HB^{N}: An HB-like protocol secure against Man-in-the-Middle Attacks

Carl Bosley* Kristiyan Haralambiev † Antonio Nicolosi ‡

June 28, 2011

Abstract

We construct a simple authentication protocol whose security is based solely on the problem of Learning Parity with Noise (LPN) that is secure against Man-in-the-Middle attacks. Our protocol is suitable for RFID devices, whose limited circuit size and power constraints rule out the use of more heavyweight operations such as modular exponentiation. The protocol is extremely simple: both parties compute a noisy bilinear function of their inputs. The proof, however, is quite technical, and we believe that some of our technical tools may be of independent interest.

1 Introduction

Motivation. Many cryptographic tasks originate from the necessity to reproduce in cyber space security properties that exist in the physical world. Examples in point include digital signatures (non-repudiation) or public-key encryption (drop-boxes). Among the basic cryptographic goals, authentication has the potential to straddle the physical and cyber world, and enable authentication cryptographically strong authentication of physical things.

For moderately powerful devices like smartphones, or even battery-operated sensors, existing authentication protocols often suffice. Computationally weak devices such as RFID devices and batteryless contactless smartcards, however, require more lightweight, dedicated solutions.

RFID devices are quickly becoming popular in many applications. They are used throughout the supply chain for inventory management. RFID can be used to replace physical keys for access control. Banking and financial institutions have also started to embrace them for account management. Mass transit authorities in several metropolitan areas have taken to used them to replace tokens; similarly, RFID-mediated access to toll roads is the norm all over the world.

RFID devices can do all this, silently. Unfortunately, this silence leaves them vulnerable to stealth queries from malicious entities. This introduces an array of security risks, including unauthorized access, fraudulent account usage, as well as privacy risks, such as stealth tracking.

Learning parity with noise. The LPN problem was introduced in the machine learning community by Angluin and Laird [AL87]. It soon became notorious for having no efficient noise-tolerant algorithm. It was proven by Kearns [Kea93] that the class of noisy parity concepts (LPN) is not learn-able within the statistical query model. Work on LPN-based protocols began with the HB protocol of Hopper and Blum [HB01], which was later proven to be secure against Passive attacks assuming the hardness of LPN.

HB-type protocols. The original motivation for the HB protocol was to enable unaided human authentication: the goal was for the protocol to be simple enough to be carried out without the help of a computational device. Subsequent work has found that the key sizes and error rates required to ensure security may be too

*Dept. of Computer Science, Stevens Institute. bosley@cs.stevens.edu.

†Dept. of Computer Science, New York University. haralambiev@cs.nyu.edu.

‡Dept. of Computer Science, Stevens Institute. nicolosi@cs.stevens.edu

large for humans to employ with ease comparable to, say, password-based authentication. Nevertheless, as noted by Juels and Weis [JW05], HB-type protocols are lightweight enough to be potentially applicable in the RFID setting. Indeed, constraints on power consumption and circuit size (1,000–4,000 transistors) for RFID devices makes it problematic to deploy conventional cryptographic algorithms like AES or modular exponentiation on these devices; HB-type protocols, on the other hand, have very simple circuit representations. For example, the interaction between the prover, or tag $\mathcal{T}$, and the verifier, or reader $\mathcal{R}$, in the HB protocol consists of two messages: first, $\mathcal{R}$ sends a random challenge $\mathbf{a} \in \mathbb{F}_2^n$. Next, $\mathcal{T}$ samples $e \in \mathbb{F}_2$ according to the Bernoulli distribution Ber_ε (i.e. $\Pr[e = 1] = \varepsilon$). $\mathcal{T}$ sends $z = \mathbf{a}^\top \mathbf{x} + e$ to $\mathcal{R}$, where $\mathbf{x} \in \mathbb{F}_2^n$ is a key shared between $\mathcal{T}$ and $\mathcal{R}$. $\mathcal{R}$ accepts if $z = \mathbf{a}^\top \mathbf{x}$. The basic protocol has soundness $\frac{1}{2}$ and completeness $1 - \varepsilon$, but this can be improved via sequential or parallel composition (*cf.* Section 2.3).

In [JW05], Juels and Weis also introduced HB^+, which was shown to be secure in a slightly stronger security model (known as Active security) than the original HB protocol. Gilbert, Robshaw, and Seurin ([GRS05]) showed that HB^+ is vulnerable to a man-in-the-middle attack. A number of variants of HB^+ were proposed to remedy this defect, including HB^{++} [BCD06], HB^* [DK08], HB-MP [MP07], HB-MP' [LMM08], and Trusted-HB [BC08]. However, all of these were proven insecure. Gilbert, Robshaw, and Seurin ([GRS08a]) extended their attack on HB^+ to break HB^{++}, HB^*, HB-MP, HB-MP', and Frumkin and Shamir [FS09] showed that Trusted-HB is insecure.

Gilbert, Robshaw, and Seurin [GRS08b] introduced $\mathsf{HB}^\#$, which was secure against the same attack that succeeded against HB^+. However, Oaufi et al [OOV08] presented an Man-in-the-Middle attack on $\mathsf{HB}^\#$.

Katz, Shin, and Smith [KSS10] provided the first proof of security for HB and HB^+ for any error rate $\varepsilon < 1/2$, via black box reductions. However, for HB^+ the reduction used rewinding, so that it achieved active security $\sqrt{\varepsilon}$ assuming LPN is hard for noise rate ε.

