Analyzing the HB and HB$^+$ Protocols in the "Large Error" Case*

Jonathan Katz† Adam Smith‡

Abstract

HB and HB$^+$ are two shared-key, unidirectional authentication protocols whose extremely low computational cost makes them potentially well-suited for severely resource-constrained devices. Security of these protocols is based on the conjectured hardness of *learning parity with noise*; that is, learning a secret $\mathbf{s}$ given "noisy" dot products of $\mathbf{s}$ that are incorrect with probability ε.

Although the problem of learning parity with noise is meaningful for any constant $\varepsilon < 1/2$, existing proofs of security for HB and HB$^+$ only imply security when $\varepsilon < 1/4$. In this note, we show how to extend these proofs to the case of arbitrary $\varepsilon < 1/2$.

1 Background[1]

The HB and HB$^+$ protocols, introduced by Hopper and Blum [7, 8] and Juels and Weis [11] respectively, are shared-key, unidirectional authentication protocols whose efficiency makes them potentially suitable for resource-constrained devices such as RFID tags. The HB protocol is intended to be secure against a passive (eavesdropping) adversary, while the HB$^+$ protocol is intended to be secure against an active adversary.

Security of these protocols is based on the problem of learning parity with noise (the *LPN problem*) [1, 2, 3, 4, 6, 13, 7, 8, 14]. Roughly speaking (see Section 2.1 for a formal definition), this problem is to determine a secret value $\mathbf{s}$ given "noisy" dot products of $\mathbf{s}$ with a sequence of randomly-chosen vectors. These dot products are "noisy" in that they are each *incorrect* with (independent) probability ε, where ε is a fixed constant. The LPN problem is meaningful for any constant $\varepsilon \in (0, \frac{1}{2})$.

Juels and Weis [11] gave the first proofs of security for the HB and HB$^+$ protocols based on the hardness of the LPN problem. Although their proofs tolerate any value of ε, their results have some limitations: (1) they do not handle multiple iterations of the protocol, but instead only analyze a "basic authentication step" which does not by itself provide adequate security; and (2) they do not handle parallel or concurrent executions of the HB$^+$ protocol. (We refer the reader to [12] for a detailed discussion.) Katz and Shin [12] gave proofs that overcame these limitations, but their proofs only imply meaningful security (in either an asymptotic or a concrete sense) for $\varepsilon < \frac{1}{4}$.

---

*Work done while the authors were visiting IPAM.

†Dept. of Computer Science, University of Maryland. jkatz@cs.umd.edu. This research was supported by NSF Trusted Computing grants #0310751 and #0627306, and NSF CAREER award #0447075.

‡Dept. of Computer Science and Engineering, Pennsylvania State University. asmith@cse.psu.edu.

[1]While we provide some minimal background, our assumption is that the reader is already familiar with [12].

Our contribution. In this note, we show how to adapt the work of Katz and Shin so as to obtain proofs of security for the full HB and HB$^+$ protocols for arbitrary $\varepsilon < \frac{1}{2}$. We stress that in doing so we retain all the advantages of their proofs relative to those of Juels and Weis.

Our focus in this manuscript is on proving *asymptotic* security only, without attempting to optimize the quality of the reduction at all. Nevertheless, our proofs readily yield expressions that can be used to calculate the concrete security of the protocols (relative to the LPN problem) for particular settings of the parameters.

2 A Brief Review

We formally define the LPN problem, and state a key technical lemma used by [12] in their analysis that we will also use here. We also quickly review the HB protocol and the security models under consideration. The HB$^+$ protocol is described in a later section.

2.1 The LPN Problem

Our formulation of the LPN problem follows [12] except that we focus on *asymptotic* rather than *concrete* security. Let k be a security parameter. If $\mathbf{s}, \mathbf{a}_1, \ldots, \mathbf{a}_\ell$ are binary vectors of length k, let $z_i = \langle \mathbf{s}, \mathbf{a}_i \rangle$ denote the dot product of $\mathbf{s}$ and $\mathbf{a}_i$ (modulo 2). Given the values $\mathbf{a}_1, z_1, \ldots, \mathbf{a}_\ell, z_\ell$ for randomly-chosen $\{\mathbf{a}_i\}$ and $\ell = \Theta(k)$, it is possible to efficiently solve for $\mathbf{s}$ using standard linear-algebraic techniques. However, in the presence of *noise* where each z_i is flipped (independently) with probability ε, finding $\mathbf{s}$ becomes more difficult and no polynomial-time algorithm for finding $\mathbf{s}$ in this case is currently known. In fact, the problem is NP-hard in the worst case [1].

Formally, let Ber_ε be the Bernoulli distribution with parameter $\varepsilon \in (0, \frac{1}{2})$ (so if $\nu \leftarrow \mathsf{Ber}_\varepsilon$ then $\Pr[\nu = 1] = \varepsilon$ and $\Pr[\nu = 0] = 1 - \varepsilon$), and let $A_{\mathbf{s},\varepsilon}$ denote an oracle which outputs (independent) samples according to the following distribution:

$$\left\{ \mathbf{a} \leftarrow \{0,1\}^k; \nu \leftarrow \mathsf{Ber}_\varepsilon : (\mathbf{a}, \langle \mathbf{s}, \mathbf{a} \rangle \oplus \nu) \right\}.$$

We say the LPN_ε *problem is hard* if for all PPT algorithms M the following is negligible:

$$\Pr\left[\mathbf{s} \leftarrow \{0,1\}^k : M^{A_{\mathbf{s},\varepsilon}}(1^k) = \mathbf{s} \right].$$

Note that ε is taken to be a fixed constant independent of k.

