a_i, l_{a_i})\}$, $a_i \in \{0, 1\}^n$, $l_{a_i} \in \{0, 1\}$, $|O| \geq 2n$;
2. find a subset $O_B = (\mathcal{O}_\mathcal{B}, \mathcal{L}_\mathcal{B}) \subset O$ where $\mathcal{O}_\mathcal{B}$ is a basis for $\{0, 1\}^n$;
3. find a representation in the basis $\mathcal{O}_\mathcal{B}$ of all other collected vectors;
4. repeat
   (a) pick at random $w \in Ber_\epsilon$;
   (b) choose $w \cdot n$ labels in $\mathcal{L}_\mathcal{B}$ and switch their values;
   (c) check if O_B with new values of the labels is a 0-Basis (perform a test on the rest vectors of the set $\mathcal{O}$);

   until O_B with new values of labels is a 0-Basis;

The above Brute-Force Attack needs on average checking of the $\binom{n}{\epsilon n}$ subsets of different values of the labels.

3.1 Attack algorithm description

Let us describe the main idea behind our algorithm. The algorithm takes a set of observations O, then divides it into two parts. One part is used for 0-Basis-testing while the second one is used as a "universe" from which we pick at random potential 0-Basis.

As we show later, we need that the size of the set of the observations must be at least $|O| \geq n + \frac{n}{1-\eta} = O(n)$. The size of the testing set should be at least of

the size of the length of the secret key. We also need about $\frac{n}{1-\eta}$ observations to be sure that in the sample space there exists at least one 0-Basis. For parameters suggested in [4] and small keys (length smaller than 128), it occurs that our algorithm needs to collect observations from only 2 successful executions of the HB.

Let us assume that in each execution of the protocol we collect r pairs $(a_i, l_{a_i,x})$, where a_i is n-bits vector, x is n-bits secret and $l_{a_i,x}$ is a label of a_i. Let O_i denote a set of challenge vectors and by $\mathcal{L}_{O_i}$ a set of the labels corresponding to these challenges.

0-Basis Walker Algorithm
Input: public values of the HB - ϵ, η, r, n
$O = O_1 \cup \ldots \cup O_k$ - set of vectors and its labels $L = \mathcal{L}_{O_1} \cup \ldots \cup \mathcal{L}_{O_k}$
from k observations of the HB instance $(\epsilon, \eta, r, n, x)$

```
 1. split sets O, L into sets T, L_T and C, L_C respectively;
 2. do
 3. {
 4.    draw a set B ∈_R C, where |B| = n with corresponding labels L_B;
 5.    if(B is a basis of vector space V)
 6.    {
 7.      accepted := 0;
 8.      foreach vector w ∈ T
 9.      {
10.         find representation B(w);
11.         calculate l_{B(w)};
12.         if(l_{B(w),x} == l_{w,x}) { accepted++; }
13.      }
14.      if(accepted > (1 − η) ∗ |T|) { found_good_basis = TRUE; }
15.    }
16. } while(found_good_basis == FALSE);
```

After execution of the above algorithm we get a basis $B = \{b_1, \ldots, b_n\}$ and corresponding set of labels $L_B = \{l_{b_1,x}, \ldots, l_{b_n,x}\}$.

3.2 012-Basis Walker

Because one has to check if the 0-Basis test holds, one has to find a representation of the testing-vectors. It takes a while, so is worth to use the same basis several times. We define i-Basis as a set $O_B \subset O$ of size n of the observations of the HB protocol for which switching values of exactly i labels turns O_B into 0-Basis. To find a representations of test vectors in a basis O_B it takes $O(n^2 \cdot |S|)$, so it is worth to check if a set O_B is 1-Basis or 2-Basis, because checking i-Basis property requires $\binom{n}{i} \cdot |O_B|$ operations.

Let us call by 012-*Basis Walker Algorithm (*012-*BWA, BWA)* a modification of the 0-*Basis Walker Algorithm* which checks also 1- and 2-Basis property for every picked set. As we will see later this has a good influence on the efficiency of the algorithm.

Algorithm analysis Firstly let us find a probability that BWA finds i-Basis.