Pietrzak then introduced Subspace LWE [Pie10], a more flexible formulation of LPN that is nevertheless equivalent to LPN. In a major advance, Kiltz et al. [KPC+11] built on Subspace LWE [Pie10] to construct a two-round Active-secure protocol, as well as two secure MACs, which imply two-round Man-in-the-Middle-secure protocols. However, both Man-in-the-Middle-secure constructions require the use of an (almost) Pairwise Independent Permutation on approximately $O(n^2)$ bits. Furthermore, the first MAC's security reduction is loose, achieving security $\sqrt{\varepsilon}$, while the second construction is much more complicated and requires a longer key.

1.1 Our Contribution

Our protocol, like the original HB protocol, is extremely simple: instead of computing a noisy linear function $\mathbf{a}^\top \mathbf{x} + e$, the parties compute a noisy *bilinear* function $\mathbf{a}^\top \mathbf{X} \mathbf{b} + e$ of their joint inputs $\mathbf{a}, \mathbf{b}$. As described in Section 3, this can be done in either 2 or 3 rounds.

However, the Man-in-the-Middle security proof is quite technically involved, particularly in the understanding of the noise distributions. We develop some technical tools, including the LHN (Learning Halfspaces with Noise) problem, which we believe will be of independent interest.

1.2 Outline

We describe LPN, HB and HB^+, and the Passive, Active, and Man-in-the-Middle security models in Section 2. In Section 3, we describe the HB^N protocol family. In order to analyze the security of HB^N, we first need to develop new tools for precisely manipulating error distributions, including the LHN (Learning Halfspaces with Noise) problem, which we present in Section 4. Finally, in Section 5, we prove that HB^N is secure against Man-in-the-Middle attacks.

2 Preliminaries

2.1 Notation

We write $x \xleftarrow{\$} X$ to denote the process of assigning a value sampled from the distribution X to the variable x. If S is a finite set, we write $s \xleftarrow{\$} S$ to denote assignment to s of a value sampled from the uniform distribution on S. Vice versa, we will abuse set-notation to identify a distribution X with its support; for example, we write $x \in X$ to denote that x is in the support of X. If $\mathcal{A}$ is a probabilistic algorithm, we let $\mathcal{A}(x)$ denote the output

distribution of $\mathcal{A}$ on input x, and write $y \xleftarrow{\$} \mathcal{A}(x)$ to denote the process of running algorithm $\mathcal{A}$ on input x and assigning its output to y. We write:

$$\Pr[x_1 \xleftarrow{\$} X_1, x_2 \xleftarrow{\$} X_2(x_1), \ldots, x_n \xleftarrow{\$} X_n(x_1, \ldots, x_{n-1}) : \phi(x_1, \ldots, x_n)]$$

to denote the probability that the predicate $\phi(x_1, \ldots, x_n)$ is true, when for all $i \in \{1, \ldots, n\}$, x_i is drawn from distribution X_i, possibly depending on the values drawn for $x_1, \ldots, x_{i-1}$. When $n = 1$, $\hat{x} \in X_1$, and $\phi(x_1)$ is of the form "$x_1 = \hat{x}_1$", we use the shorthand $\Pr[\hat{x}_1 \xleftarrow{\$} X]$ to denote $\Pr[x_1 \xleftarrow{\$} X_1 : x_1 = \hat{x}_1]$. For two probability distributions X_1, X_2, we write $X_1 \equiv X_2$ if and only if $\forall \hat{x} \in X_1 \cup X_2, \Pr[\hat{x} \xleftarrow{\$} X_1] = \Pr[\hat{x} \xleftarrow{\$} X_2]$.

Let $\mathbb{F}_q$ represent the finite field with q elements. We denote the uniform distribution over $\mathbb{F}_2^n$ by $\mathsf{U}_{n \times n}$, and the Bernoulli distribution with bias ε by Ber_ε. (Recall that Ber_ε is the distribution over $\mathbb{F}_2$ with $\Pr[1 \xleftarrow{\$} \mathsf{Ber}_\varepsilon] = \varepsilon$, $\Pr[0 \xleftarrow{\$} \mathsf{Ber}_\varepsilon] = 1 - \varepsilon$.) We use the binary operator $\oplus \colon \mathbb{F}_2 \times \mathbb{F}_2 \to \mathbb{F}_2$ to represent finite field addition, and we let $\overline{b} = 1 \oplus b$ be the complement of b.

We denote column vectors by lower-case bold letters such as $\mathbf{x}$, and matrices by upper-case bold letters such as $\mathbf{X}$. We denote the transpose of $\mathbf{X}$ by $\mathbf{X}^\top$. For a matrix $\mathbf{A} \in \mathbb{F}_2^{m \times n}$, $\mathsf{rank}(\mathbf{A})$ denotes the rank of $\mathbf{A}$. $\ker(\mathbf{A}) = \{\mathbf{x} : \mathbf{A}\mathbf{x} = 0\}$ denotes the kernel of $\mathbf{X}$, the set of all vectors orthogonal to $\mathbf{A}$, and $\mathsf{Im}(\mathbf{A}) = \{\mathbf{y} : \exists \mathbf{x} \text{ s.t. } \mathbf{A}\mathbf{x} = \mathbf{y}\}$ denotes the image of $\mathbf{A}$, the set of all linear combinations of columns of $\mathbf{A}$.