A technical lemma. The following lemma, implicit in [14] and reproved in [12], states that hardness of the LPN_ε problem implies the pseudorandomness of $A_{\mathbf{s},\varepsilon}$ (for randomly-chosen $\mathbf{s}$). In the following, U_{k+1} denotes an oracle that returns uniformly-distributed strings of length $k+1$.

Lemma 1 *Assuming the hardness of the* LPN_ε *problem, the following is negligible for all* PPT *algorithms* D*:*

$$\left| \Pr\left[\mathbf{s} \leftarrow \{0,1\}^k : D^{A_{\mathbf{s},\varepsilon}}(1^k) = 1 \right] - \Pr\left[D^{U_{k+1}}(1^k) = 1 \right] \right|.$$

2.2 Overview of the HB/HB$^+$ Protocols, and Security Definitions

Recall that we let k denote our security parameter. The HB and HB$^+$ protocols as analyzed here consist of $n = n(k)$ *parallel* iterations of a "basic authentication step." We describe the basic authentication step for the HB protocol, and defer a discussion of the HB$^+$ protocol to Section 3.2.

In the HB protocol, a tag $\mathcal{T}$ and a reader $\mathcal{R}$ share a random secret key $\mathbf{s} \in \{0,1\}^k$; a basic authentication step consists of the reader sending a random challenge $\mathbf{a} \in \{0,1\}^k$ to the tag, which replies with $z = \langle \mathbf{s}, \mathbf{a} \rangle \oplus \nu$ for $\nu \sim \mathsf{Ber}_\varepsilon$. The reader can then verify whether the response z of the tag satisfies $z \stackrel{?}{=} \langle \mathbf{s}, \mathbf{a} \rangle$; we say the iteration is *successful* if this is the case. See Figure 1.

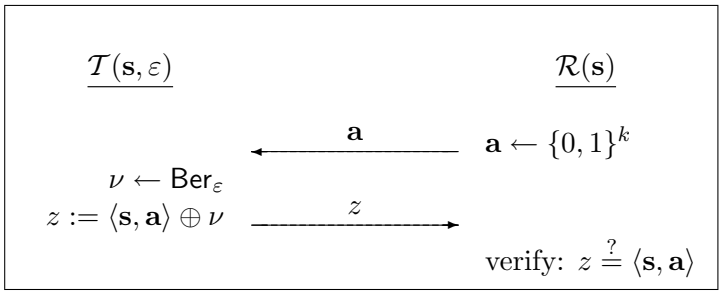

Figure 1: The basic authentication step of the HB protocol.

Even for an honest tag a basic iteration is unsuccessful with probability ε. For this reason, a reader accepts upon completion of all n iterations of the basic authentication step as long as at most $\approx \varepsilon \cdot n$ of these iterations were unsuccessful. More precisely, let $\mathsf{u} = \mathsf{u}(k)$ be such that $\varepsilon \cdot n \leq \mathsf{u}$; then the reader accepts as long as the number of unsuccessful iterations is at most[2] u. (Overall, then, the entire HB protocol is parameterized by ε, u, and n.) Since εn is the expected number of unsuccessful iterations for an honest tag, the completeness error ε_c (i.e., the probability that an honest tag is rejected) can be calculated via a Chernoff bound. In particular, we have that for any positive constant δ, setting $\mathsf{u} = (1+\delta)\varepsilon n$ suffices to achieve ε_c negligible in n.

Observe that by sending random answers in each of the n iterations, an adversary trying to impersonate a valid tag succeeds with probability

$$\delta^*_{\varepsilon,\mathsf{u},n} \stackrel{\text{def}}{=} 2^{-n} \cdot \sum_{i=0}^{\mathsf{u}} \binom{n}{i};$$

that is, $\delta^*_{\varepsilon,\mathsf{u},n}$ is the *best* possible soundness error we can hope to achieve for the given setting of the parameters. Asymptotically, as long as $\mathsf{u} \leq (1-\delta) \cdot n/2$ for positive constant δ, the success of this trivial attack will be negligible in n. (This can again be analyzed using a Chernoff bound.)

Let $\mathcal{T}^{\mathsf{HB}}_{\mathbf{s},\varepsilon,n}$ denote the tag algorithm in the HB protocol when the tag holds secret key $\mathbf{s}$ (note that the tag algorithm is independent of u), and let $\mathcal{R}^{\mathsf{HB}}_{\mathbf{s},\varepsilon,\mathsf{u},n}$ similarly denote the algorithm run by the tag reader. We denote a complete execution of the HB protocol between a party $\hat{\mathcal{T}}$ and the reader $\mathcal{R}$ by $\left\langle \hat{\mathcal{T}}, \mathcal{R}^{\mathsf{HB}}_{\mathbf{s},\varepsilon,\mathsf{u},n} \right\rangle$ and say this equals 1 iff the reader accepts.