Lemma 1. *Let r be the number of rounds of the HB protocol. Let ϵ be the probability that a Tag during the HB answers with noise (label has incorrect value). Let η be the Fouque acceptance threshold - $\eta = \eta(\epsilon)$ [4] parameter (i. e. a Tag succeed in authenticating itself when the number of incorrect answers (labels) was less than $\eta \cdot r$. Let C denote a subset of the observations from which a basis B is picked at random ($|C| = c$). Let p_B denote the probability that randomly chosen set B is a basis of a vector space V. Then the probability that the set B picked uniformly at random from the set C is i-basis equals to*

$$p_i = p_B \binom{n}{i} \sum_{j=i}^{\lfloor \eta \cdot c \rfloor} \binom{c-n}{j-i} \epsilon^j (1-\epsilon)^{c-j}.$$

Proof. The probability that random chosen set B of size n from the set C is i-basis can be calculated as follows.

Let $C[j]$ denotes an event that are exactly j incorrect labels in the set C. Because $|C| = c$ and noise parameter equals to ϵ thus $P(C[j]) = \binom{c}{j}\epsilon^i(1-\epsilon)^{c-j}$

$$P(B \in_R C \text{ is an } i\text{-Basis} \mid \text{there is less than } \eta c \text{ incorrect labels in } C) = \sum_{j=i}^{\lfloor c \cdot \eta \rfloor} P(B \in_R C \text{ is an } i\text{-Basis} \mid \text{there is exactly } j \text{ incorrect labels in } C) \cdot P(C[j])$$

The sum starts from $j = i$ because if B is i-basis then in the set C must have at least i incorrect labels. The upper bound of the sum is $\lfloor c \cdot \eta \rfloor$ because we are assuming that C comes only from successful authentications. The probability p_B that set B is a basis over V is independent on the choices of the labels. The value $p_B \approx 0.2887$ comes from [2], it is approximately the probability that a random chosen set of the size n represents a basis (for $n \geq 20$). The probability that one taking n labels from the set of c labels, takes exactly i wrong labels is equal to $\frac{\binom{c-j}{n-i}\binom{j}{i}}{\binom{c}{n}}$. After few elementary simplifications we obtain $\frac{\binom{c-j}{n-i}\binom{j}{i}}{\binom{c}{n}} \cdot \binom{c}{j} = \binom{n}{i} \cdot \binom{c-n}{j-i}$. □

The expected value and the variance of the number of basis that should be tested.

Let X_i denote random variable that count the number of basis that should be tested before at most one i-basis is found. Variable X_i is obviously geometrically distributed with the success probability equal to $p_{X_i} = \sum_{j=0}^{i} p_i$. Thus the expected value of X_i is equal to $\frac{1}{p_{X_i}}$ and the variance is equal to $\frac{1-p_{X_i}}{p_{X_i}^2}$.

The expected value of the number of basis that should be tested for $|C| = 3 * n$.

	$\epsilon = 0.125, \eta = 0.256$	$\epsilon = 0.25, \eta = 0.348$
$n = 48$	68	24172
$n = 64$	348	$1.39 \cdot 10^6$
$n = 80$	1963	$9.04 \cdot 10^7$
$n = 96$	11865	$6.33 \cdot 10^9$
$n = 112$	75287	$4.67 \cdot 10^{11}$
$n = 128$	495413	$3.59 \cdot 10^{13}$
$n = 144$	$3.35 \cdot 10^6$	$2.84 \cdot 10^{15}$
$n = 160$	$2.32 \cdot 10^7$	$2.3 \cdot 10^{17}$

The expected value of the number of basis that should be tested for $|C| = n^2$.

	$\epsilon = 0.125$	$\epsilon = 0.25$
$n = 48$	44	5271
$n = 64$	167	146704
$n = 80$	694	$4.55 \cdot 10^6$
$n = 96$	3062	$1.51 \cdot 10^8$
$n = 112$	14108	$5.3 \cdot 10^9$
$n = 128$	67206	$1.92 \cdot 10^{11}$
$n = 144$	328581	$7.21 \cdot 10^{12}$
$n = 160$	$1.64 \cdot 10^6$	$2.76 \cdot 10^{14}$

Table 1. The value of η suggested in ([4]).