We will often consider column vectors $\mathbf{x}, \mathbf{y} \in \mathbb{F}_2^\ell$ as matrices in $\mathbb{F}_2^{\ell \times 1}$. Considering $\mathbf{x}, \mathbf{y}$ as matrices allows us to extend operations on matrices to vectors. For example, we can form the outer product $\mathbf{x}\mathbf{y}^\top \in \mathbb{F}_2^{\ell \times \ell}$, and form the kernel $\ker(\mathbf{x})$. The dot product of two column vectors $\mathbf{x}, \mathbf{y}$ can be written as the matrix multiplication $\mathbf{x}^\top \mathbf{y}$. For a vector $\mathbf{x}$, we denote the scalar i-th element of $\mathbf{x}$ by $\mathbf{x}_i$. $\mathbf{0}^n$ denotes the all-zero column vector of length n. $\mathbf{e}^{(i,\ell)} \in \mathbb{F}_2^\ell$ denotes the i-th vector of the canonical basis, for which $\mathbf{e}_i^{(i)} = 1$, and $\mathbf{e}_j^{(i)} = 0$ for $j \neq i$. In practice, when the dimension can be determined from context, we drop it, letting $\mathbf{e}^{(i)} = \mathbf{e}^{(i,\ell)}$. For a vector $\mathbf{x}$, let $|\mathbf{x}|$ denote the number of nonzero entries of $\mathbf{x}$. We use $[n]$ to denote the set $\{1, 2, \ldots, n\}$.

We denote an arbitrary polynomial function of n by $\mathsf{poly}(n)$. We write $f = \mathsf{negl}$ to mean that f is negligible as a function of n, that is, $f = o(n^{-c})$ for any constant $c > 0$.

2.2 Learning Parity with Noise (LPN)

Roughly speaking, the problem of Learning Parity with Noise amounts to distinguishing two distributions over $\mathbb{F}_2^n \times \mathbb{F}_2$: the uniform distribution and the LPN *distribution*. For a random secret vector $\mathbf{x} \in \mathbb{F}_2^n$, the LPN distribution is in turn defined in terms of its sampling algorithm $\mathsf{LPN}_\varepsilon^{\mathbf{x}}$, shown in Algorithm 2.2. Algorithm $\mathsf{LPN}_\varepsilon^{\mathbf{x}}$ is initialized with a uniform secret vector $\mathbf{x} \xleftarrow{\$} \mathbb{F}_2^n$. Thereafter, whenever an LPN sample is requested, the algorithm chooses random $\mathbf{a} \xleftarrow{\$} \mathbb{F}_2^n$ and $e \xleftarrow{\$} \mathsf{Ber}_\varepsilon$ and outputs $(\mathbf{a}, b)$, where $b = \mathbf{a}^\top \mathbf{x} \oplus e$. For $\varepsilon = \frac{1}{2}$, LPN becomes the uniform distribution.

```
1: function LPN_ε^x
2:     a ←$ F_2^n
3:     e ←$ Ber_ε
4:     b = a^T x + e
5:     return (a, b)
```

Algorithm 1: LPN

We will use the decisional version of the LPN hardness assumption, which is defined using an indistinguishability game. It has been shown [KSS10] that hardness of the decisional version is equivalent (up to polynomial factors) to hardness of recovering the entire key. The decisional variant of LPN is hard if it is difficult to distinguish between an oracle with distribution $\mathsf{LPN}_\varepsilon^{\mathbf{x}}$ versus an oracle with a random distribution $\mathsf{U}_n \times \mathsf{U}_1$, which (by Corollary 8) can be represented as $\mathsf{LPN}_{1/2}^{\mathbf{x}}$. More formally, the advantage of an algorithm $\mathcal{A}$ against LPN for a given (ε, n) is defined using a game in which the adversary attempts to guess which oracle was selected:

Definition 1. *The decisional* LPN *assumption states that for all efficient adversaries* $\mathcal{A}$, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{LPN}}(\varepsilon, n) \leq \varepsilon_{\mathsf{LPN}} =$

negl, *where* $\mathsf{Adv}^{\mathsf{LPN}}_{\mathcal{A}}(\varepsilon, n)$ *is defined as*

$$\mathsf{Adv}^{\mathsf{LPN}}_{\mathcal{A}}(\varepsilon,n) = \left| \Pr\left[\begin{array}{l} \mathbf{x} \xleftarrow{\$} \mathbb{F}_2^n, b \xleftarrow{\$} \mathbb{F}_2, \\ \mathcal{O}_b = \begin{cases} \mathsf{LPN}^{\mathbf{x}}_{1/2} & \text{if } b = 0 \\ \mathsf{LPN}^{\mathbf{x}}_{\varepsilon}, & \text{if } b = 1 \end{cases}, \ : \tilde{b} = b \\ \tilde{b} \xleftarrow{\$} \mathcal{A}^{\mathcal{O}_b}(1^n) \end{array} \right] - \frac{1}{2} \right| \tag{1}$$

2.3 HB and HB$^+$ protocols

The HB, HB$^+$ protocols consist of $k = \mathsf{poly}(n)$ iterations of what is known as a "basic authentication step". The protocols are executed by two parties: the tag $\mathcal{T}$, who wishes to authenticate, and the reader $\mathcal{R}$, who verifies the tag. [1] The key for HB is a vector $\mathbf{x}$ of length n, where n is the security parameter. For HB$^+$, the key consists of two vectors $\mathbf{x}, \mathbf{y}$ of length n. For $i \in [k]$, $\mathbf{a}^{(i)}, \mathbf{b}^{(i)} \in \mathbb{F}_2^n$ are column vectors used in the execution. In HB, as shown in Figure 1, a tag $\mathcal{T}$ and a reader $\mathcal{R}$ share a random secret key $\mathbf{x} \in \mathbb{F}_2^n$. In the i-th round authentication step, the reader sends a random challenge $\mathbf{a}^{(i)} \in \mathbb{F}_2^n$ to the tag, and the tag replies with $\mathbf{z}_i = {\mathbf{a}^{(i)}}^\top \mathbf{x} \oplus \mathbf{e}_i$, where $\mathbf{e}_i \xleftarrow{\$} \mathsf{Ber}_\varepsilon$. HB$^+$ adds a second secret $\mathbf{y}$ and a third round, as shown in Figure 2.