For the case of a passive attack on the HB protocol, we imagine a polynomial-time adversary $\mathcal{A}$ running in two stages: in the first stage the adversary obtains polynomially-many transcripts[3] of (honest) executions of the protocol by interacting with an oracle $\mathsf{trans}^{\mathsf{HB}}_{\mathbf{s},\varepsilon,n}$ (this models eavesdropping); in the second stage, the adversary interacts with the reader and tries to impersonate

[2] As suggested in [12], a slight improvement in practice is to also fix a *lower bound* l and accept iff the number of unsuccessful iterations is in the range $[\mathsf{l}, \mathsf{u}]$. Setting $\mathsf{l} = 0$ (as we do here) makes no difference in an asymptotic sense.

[3] Note in particular that the adversary is assumed not to learn whether or not the reader accepts. Since, as discussed earlier, the parameters can be set such that the reader accepts an honest tag with all but negligible probability, this makes no difference as far as asymptotic security is concerned.

the tag. We define the adversary's advantage as

$$\mathsf{Adv}^{passive}_{\mathcal{A},\mathsf{HB}}(\varepsilon,\mathsf{u},n) \stackrel{\text{def}}{=} \Pr\left[\mathbf{s} \leftarrow \{0,1\}^k; \mathcal{A}^{\mathsf{trans}^{\mathsf{HB}}_{\mathbf{s},\varepsilon,n}}(1^k) : \left\langle \mathcal{A}, \mathcal{R}^{\mathsf{HB}}_{\mathbf{s},\varepsilon,\mathsf{u},n} \right\rangle = 1\right].$$

We say the HB protocol is *secure against passive attacks* (for a particular setting of ε and $\mathsf{u} = \mathsf{u}(k)$, $n = n(k)$) if for all PPT adversaries $\mathcal{A}$ we have that $\mathsf{Adv}^{passive}_{\mathcal{A},\mathsf{HB}}(\varepsilon,\mathsf{u},n)$ is negligible in k.

As we will describe in Section 3.2, the HB$^+$ protocol uses two keys $\mathbf{s}_1, \mathbf{s}_2$. We let $\mathcal{T}^{\mathsf{HB}^+}_{\mathbf{s}_1,\mathbf{s}_2,\varepsilon,n}$ denote the tag algorithm in this case, and let $\mathcal{R}^{\mathsf{HB}^+}_{\mathbf{s}_1,\mathbf{s}_2,\varepsilon,\mathsf{u},n}$ denote the algorithm run by the tag reader. For the case of an active attack on the HB$^+$ protocol, we again imagine an adversary running in two stages: in the first stage the adversary interacts polynomially-many times with the honest tag algorithm (with concurrent executions allowed), while in the second stage the adversary interacts only with the reader. The adversary's advantage in this case is

$$\mathsf{Adv}^{active}_{\mathcal{A},\mathsf{HB}^+}(\varepsilon,\mathsf{u},n) \stackrel{\text{def}}{=} \Pr\left[\mathbf{s}_1,\mathbf{s}_2 \leftarrow \{0,1\}^k; \mathcal{A}^{\mathcal{T}^{\mathsf{HB}^+}_{\mathbf{s}_1,\mathbf{s}_2,\varepsilon,n}}(1^k) : \left\langle \mathcal{A}, \mathcal{R}^{\mathsf{HB}^+}_{\mathbf{s}_1,\mathbf{s}_2,\varepsilon,\mathsf{u},n} \right\rangle = 1\right].$$

We say the HB$^+$ protocol is *secure against active attacks* (for a particular setting of ε and $\mathsf{u} = \mathsf{u}(k)$, $n = n(k)$) if for all PPT adversaries $\mathcal{A}$ we have that $\mathsf{Adv}^{active}_{\mathcal{A},\mathsf{HB}^+}(\varepsilon,\mathsf{u},n)$ is negligible in k.

We remark that allowing the adversary to interact with the reader multiple times (even concurrently) does not give the adversary any additional advantage other than the fact that, as usual, the probability that the adversary succeeds in at least one impersonation attempt scales linearly with the number of attempts.

3 Proofs of Security for the HB and HB$^+$ Protocols

In this section, we show how to modify the proofs of security given in [12] so as to obtain a meaningful security reduction for arbitrary $\varepsilon < \frac{1}{2}$.

3.1 Security of the HB Protocol against Passive Attacks

Recall from the previous section that the HB protocols is parameterized by ε (a measure of the noise introduced by the tag), u (which determines the completeness error ε_c as well as the best achievable soundness), and n (the number of iterations of the basic authentication step given in Figure 1). We stress that these n iterations are run *in parallel*, so the entire protocol requires only two rounds.