Finding Wrong Secrets Now we deal with the problem of getting secret keys different from the searched ones. In the current section we show how often a "bad" basis passes the test.

Lemma 2. *Let C be a set of observations of a tag with a secret key x. Let $i > 0$ and B be an i-Basis. Then for a test set T:*

$$P(l_t \neq l_{B(t)} \,|\, t \in T) = \frac{1}{2}.$$

Proof. (Sketch) We prove it by the induction. For $i = 1$ there exists one vector $b \in B$ for which label is wrong. This vector occurs in the representation of every vector $w \in T$ with probability $\frac{1}{2}$. Let us divide a set T into two subsets. In the subset T_c there are all observations with the correct value of the label, i. e. for $w \in T_c : l_w = <w|x>$. In the subset T_b there are vectors with wrong values of the labels.
$P(l_t \neq l_{B(t)} \,|\, t \in T) = P(l_t \neq l_{B(t)} \,|\, t \in T_b) + P(l_t \neq l_{B(t)} \,|\, t_i \in T_c) = \frac{\epsilon}{2} \cdot |T| + \frac{1-\epsilon}{2} \cdot |T| = \frac{1}{2}$. We have proved lemma for $k = 1$, now let us assume that we have it prover for k, we show that it holds for $k + 1$. Let B_{k+1} be a $(k + 1)$-Basis, then there exists exactly $k + 1$ k-Basis that differ from B_{k+1} on exactly one label. Let B_k be such a basis. Let us assume that B_k and B_{k+1} differ on the vector b. We have to consider two cases:
1^{st} For the vectors w that do not have b in the representation $l_{B_k(w)} = l_{B_{k+1}(w)}$, therefore $P(l_w = l_{B_{k+1}(w)}) = P(l_w = l_{B_k(w)}) = \frac{1}{2}$
2^{nd} For the vectors w that do have b in the representation $l_{B_k(w)} \neq l_{B_{k+1}(w)}$, therefore $P(l_w = l_{B_{k+1}(w)}) = P(l_w \neq l_{B_k(w)}) = \frac{1}{2}$ □

Corollary 1. *Let $i > 0$ and let B be an i-Basis. Let T be a set of test vectors, $|T| = t$. Then the probability that B passes a test is equal to $\left(\frac{1}{2}\right)^t \sum_{i=0}^{\eta \cdot t} \binom{t}{i}$.*

Proof. Having result form lemma 2 the proof is quite easy. (B, L_B) passes a test for at most $\eta \cdot t$ out of t vectors $w_j \in T$: $l_{w_i,x} \neq l_{B(w_i)}$. The probability that exactly d vectors from T disagree, is equal to $\binom{t}{d}\left(\frac{1}{2}\right)^d\left(1-\frac{1}{2}\right)^{t-d} = \binom{t}{d}\left(\frac{1}{2}\right)^t$. Now to end the proof we just sum such probabilities for all $d \in [0, \eta \cdot t]$. □

	$\eta = 0.125$	$\eta = 0.25$
$t = 50$	$1.62187 \cdot 10^{-8}$	0.000152932
$t = 100$	$9.55679 \cdot 10^{-16}$	$2.81814 \cdot 10^{-7}$
$t = 150$	$6.45682 \cdot 10^{-23}$	$1.91504 \cdot 10^{-10}$
$t = 200$	$3.27574 \cdot 10^{-29}$	$4.19651 \cdot 10^{-13}$
$t = 250$	$2.43421 \cdot 10^{-36}$	$3.11924 \cdot 10^{-16}$
$t = 300$	$1.84152 \cdot 10^{-43}$	$7.16702 \cdot 10^{-19}$

Table 2. The probability that wrong secret passes a test.

Improved basis selection In our algorithm we draw sets B and L_B of n vectors from the sets C and L_C respectively. The set B is a basis of vector space V with probability $p_B \approx 0.2887$. The value of this probability was introduced in [2].