$\mathcal{T}$		$\mathcal{R}$
$\mathbf{e}_i \xleftarrow{\$} \mathsf{Ber}_\varepsilon$	$\xleftarrow{\mathbf{a}^{(i)}}$	$\mathbf{a}^{(i)} \xleftarrow{\$} \mathbb{F}_2^n$
$\mathbf{z}_i = {\mathbf{a}^{(i)}}^\top \mathbf{x} \oplus \mathbf{e}_i$	$\xrightarrow{\mathbf{z}_i}$	$\mathbf{w}_i = {\mathbf{a}^{(i)}}^\top \mathbf{x}$
$\mathsf{verify}(\mathbf{z}, \mathbf{w}) = \lvert \mathbf{z} \oplus \mathbf{w} \rvert \overset{?}{\leq} k \cdot \mathsf{u}_{\mathsf{HB}}(\varepsilon)$		

Figure 1: HB

In both HB and HB$^+$, at the end of k rounds, $\mathcal{R}$ checks to see what fraction of answers $\mathbf{z}_i$ were correct. If more than $k \cdot \mathsf{u}(\varepsilon)$ are correct, for $\mathsf{u}(\varepsilon)$ some function of ε, then $\mathsf{verify}(\mathbf{z}, \mathbf{w})$ returns true, and the reader accepts. Otherwise, the reader rejects. k and $\mathsf{u}(\varepsilon)$ should be set high enough to allow the honest tag to authenticate w.h.p., but low enough that a malicious third party should not be able to authenticate by randomly guessing. In particular, as noted in [KSS10], for both HB and HB$^+$, $\mathsf{u}(\varepsilon) = (1 + \delta)\varepsilon$ suffices to achieve completeness error negligible in the security parameter, for any positive constant δ.

$\mathcal{T}$		$\mathcal{R}$
$\mathbf{a}^{(i)} \xleftarrow{\$} \mathbb{F}_2^n$	$\xleftarrow{\mathbf{b}^{(i)}}$	$\mathbf{b}^{(i)} \xleftarrow{\$} \mathbb{F}_2^n$
$\mathbf{e}_i \xleftarrow{\$} \mathsf{Ber}_\varepsilon$	$\xrightarrow{\mathbf{a}^{(i)}}$	$\mathbf{w}_i = {\mathbf{a}^{(i)}}^\top \mathbf{x} \oplus {\mathbf{b}^{(i)}}^\top \mathbf{y}$
$\mathbf{z}_i = {\mathbf{a}^{(i)}}^\top \mathbf{x} \oplus {\mathbf{b}^{(i)}}^\top \mathbf{y} \oplus \mathbf{e}_i$	$\xrightarrow{\mathbf{z}_i}$	
$\mathsf{verify}(\mathbf{z}, \mathbf{w}) = \lvert \mathbf{z} \oplus \mathbf{w} \rvert \overset{?}{\leq} k \cdot \mathsf{u}_{\mathsf{HB}^+}(\varepsilon)$		

Figure 2: HB$^+$

2.4 Security Models

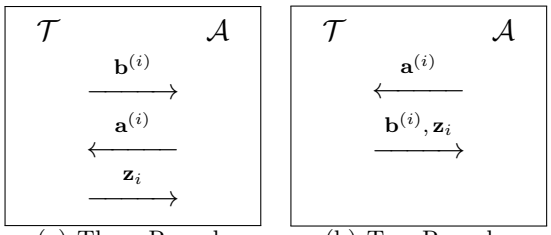
In this subsection we present several natural security models that have been used for authentication and for HB-type protocols in particular. The more general models are Passive, Active, and Man-in-the-Middle. Additionally,

[1] $\mathcal{T}$ is also known as the prover $\mathcal{P}$, and $\mathcal{R}$ as the verifier $\mathcal{V}$.

several works have used an intermediate model, GRS-MIM, which is stronger than Active yet weaker than the full Man-in-the-Middle model.

Passive **Model:** In Phase I, the attacker can only observe the interactions between $\mathcal{T}$ and $\mathcal{R}$.

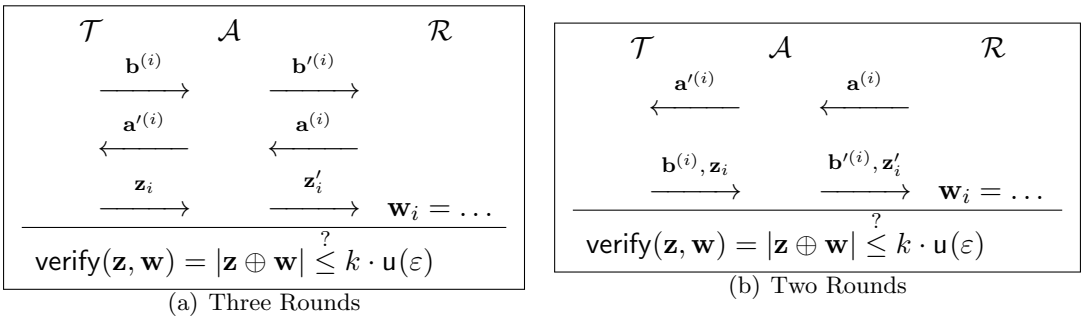
Active **Model:** In Phase I, as shown in Figure 3, the tag interacts with the attacker, who is free to choose non-random $\mathbf{a}$. However, $\mathbf{b}$ remains randomly chosen. Note that the attacker does not have access to a reader, and thus is unaware of the results of the reader's verification step.