Theorem 2 *Assume the* LPN_ε *problem is hard, where* $0 < \varepsilon < \frac{1}{2}$. *Let* $n = \Theta(k)$ *and* $\mathsf{u} = \varepsilon^+ \cdot n$, *where* ε^+ *is a non-negative constant satisfying*

$$\varepsilon < \varepsilon^+ < \frac{1}{2}.$$

Then the HB protocol with these settings of the parameters has negligible completeness error, and for all PPT *adversaries* $\mathcal{A}$ *the quantity* $\delta \stackrel{\text{def}}{=} \mathsf{Adv}^{passive}_{\mathcal{A},\mathsf{HB}}(\varepsilon,\mathsf{u},n)$ *is negligible.*

Proof A standard Chernoff bound shows that the completeness error is negligible for the given setting of the parameters. To prove security of the protocol, we use the reduction given in [12]; only the analysis is different. For any PPT adversary $\mathcal{A}$ attacking the HB protocol, we construct a PPT adversary D attempting to distinguish whether it is given oracle access to $A_{\mathbf{s},\varepsilon}$ or to U_{k+1} (as in Lemma 1). Relating the advantage of D to the advantage of $\mathcal{A}$ gives the stated result.

D, given access to an oracle returning $(k+1)$-bit strings $(\mathbf{a}, z)$, proceeds as follows:

1. D runs the first phase of $\mathcal{A}$. Each time $\mathcal{A}$ requests to view a transcript of the protocol, D obtains n samples $\{(\mathbf{a}_i, z_i)\}_{i=1}^n$ from its oracle and returns these to $\mathcal{A}$.
2. When $\mathcal{A}$ is ready for the second phase, D again obtains n samples $\{(\bar{\mathbf{a}}_i, \bar{z}_i)\}_{i=1}^n$ from its oracle. D then sends the challenge $(\bar{\mathbf{a}}_1, \ldots, \bar{\mathbf{a}}_n)$ to $\mathcal{A}$ and receives in return a response $Z' = (z'_1, \ldots, z'_n)$.
3. D outputs 1 iff $\bar{Z} = (\bar{z}_1, \ldots, \bar{z}_n)$ and Z' differ in at most $\mathsf{u}' \stackrel{\text{def}}{=} \varepsilon^{++} \cdot n$ entries, where ε^{++} is a constant satisfying $\varepsilon^+ - 2\varepsilon^+\varepsilon + \varepsilon < \varepsilon^{++} < \frac{1}{2}$.

When D's oracle is U_{k+1}, it is clear that D outputs 1 with probability $2^{-n} \cdot \sum_{i=0}^{\mathsf{u}'} \binom{n}{i}$ since $\bar{Z}$ is in this case uniformly distributed and independent of everything else. Since $\mathsf{u}' < n/2$, this quantity is negligible in k for the given setting of the other parameters.

When D's oracle is $A_{\mathbf{s},\varepsilon}$ then the transcripts D provides to $\mathcal{A}$ during the first phase of $\mathcal{A}$'s execution are distributed identically to real transcripts in an execution of the HB protocol. Letting $Z^* \stackrel{\text{def}}{=} (\langle \mathbf{s}, \bar{\mathbf{a}}_1 \rangle, \ldots, \langle \mathbf{s}, \bar{\mathbf{a}}_n \rangle)$ be the vector of correct answers to the challenge $(\bar{\mathbf{a}}_1, \ldots, \bar{\mathbf{a}}_n)$ sent by D in the second phase, it follows that with probability δ (i.e., the impersonation probability of $\mathcal{A}$) the vector of responses Z' given by $\mathcal{A}$ differs from Z^* in at most u entries. We show below that conditioned on this event, Z' and $\bar{Z}$ differ in at most u' entries with all but negligible probability. Thus, D outputs 1 in this case with probability negligibly close to δ. We conclude from Lemma 1 that δ must be negligible.

Let $\mathbf{wt}(Z)$ denote the *weight* of a vector Z; i.e., the number of entries of Z equal to 1. Note that the distance between two binary vectors Z_1, Z_2 is equal to $\mathbf{wt}(Z_1 \oplus Z_2)$. It remains to show that, conditioned on $\mathbf{wt}(Z' \oplus Z^*) \leq \mathsf{u}$, we have $\mathbf{wt}(Z' \oplus \bar{Z}) \leq \mathsf{u}'$ with all but negligible probability.

Write $Z' = Z^* \oplus \mathbf{w}$ for some vector $\mathbf{w}$ of weight at most $\mathsf{u} = \varepsilon^+ n$. The vector $\bar{Z}$ is selected by the following process: choose an error vector $\mathbf{e}$ by setting each position of $\mathbf{e}$ (independently) to 1 with probability ε, and then set $\bar{Z} = Z^* \oplus \mathbf{e}$. We see that the probability that $\bar{Z}$ differs from Z' in at most u' entries is equal to the probability that

$$\mathbf{wt}(Z' \oplus \bar{Z}) = \mathbf{wt}(\mathbf{w} \oplus \mathbf{e}) \leq \mathsf{u}'.$$

It is easy to see that this probability is minimized when $\mathbf{wt}(\mathbf{w}) = \mathsf{u}$, and so we assume this to be the case. The random variable $\mathbf{wt}(\mathbf{w} \oplus \mathbf{e})$ can be written as a sum of n indicator random variables, one for each position of the vector $\mathbf{w} \oplus \mathbf{e}$. The expectation of $\mathbf{wt}(\mathbf{w} \oplus \mathbf{e})$ is

$$\mathsf{u} \cdot (1 - \varepsilon) + (n - \mathsf{u}) \cdot \varepsilon = (\varepsilon^+ - 2\varepsilon^+\varepsilon + \varepsilon) \cdot n.$$