One can significantly increase the probability of picking i-Basis (for $i = 0, 1, 2$):

1. Uniform random draw of the set B.
B can be drawn uniformly at random from the set C. Then the probability that B is i-basis is equal to

$$\Pr(B \text{ is } i\text{-basis} \mid B \in_R C) = p_B \cdot \frac{\binom{(1-\eta)c}{n-i}\binom{\eta c}{i}}{\binom{c}{n}}.$$

Thus the probability that such drawn set B is at most 2-basis and B pass the test in our algorithm is equal to:

$$p_C = \Pr(B \text{ is i-Basis } \cap\ B \text{ “passes the test”} \mid B \in_R C) =$$

$$= p_B \cdot \sum_{j=0}^{i} \frac{\binom{(1-\eta)c}{n-j}\binom{\eta c}{j}}{\binom{c}{n}}.$$

2. Uniform random draw of the set B from infinite set.

If we draw set of vectors B from infinite set then the probability the B is i-basis is equal to

$$\Pr(B \text{ is } i\text{-basis}) = p_B \binom{n}{i} \eta^i (1-\eta)^{n-i}.$$

Thus the probability that B is at most 2-basis and B passes the test in our algorithm is equal to:

$$\Pr(B \text{ is } i\text{-basis} \mid B \in_R C) =$$

$$\Pr(B \text{ is at most } i\text{-basis } \cap \ B \text{ "passes the test"}) = p_B \cdot \sum_{j=0}^{i} \binom{n}{j} \eta^j (1-\eta)^{n-j}.$$

It is obvious that this probability is the limit of the previous probability p_C. Values of the expected value for the limit distribution are in the chart 1.

3. Improved uniform random draw of the set B.

As we have written earlier the sets C, L_C are sums of k sets of the vectors collected in one course of HB protocol (i. e.. $C = C_1 \cup \ldots \cup C_k$ and $L_C = L_{C_1} \cup \ldots \cup L_{C_k}$). We also know that in each well ended HB protocol course fewer than ηr vectors are incorrect. Thus if we draw $\frac{n}{k}$ vectors and its labels from each C_i and L_{L_i} respectively then the probability that B is 0-basis is

$$\Pr(B \text{ is 0-basis} | B \text{ is improved drawn from a set of size } c) = p_B \cdot \left(\frac{\binom{(1-\eta)\frac{c}{k}}{\frac{n}{k}}}{\binom{\frac{c}{k}}{\frac{n}{k}}} \right)^k.$$

Equivalently we can derive formulas for the probability that B is 1-basis and 2-basis. If a number of collected observations of at least n successful the HB protocol courses then we can pick a basis' with the probability of the limit distribution (point 2).

3.3 Experimental Results

We have implemented and tested our algorithm for several values. We have broken HB for the parameter $\epsilon = 0.125, \eta = 0.256, n = 144$ it took few hours on home PC. For the parameters $\epsilon = 0.25, \eta = 0.348, n = 80$ it takes average 10 hours on home PC.

This results and the values in chart 1 suggest, that we are able to break $n = 96$ bit version of 0.25-HB and $n = 154$ bit version of 0.125-HB protocol.

Final Remarks

We have shown a passive attack for the HB protocol which allow to perform an active attack for HB+ scheme (not man-in-the middle). Our attack needs only $O(n)$ eavesdropped pairs of challenge-response, where n is the length of a secret key, while the best known algorithm *LF2* needs exponential number of samples.

References

1. A. Blum, A. Kalai, and H. Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. In *Journal of the ACM, vol. 50, no. 4*, pages 506–519, 2003. Available from: `citeseer.ist.psu.edu/article/blum03noisetolerant.html`.
2. Jacek Cichon, Marek Klonowski, and Miroslaw Kutylowski. Privacy protection for rfid with hidden subset identifiers. In *Pervasive Computing*, 2008.
3. H. Gilbert, H. Sibert, and M. Robshaw. An active attack against a provably secure lightweight authentication protocol. In *IEEE Electronic Letters 41*, 2005.
4. Éric Levieil and Pierre-Alain Fouque. An improved lpn algorithm. In Roberto De Prisco and Moti Yung, editors, *SCN*, volume 4116 of *Lecture Notes in Computer Science*, pages 348–359. Springer, 2006.