(a) Three Rounds (b) Two Rounds

Figure 3: Active

Man-in-the-Middle **Model:** In Phase I, the attacker may eavesdrop on and modify any message, as shown in Figure 4. Additionally, the attacker learns the decisions made by the reader's verification step.

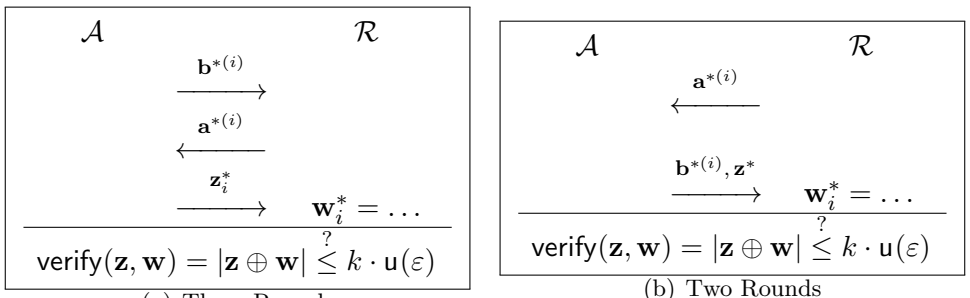


(a) Three Rounds (b) Two Rounds

Figure 4: Man-in-the-Middle

GRS-MIM **Model:** The GRS-MIM model of Gilbert, Robshaw, and Seurin [GRS08b] is a variant of the Man-in-the-Middle model, in which the adversary is not allowed to modify $\mathbf{z}_i$. That is, $\forall i, \mathbf{z}_i = \mathbf{z}'_i$. GRS-MIM includes the attack on HB^+, so that HB^+ is not secure in the GRS-MIM model. The restriction $\mathbf{z}_i = \mathbf{z}'_i$ is unrealistic in practice, but GRS-MIM was used by a number of recent works in an attempt to improve on HB^+, due to the difficulty of proving security in the full Man-in-the-Middle model. However, GRS-MIM-security does not imply Man-in-the-Middle-security, and indeed, GRS-MIM-secure protocols have been successfully attacked in the full model [OOV08].

Phase II. In all three models, the goal of the attacker $\mathcal{A}$ is to authenticate successfully to the reader $\mathcal{R}$ in k rounds of Phase II, as shown in Figure 5. $\mathcal{A}$ is successful iff $\mathsf{verify}(\mathbf{z})$ returns true and $\mathbf{b}^* \neq \mathbf{0}$ in all k rounds.



(a) Three Rounds

(b) Two Rounds

Figure 5: Phase II (All Models)

3 New protocols

We describe the HB^{N} family of "New HB" protocols. There are 2-round and 3-round variants. While HB is linear, our protocols are bilinear. Our secret key will be a matrix $\mathbf{X} \in \mathbb{F}_2^{n \times n}$. As before, $\mathbf{a}^{(i)}, \mathbf{b}^{(i)} \in \mathbb{F}_2^n$ are column vectors used in the execution. The protocol consists of the key generation step KeyGen and the authentication step Auth.

KeyGen. $\mathsf{KeyGen}(1^n)$ produces a matrix $\mathbf{X} \in \mathbb{F}_2^{n \times n}$.

- For the non-symmetric bilinear protocol, $\mathsf{KeyGen}_{\mathsf{HB}^{\mathsf{N}}}(1^n)$ returns a uniformly random matrix $\mathbf{X} \xleftarrow{\$} \mathbb{F}_2^{n \times n}$,
- For the symmetric version, $\mathsf{KeyGen.Symm}_{\mathsf{HB}^{\mathsf{N}}}(1^n)$ returns a random symmetric matrix $\mathbf{X} = \mathbf{X}^\top$. $\mathbf{X}$ can be computed, for example, by setting $\mathbf{X}_{i,j} = \mathbf{X}_{j,i} \xleftarrow{\$} \mathbb{F}_2$ for all (i, j) such that $0 \leq i \leq j < n$. Alternatively, $\mathsf{KeyGen.Symm}_{\mathsf{HB}^{\mathsf{N}}}(1^n)$ can set $\mathbf{X}' \xleftarrow{\$} \mathsf{KeyGen}_{\mathsf{HB}^{\mathsf{N}}}(1^n)$, and return $\mathbf{X}'\mathbf{X}'^\top$.

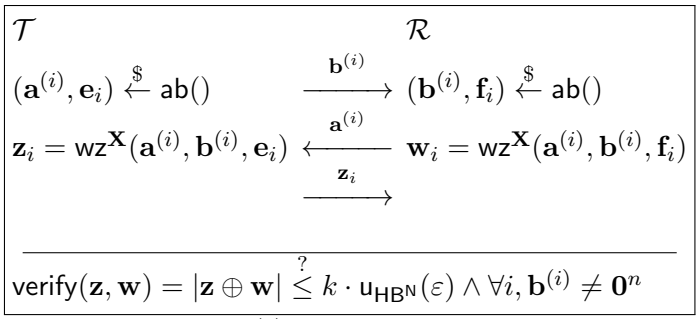
Auth. HB^{N} can be run in serial or in parallel. We describe the serial version first, and then modify the notation for the parallel version. The tag $\mathcal{T}_\varepsilon^{\mathbf{X}} = (\mathsf{Tb}(), \mathsf{Tz}^{\mathbf{X}}(\cdot,\cdot,\cdot))$ authenticates to the reader $\mathcal{R}_\varepsilon^{\mathbf{X}} = (\mathsf{Ra}(), \mathsf{Rw}^{\mathbf{X}}(\cdot,\cdot,\cdot))$ by performing k rounds of the protocol, as shown in Figure 6. Let $\mathsf{Ra}() = \mathsf{Tb}() = \mathsf{ab}()$, and $\mathsf{Rw}(\cdot,\cdot,\cdot) = \mathsf{Tz}(\cdot,\cdot,\cdot) = \mathsf{wz}(\cdot,\cdot,\cdot)$, as shown in Algorithm 2.