Since ε^{++} is a constant strictly larger than $(\varepsilon^+ - 2\varepsilon^+\varepsilon + \varepsilon)$, a Chernoff bound then implies that $\mathbf{wt}(\mathbf{w} \oplus \mathbf{e}) \leq \varepsilon^{++} n$ with all but negligible probability. ■

3.2 Security of the HB$^+$ Protocol against Active Attacks

It is easy to see that the HB protocol is insecure against an active adversary. To achieve security against active attacks, Juels and Weis propose to modify the HB protocol by having the tag and reader share *two* (independent) keys $\mathbf{s}_1, \mathbf{s}_2 \in \{0,1\}^k$. A basic authentication step now consists of three rounds: first the tag sends a random "blinding factor" $\mathbf{b} \in \{0,1\}^k$; the reader replies with a random challenge $\mathbf{a} \in \{0,1\}^k$ as before; and finally the tag replies with $z = \langle \mathbf{s}_1, \mathbf{b} \rangle \oplus \langle \mathbf{s}_2, \mathbf{a} \rangle \oplus \nu$ for $\nu \leftarrow \mathsf{Ber}_\varepsilon$. As in the HB protocol, the tag reader can verify whether the response z satisfies $z \stackrel{?}{=} \langle \mathbf{s}_1, \mathbf{b} \rangle \oplus \langle \mathbf{s}_2, \mathbf{a} \rangle$, and we again say the iteration is *successful* if this is the case. See Figure 2.

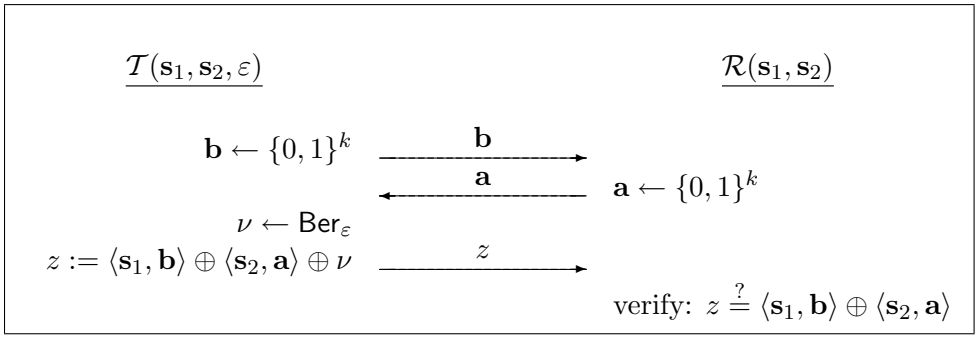

Figure 2: The basic authentication step of the HB$^+$ protocol.

The actual HB$^+$ protocol consists of n parallel iterations of the basic authentication step (and so the entire protocol requires only three rounds). The protocol also depends upon a parameter u as in the case of the HB protocol, and this will again affect the completeness error as well as the best achievable soundness.

Theorem 3 *Assume the* LPN_ε *problem is hard, where* $0 < \varepsilon < \frac{1}{2}$. *Let* $n = \Theta(k)$ *and* $\mathsf{u} = \varepsilon^+ \cdot n$ *where* ε^+ *is a constant satisfying*

$$\varepsilon < \varepsilon^+ < \frac{1}{2}.$$

Then the HB$^+$ protocol with these settings of the parameters has negligible completeness error, and for all PPT *adversaries* $\mathcal{A}$ *the quantity* $\delta_{\mathcal{A}} = \mathsf{Adv}^{active}_{\mathcal{A},\mathsf{HB}^+}(\varepsilon, \mathsf{u}, n)$ *is negligible.*

Proof A standard Chernoff bound shows that the completeness error is negligible for the given setting of the parameters. To prove security of the protocol, we use the reduction given in [12]; only the analysis is different. For any PPT adversary $\mathcal{A}$ attacking the HB$^+$ protocol, we construct a PPT adversary D attempting to distinguish whether it is given oracle access to $A_{\mathbf{s},\varepsilon}$ or to U_{k+1} (as in Lemma 1). Relating the advantage of D to the advantage of $\mathcal{A}$ gives the stated result.

D, given access to an oracle returning $(k+1)$-bit strings $(\mathbf{b}, \bar{z})$, proceeds as follows:

1. D chooses $\mathbf{s}_2 \in \{0,1\}^k$ uniformly at random. Then, it runs the first phase of $\mathcal{A}$. To simulate a basic authentication step, D does the following: it obtains a sample $(\mathbf{b}, \bar{z})$ from its oracle and sends $\mathbf{b}$ as the initial message. $\mathcal{A}$ replies with a challenge $\mathbf{a}$, and then D responds with $z = \bar{z} \oplus \langle \mathbf{s}_2, \mathbf{a} \rangle$. Note that D does not rewind $\mathcal{A}$ here, so there is no difficulty simulating parallel or concurrent executions.

2. When $\mathcal{A}$ is ready for the second phase of its attack, $\mathcal{A}$ sends an initial message $\mathbf{b}_1, \ldots, \mathbf{b}_n$. In response, D chooses random $\mathbf{a}_1^1, \ldots, \mathbf{a}_n^1 \in \{0,1\}^k$, sends these challenges to $\mathcal{A}$, and records $\mathcal{A}$'s response $z_1^1, \ldots, z_n^1$. Then D rewinds $\mathcal{A}$, chooses random $\mathbf{a}_1^2, \ldots, \mathbf{a}_n^2 \in \{0,1\}^k$, sends these to $\mathcal{A}$, and records $\mathcal{A}$'s response $z_1^2, \ldots, z_n^2$.