In each of k rounds, which can be executed in serial or in parallel, $\mathcal{T}_\varepsilon(\mathbf{X})$ draws $(\mathbf{b}^{(i)}, \mathbf{f}_i) \xleftarrow{\$} \mathsf{Tb}()$ and sends $\mathbf{b}^{(i)}$ to $\mathcal{R}_\varepsilon^{\mathbf{X}}$. Meanwhile, $\mathcal{R}$ draws $(\mathbf{a}^{(i)}, \mathbf{e}_i) \xleftarrow{\$} \mathsf{Ra}()$ and sends the challenge $\mathbf{a}^{(i)}$ to $\mathcal{T}_\varepsilon(\mathbf{X})$. Finally, $\mathcal{T}_\varepsilon(\mathbf{X})$ computes $\mathbf{z}_i = \mathsf{Tz}^{\mathbf{X}}(\mathbf{a}^{(i)}, \mathbf{b}^{(i)}, \mathbf{f}_i)$ and sends to $\mathcal{R}_\varepsilon^{\mathbf{X}}$. $\mathcal{R}_\varepsilon(\mathbf{X})$ computes $\mathbf{w}_i = \mathsf{Rw}^{\mathbf{X}}(\mathbf{a}^{(i)}, \mathbf{b}^{(i)}, \mathbf{f}_i)$. At the end of k rounds, $\mathcal{R}$ computes $|\mathbf{z} \oplus \mathbf{w}|$ and to determine what fraction of responses were correct. $\mathcal{R}$ also tests to ensure that $\forall i \in [k], \mathbf{b}^{(i)} \neq \mathbf{0}^n$. If all $\mathbf{b}^{(i)}$ are nonzero and more than $k \cdot \mathsf{u}_{\mathsf{HB}^{\mathsf{N}}}(\varepsilon) = k(1+\delta)(\varepsilon \oplus \varepsilon)$ for some completeness parameter δ, the reader accepts. [2]

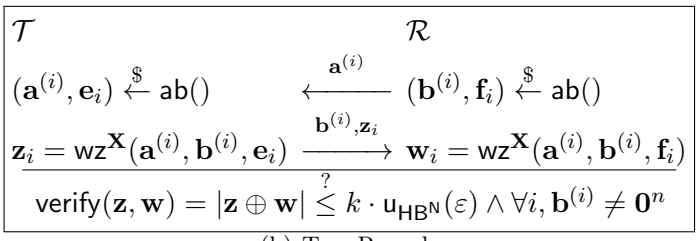
In the two-round version of HB^{N}, $\mathcal{R}$ sends $\mathbf{a}^{(i)}$ in the first round, and $\mathcal{T}$ sends $\mathbf{b}^{(i)}$ and $\mathbf{z}_i$ in the second round. In the three-round version, $\mathcal{T}$ sends $\mathbf{b}^{(i)}$ first, then receives $\mathbf{a}^{(i)}$ from $\mathcal{R}$, and finally sends $\mathbf{z}_i$ to $\mathcal{R}$.

Parallel version. We can use matrix notation to simplify working with HB^{N} in parallel, as shown in Figure 7 and Algorithm 3. Let $\mathbf{A}, \mathbf{B} \in \mathbb{F}_2^{n \times k}$ be matrices for which $\forall i \in [k], \mathbf{A}\mathbf{e}^{(i)} = \mathbf{a}^{(i)}, \mathbf{B}\mathbf{e}^{(i)} = \mathbf{b}^{(i)}$. That is, the columns of $\mathbf{A}, \mathbf{B}$ respectively are the vectors $\mathbf{a}^{(i)}, \mathbf{b}^{(i)}$ respectively. Then in the two-round version, for example, $\mathcal{R}$ sends the challenge $\mathbf{A} \xleftarrow{\$} \mathbb{F}_2^{n \times k}$. $\mathcal{T}$ replies with $\mathbf{B} \xleftarrow{\$} \mathbb{F}_2^{n \times k}$ and $\mathbf{z} = \mathsf{diag}(\mathbf{A}^\top \mathbf{X} \mathbf{B}) \oplus \mathbf{e}$, where $\mathbf{e} \xleftarrow{\$} \mathsf{Ber}_\varepsilon^n$. $\mathcal{R}$ computes $\mathbf{w} = \mathsf{diag}(\mathbf{A}^\top \mathbf{X} \mathbf{B}) \oplus \mathbf{f}$, where $\mathbf{f} \xleftarrow{\$} \mathsf{Ber}_\varepsilon^n$, and accepts iff $\forall i \in [k], \mathbf{B}\mathbf{e}^{(i)} \neq \mathbf{0}^n$ and $|\mathbf{z} \oplus \mathbf{w}| \leq \mathsf{u}_{\mathsf{HB}^{\mathsf{N}}}(\varepsilon)$.