3. Let $z_i^\oplus := z_i^1 \oplus z_i^2$ and set $Z^\oplus \stackrel{\text{def}}{=} (z_1^\oplus, \ldots, z_n^\oplus)$. Let $\hat{\mathbf{a}}_i = \mathbf{a}_i^1 \oplus \mathbf{a}_i^2$ and $\hat{z}_i = \langle \mathbf{s}_2, \hat{\mathbf{a}}_i \rangle$, and set $\hat{Z} \stackrel{\text{def}}{=} (\hat{z}_1, \ldots, \hat{z}_n)$. D outputs 1 iff $Z^\oplus$ and $\hat{Z}$ differ in fewer than $\mathsf{u}' = \varepsilon^{++} n$ entries. We will fix the constant ε^{++} later, but it will satisfy $\varepsilon^{++} < \frac{1}{2}$.

Let us analyze the behavior of D:

Case 1: Say D's oracle is U_{k+1}. In step 1, above, since $\bar{z}$ is uniformly distributed and independent of everything else, the answers z that D returns to $\mathcal{A}$ are uniformly distributed and independent

of everything else. It follows that $\mathcal{A}$'s view throughout the entire experiment is independent of the secret $\mathbf{s}_2$ chosen by D.

The $\{\hat{\mathbf{a}}_i\}_{i=1}^n$ are uniformly and independently distributed, and so except with probability $\frac{2^n}{2^k}$ they are linearly independent and non-zero (see [12]). Assuming this to be the case, $\hat{Z}$ is uniformly distributed over $\{0,1\}^n$ from the point of view of $\mathcal{A}$. But then the probability that $Z^{\oplus}$ and $\hat{Z}$ differ in fewer than u' entries is at most $2^{-n} \cdot \sum_{i=0}^{\lfloor \mathsf{u}' \rfloor} \binom{n}{i}$. Since $\mathsf{u}' < n/2$, we conclude that D outputs 1 in this case with negligible probability $\frac{2^n}{2^k} + 2^{-n} \cdot \sum_{i=0}^{\lfloor \mathsf{u}' \rfloor} \binom{n}{i}$.

Case 2: Say D's oracle is $A_{\mathbf{s}_1,\varepsilon}$ for randomly-chosen $\mathbf{s}_1$. In this case, D provides a perfect simulation for the first phase of $\mathcal{A}$. Let ω denote all the randomness used to simulate the first phase of $\mathcal{A}$, which includes the keys $\mathbf{s}_1, \mathbf{s}_2$, the randomness of $\mathcal{A}$, and the randomness used in responding to $\mathcal{A}$'s queries. For a fixed ω, let δ_ω denote the probability (over random challenges $\mathbf{a}_1, \ldots, \mathbf{a}_n$ sent by the tag reader) that $\mathcal{A}$ successfully impersonates the tag in the second phase. Note that the probability that $\mathcal{A}$ successfully responds to both sets of queries $\mathbf{a}_1^1, \ldots, \mathbf{a}_n^1$ and $\mathbf{a}_1^2, \ldots, \mathbf{a}_n^2$ sent by D is δ_ω^2. The overall probability that $\mathcal{A}$ successfully responds to both sets of queries is thus

$$\mathsf{Exp}_\omega\left(\delta_\omega^2\right) \geq \left(\mathsf{Exp}_\omega(\delta_\omega)\right)^2 = \delta_{\mathcal{A}}^2,$$

using Jensen's inequality.[4]

We show below that conditioned on both challenges being answered successfully (and for appropriate choice of ε^{++}), $Z^{\oplus}$ differs from $\hat{Z}$ in fewer than u' entries with *constant* probability. Putting everything together, we conclude that D outputs 1 in this case with probability $\Omega(\delta_{\mathcal{A}}^2)$. It follows from Lemma 1 that $\delta_{\mathcal{A}}$ must be negligible.

We now prove the above claim regarding the probability that $Z^{\oplus}$ differs from $\hat{Z}$ in fewer than u' entries. Set $\frac{1}{2} > \varepsilon^{++} > \frac{1}{2} \cdot (1 - (1-2\varepsilon^{+})^2)$. Fixing all randomness used in the first phase (as above) induces a function $f_{\mathcal{A}}$ from queries $\mathbf{a}_1, \ldots, \mathbf{a}_n$ (with each $\mathbf{a}_i \in \{0,1\}^k$) to vectors $(z_1, \ldots, z_n)$ (with each $z_i \in \{0,1\}$) given by the response function of $\mathcal{A}$ in the second phase. Define the function f_{correct} that returns the "correct" answers for a particular query; i.e.,

$$f_{\text{correct}}(\mathbf{a}_1, \ldots, \mathbf{a}_n) \stackrel{\text{def}}{=} \left(\langle \mathbf{s}_1, \mathbf{b}_1 \rangle \oplus \langle \mathbf{s}_2, \mathbf{a}_1 \rangle, \ldots, \langle \mathbf{s}_1, \mathbf{b}_n \rangle \oplus \langle \mathbf{s}_2, \mathbf{a}_n \rangle\right)$$