[2] δ also governs the soundness of the protocol, which will be discussed in Section 5.5.

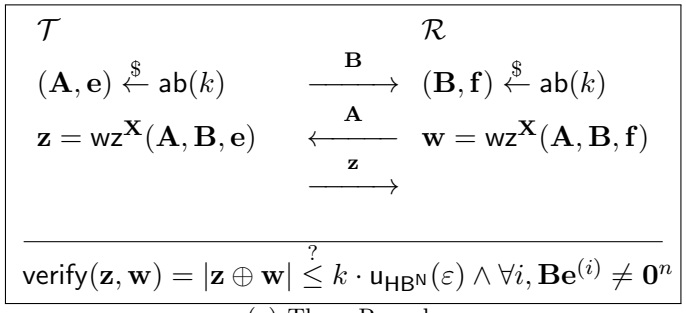


(a) Three Rounds

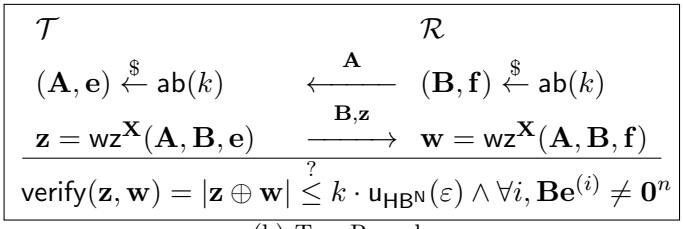


(b) Two Rounds

Figure 6: Protocols (Serial notation)



(a) Three Rounds



(b) Two Rounds

Figure 7: Protocols (Parallel notation)

1: **function** $\mathsf{wz}^{\mathbf{X}}(\mathbf{a}, \mathbf{b}, e)$
2: **return** $\mathbf{a}^\top \mathbf{X} \mathbf{b} \oplus e$

3: **function** $\mathsf{ab}()$
4: **return** $(\mathbf{a}, e) \xleftarrow{\$} (\mathsf{U}_n, \mathsf{Ber}_\varepsilon)$

Algorithm 2: Algorithms for HB^{N} (Serial notation)

1: **function** $\mathsf{wz}^{\mathbf{X}}(\mathbf{A}, \mathbf{B}, \mathbf{e})$
2: **return** $\mathsf{diag}(\mathbf{A}^\top \mathbf{X} \mathbf{B}) \oplus \mathbf{e}$

3: **function** $\mathsf{ab}(k)$
4: **return** $(\mathbf{A}, \mathbf{e}) \xleftarrow{\$} (\mathsf{U}_{n \times k}, \mathsf{Ber}_\varepsilon^k)$

Algorithm 3: Algorithms for HB^{N} (Parallel notation)

4 Learning Halfspaces with Noise (LHN)

Outline. In this section, we present a new conceptual tool in for analyzing HB-like protocol, the LHN (Learning Halfspaces with Noise) problem, as shown in Algorithm 4.3. The security of LHN is equivalent to that of LPN. First, in Section 4.1, we introduce a new (to our knowledge) compact notation for precisely working with sums of random variables over $\mathbb{F}_2$, in order to simplify working with LPN and LHN. Next, in Section 4.2, we establish several fundamental properties of LPN. We work with LHN itself in Section 4.3.

LHN. $\mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon}$ uses $\mathsf{LPN}^{\mathbf{x}}_{\varepsilon}$ to produce a biased halfspace distribution: $\mathbf{a}$ is chosen randomly subject to the condition that $\mathbf{a}^\top \mathbf{x}$ is distributed according to $\mathsf{Ber}_\rho \oplus \mathsf{Ber}_\varepsilon$. We also introduce the $\overline{\mathsf{LHN}}^{\mathbf{x}}_{\rho,\varepsilon}$ function, an alternative method of generating the LHN distribution which first chooses $\mathbf{a}$ and then b, and prove that the two distributions are equivalent.

4.1 Working with probability distributions of additive variables over $\mathbb{F}_2$

We will need to analyze sums of noise distributions. Our task will be made easier by the use of a compact and flexible notation describing our distributions. At the most basic level, we need to understand the sum of two different Bernoulli distributions, $\mathsf{Ber}_\varepsilon \oplus \mathsf{Ber}_\rho$. Intuitively, noise is additive, and bounded above by $\varepsilon+\rho$. However, it is also possible for errors to cancel. Indeed,

$$\begin{aligned}\Pr[1 \leftarrow \mathsf{Ber}_\varepsilon \oplus \mathsf{Ber}_\rho] &= \Pr[1 \leftarrow \mathsf{Ber}_\varepsilon \wedge 0 \leftarrow \mathsf{Ber}_\rho] + \Pr[0 \leftarrow \mathsf{Ber}_\varepsilon \wedge 1 \leftarrow \mathsf{Ber}_\rho] \\ &= \varepsilon(1-\rho) + \rho(1-\varepsilon) = \varepsilon + \rho - 2\rho\varepsilon \end{aligned} \tag{2}$$

We would like to define an operator that adds these distributions, in the same sense that $\oplus$ is the additive operator over $\mathbb{F}_2$. We can describe each distribution X by a single scalar, $\varepsilon_X = \Pr[X=1]$, with ε_X an element of the closed interval $[0,1]$. So, given $\oplus : \mathbb{F}_2 \times \mathbb{F}_2 \to \mathbb{F}_2$, we define an induced operator $\oplus^* : [0,1]\times[0,1] \to [0,1]$ which adds distributions:

$$\mathsf{Ber}_{\rho \oplus^* \varepsilon} = \mathsf{Ber}_\rho \oplus \mathsf{Ber}_\varepsilon$$