(recall that $\mathbf{b}_1, \ldots, \mathbf{b}_n$ are the vectors sent by $\mathcal{A}$ in the first round). Define

$$\Delta(\mathbf{a}_1, \ldots, \mathbf{a}_n) \stackrel{\text{def}}{=} f_{\mathcal{A}}(\mathbf{a}_1, \ldots, \mathbf{a}_n) \oplus f_{\text{correct}}(\mathbf{a}_1, \ldots, \mathbf{a}_n),$$

and say a query $\mathbf{a}_1, \ldots, \mathbf{a}_n$ is *good* if[5] $\mathbf{wt}(\Delta(\mathbf{a}_1, \ldots, \mathbf{a}_n)) \leq \mathsf{u}$. That is, a query $\mathbf{a}_1, \ldots, \mathbf{a}_n$ is good if $\mathcal{A}$'s response is within distance u of the "correct" response; i.e., $\mathcal{A}$ successfully impersonates the tag in response to such a query.

Let $\mathcal{D}$ denote the distribution over $\Delta(\mathbf{a}_1, \ldots, \mathbf{a}_n)$ induced by a uniform choice of a good query $\mathbf{a}_1, \ldots, \mathbf{a}_n$ (we assume at least one good query exists since we are only interested in analyzing this case). Note that, by definition of a good query, each vector in the support of $\mathcal{D}$ has weight at most u. Our goal is to show that with constant probability over Δ^1, Δ^2 generated according to $\mathcal{D}$, we have $\mathbf{wt}(\Delta^1 \oplus \Delta^2) < \mathsf{u}'$. We remark that this claim does not involve any assumptions regarding the probability that a randomly-chosen query is good.

To see how this maps on to the reduction being analyzed above, note that conditioning on the event that $\mathcal{A}$ successfully responds to queries $\mathbf{a}_1^1, \ldots, \mathbf{a}_n^1$ and $\mathbf{a}_1^2, \ldots, \mathbf{a}_n^2$ is equivalent to choosing

[4] Note that this analysis improves on what is claimed in [12].

[5] As in the proof of the previous theorem, the weight $\mathbf{wt}(Z)$ of a vector Z is the number of its entries equal to 1.

these two queries uniformly from the set of good queries. Setting $\Delta^1 \stackrel{\text{def}}{=} \Delta(\mathbf{a}_1^1, \ldots, \mathbf{a}_n^1)$ and Δ^2 analogously, we have

$$\begin{aligned}\Delta^1 \oplus \Delta^2 &= f_{\mathcal{A}}(\mathbf{a}_1^1, \ldots, \mathbf{a}_n^1) \oplus f_{\text{correct}}(\mathbf{a}_1^1, \ldots, \mathbf{a}_n^1) \oplus f_{\mathcal{A}}(\mathbf{a}_1^2, \ldots, \mathbf{a}_n^2) \oplus f_{\text{correct}}(\mathbf{a}_1^2, \ldots, \mathbf{a}_n^2) \\ &= Z^{\oplus} \oplus f_{\text{correct}}(\mathbf{a}_1^1, \ldots, \mathbf{a}_n^1) \oplus f_{\text{correct}}(\mathbf{a}_1^2, \ldots, \mathbf{a}_n^2).\end{aligned}$$

D cannot compute $f_{\text{correct}}(\mathbf{a}_1^1, \ldots, \mathbf{a}_n^1)$ or $f_{\text{correct}}(\mathbf{a}_1^2, \ldots, \mathbf{a}_n^2)$ since it does not know $\mathbf{s}_1$. However, it *can* compute

$$\begin{aligned}f_{\text{correct}}(\mathbf{a}_1^1, \ldots, \mathbf{a}_n^1) \oplus f_{\text{correct}}(\mathbf{a}_1^2, \ldots, \mathbf{a}_n^2) &= \left(\langle \mathbf{s}_1, \mathbf{b}_1 \rangle \oplus \left\langle \mathbf{s}_2, \mathbf{a}_1^1 \right\rangle, \ldots, \langle \mathbf{s}_1, \mathbf{b}_n \rangle \oplus \left\langle \mathbf{s}_2, \mathbf{a}_n^1 \right\rangle\right) \\ &\quad + \left(\langle \mathbf{s}_1, \mathbf{b}_1 \rangle \oplus \left\langle \mathbf{s}_2, \mathbf{a}_1^2 \right\rangle, \ldots, \langle \mathbf{s}_1, \mathbf{b}_n \rangle \oplus \left\langle \mathbf{s}_2, \mathbf{a}_n^2 \right\rangle\right) \\ &= \left(\left\langle \mathbf{s}_2, \mathbf{a}_1^1 \right\rangle \oplus \left\langle \mathbf{s}_2, \mathbf{a}_1^2 \right\rangle, \ldots, \left\langle \mathbf{s}_2, \mathbf{a}_n^1 \right\rangle \oplus \left\langle \mathbf{s}_2, \mathbf{a}_n^2 \right\rangle\right) \\ &= \left(\left\langle \mathbf{s}_2, (\mathbf{a}_1^1 \oplus \mathbf{a}_1^2) \right\rangle, \ldots, \left\langle \mathbf{s}_2, (\mathbf{a}_n^1 \oplus \mathbf{a}_n^2) \right\rangle\right) = \hat{Z}.\end{aligned}$$

We thus see that $Z^{\oplus}$ and $\hat{Z}$ differ in fewer than u' entries exactly when Δ^1 and Δ^2 differ in fewer than $\mathsf{u}' = \varepsilon^{++} n$ entries. It is the latter probability that we now analyze.

Let δ be a (positive) constant such that $\varepsilon^{++} = \frac{1}{2} \cdot (1 - \delta)$. Let $\gamma \stackrel{\text{def}}{=} 1 - 2\varepsilon^{+}$, and note that by our choice of ε^{++} we have $\delta < \gamma^2$. Set

$$c \stackrel{\text{def}}{=} \frac{1 - \delta}{\gamma^2 - \delta} + 1.$$

We show that for two vectors Δ^1, Δ^2 chosen independently according to distribution $\mathcal{D}$, we have $\mathbf{wt}(\Delta^1 \oplus \Delta^2) < \varepsilon^{++} n$ with (constant) probability at least $\frac{1}{c^2}$. Assume not. So

$$\Pr[\Delta^1, \Delta^2 \leftarrow \mathcal{D} : \mathbf{wt}(\Delta^1 \oplus \Delta^2) < \varepsilon^{++} n] < \frac{1}{c^2}.$$

But then, by a union bound,

$$\Pr[\Delta^1, \ldots, \Delta^c \leftarrow \mathcal{D} : \exists i \neq j \text{ s.t. } \mathbf{wt}(\Delta^1 \oplus \Delta^2) < \varepsilon^{++} n] < \frac{1}{2}.$$

In particular, there exist c vectors $\Delta^1, \ldots, \Delta^c$ in the support of $\mathcal{D}$ whose pairwise distances are all at least $\varepsilon^{++} n = \frac{1}{2} \cdot (1 - \delta) n$. Furthermore, each Δ^i has weight at most $\mathsf{u} = \frac{1}{2} \cdot (1 - \gamma) n$ since it lies in the support of $\mathcal{D}$. However, the Johnson bound [9, 10] (our notation was chosen to be consistent with the formulation in [5, Theorem 1]), which gives bounds on the size of constant-weight codes of certain minimum distance, shows that no such set $\{\Delta^i\}_{i=1}^{c}$ exists. ■

References

[1] E.R. Berlekamp, R.J. McEliece, and H.C.A. van Tilborg. On the Inherent Intractability of Certain Coding Problems. *IEEE Trans. Info. Theory* 24: 384–386, 1978.

[2] A. Blum, M. Furst, M. Kearns, and R. Lipton. Cryptographic Primitives Based on Hard Learning Problems. *Adv. in Cryptology — Crypto '93*, LNCS vol. 773, Springer-Verlag, pp. 278–291, 1994.

[3] A. Blum, A. Kalai, and H. Wasserman. Noise-Tolerant Learning, the Parity Problem, and the Statistical Query Model. *J. ACM* 50(4): 506–519, 2003.

[4] F. Chabaud. On the Security of Some Cryptosystems Based on Error-Correcting Codes. *Adv. in Cryptology — Eurocrypt '94*, LNCS vol. 950, Springer-Verlag, pp. 131–139, 1995.

[5] V. Guruswami and M. Sudan. Extensions to the Johnson Bound. Unpublished manuscript, 2001. Available at `http://citeseer.ist.psu.edu/guruswami01extensions.html`.

[6] J. Håstad. Some Optimal Inapproximability Results. *J. ACM* 48(4): 798–859, 2001.

[7] N. Hopper and M. Blum. A Secure Human-Computer Authentication Scheme. Technical Report CMU-CS-00-139, Carnegie Mellon University, 2000.

[8] N. Hopper and M. Blum. Secure Human Identification Protocols. *Adv. in Cryptology — Asiacrypt 2001*, LNCS vol. 2248, pp. 52–66, 2001.

[9] S.M. Johnson. A New Upper Bound for Error-Correcting Codes. *IEEE Trans. Info. Theory* 8: 203–207, 1962.

[10] S.M. Johnson. Improved Asympototic Bounds for Error-Correcting Codes. *IEEE Trans. Info. Theory* 9: 198–205, 1963.

[11] A. Juels and S. Weis. Authenticating Pervasive Devices with Human Protocols. *Adv. in Cryptology — Crypto 2005*, LNCS vol. 3621, Springer-Verlag, pp. 293–308, 2005. Updated version available at: http://www.rsasecurity.com/rsalabs/staff/bios/ajuels/publications/pdfs/lpn.pdf

[12] J. Katz and J.-S. Shin. Parallel and Concurrent Security of the HB and HB$^+$ Protocols. *Adv. in Cryptology — Eurocrypt 2006*.

[13] M. Kearns. Efficient Noise-Tolerant Learning from Statistical Queries. *J. ACM* 45(6): 983–1006, 1998.

[14] O. Regev. On Lattices, Learning with Errors, Random Linear Codes, and Cryptography. *37th ACM Symposium on Theory of Computing*, ACM, pp. 84–93, 2005.