It follows from Equation 2 that for all $\rho, \varepsilon \in [0,1]$, $\oplus^*$ must satisfy $\rho \oplus^* \varepsilon = \varepsilon + \rho - 2\rho\varepsilon$. This is sufficient to uniquely define the operator. $\oplus^*$ acts similarly to the familiar binary operator $\oplus$: it is associative, commutative, and obeys the equalities $0 \oplus^* x = x$ and $1 \oplus^* x = 1 - x$ for all $x \in [0,1]$. For this reason, we drop the $*$ and simply refer to our operator as $\oplus$. We also observe that we can extend the complement operator to all of $[0,1]$, so that for all $\varepsilon \in [0,1]$, $\overline{\varepsilon} = 1 \oplus \varepsilon$. In summary, we have defined $\oplus, \bar{\cdot}$ so that

$$\begin{aligned}\forall \varepsilon \in [0,1],\ \overline{\varepsilon} \doteq 1 \oplus \varepsilon = 1 - \varepsilon \\ \forall \rho, \varepsilon \in [0,1],\ \rho \oplus \varepsilon \doteq \overline{\varepsilon}\rho + \varepsilon\overline{\rho} = (1-\varepsilon)\rho + (1-\rho)\varepsilon = \rho + \varepsilon - 2\rho\varepsilon\end{aligned}$$

Other useful facts about $\oplus$ over $[0,1]$ that we will use in the following are:

Fact 2. $\forall \varepsilon \in [0,1], \frac{1}{2} \oplus \varepsilon = \frac{1}{2}$.

Fact 3. $\forall \hat{b} \in \mathbb{F}_2, \Pr[e \xleftarrow{\$} \mathsf{Ber}_\varepsilon : e = \hat{b}] = \hat{b} \oplus \overline{\varepsilon} = \begin{cases} \varepsilon & \text{if } \hat{b} = 1 \\ 1-\varepsilon & \text{if } \hat{b} = 0 \end{cases}$.

The presence of the complement operator is due to the convention of parameterizing the Bernoulli distribution by $\Pr[\mathsf{Ber}_\varepsilon = 1] = \varepsilon$. If $\Pr[\mathsf{Ber}_\varepsilon = 0]$ was used instead, we would obtain the simpler expression $\hat{b} \oplus \varepsilon$. For this reason, we have chosen to complement the error term ε rather than the desired bit $\hat{b}$.

Fact 4. *Let* $\varepsilon^{\oplus n} = \overbrace{\varepsilon \oplus \varepsilon \oplus \ldots \oplus \varepsilon}^{n}$. *Then* $\varepsilon^{\oplus n} = \frac{1-(1-2\varepsilon)^n}{2}$.

Fact 4 tells us that noise behaves multiplicatively rather than additively. The reason it appears additive for small noise rates corresponds to the approximation $\exp(x) \approx 1 + x$ for small x. More precisely, the scaled distance from $\frac{1}{2}$ behaves multiplicatively:

Fact 5. *For all* $\delta, \tau \in [0,1]$, $\frac{1}{2}(1-\delta) \oplus \frac{1}{2}(1-\tau) = \frac{1}{2}(1-\delta\tau)$.

4.2 Learning Parity with Noise (LPN)

Next we establish a characterization of the LPN distribution in Lemma 6 and examine its consequences.

Lemma 6. $\forall (\hat{\mathbf{a}}, \hat{b}) \in \mathbb{F}_2^n \times \mathbb{F}_2, \Pr[(\hat{\mathbf{a}}, \hat{b}) \leftarrow \mathsf{LPN}_\varepsilon^{\mathbf{x}}] = (\hat{\mathbf{a}}^\top \mathbf{x} \oplus \hat{b} \oplus \overline{\varepsilon}) 2^{-n} = \begin{cases} \varepsilon 2^{-n} & \text{if } \hat{\mathbf{a}}^\top \mathbf{x} \neq \hat{b} \\ (1-\varepsilon)2^{-n} & \text{if } \hat{\mathbf{a}}^\top \mathbf{x} = \hat{b} \end{cases}$.

Proof. Since $\mathbf{a}, e$ are chosen independently, we have:

$$\begin{aligned}\Pr[(\hat{\mathbf{a}}, \hat{b}) \leftarrow \mathsf{LPN}_\varepsilon^{\mathbf{x}}] &= \Pr[(\mathbf{a}, b) \leftarrow \mathsf{LPN}_\varepsilon^{\mathbf{x}} : \mathbf{a} = \hat{\mathbf{a}}] \cdot \Pr[e \leftarrow \mathsf{Ber}_\varepsilon : e = \hat{\mathbf{a}}^\top \mathbf{x} \oplus \hat{b}] \\ &= \Pr[\hat{\mathbf{a}} \xleftarrow{\$} \mathbb{F}_2^n] \cdot \Pr[e \leftarrow \mathsf{Ber}_\varepsilon : e = \hat{\mathbf{a}}^\top \mathbf{x} \oplus \hat{b}] \\ &= 2^{-n}(\hat{\mathbf{a}}^\top \mathbf{x} \oplus \hat{b} \oplus \overline{\varepsilon}) \end{aligned} \tag{3}$$

Equation 3 follows from Fact 3. □

Summing over all $\hat{\mathbf{a}} \in \mathbb{F}_2^n$ yields the following corollary:

Corollary 7. $\forall \mathbf{x} \neq \mathbf{0}^n$, $\Pr[(\mathbf{a}, b) \leftarrow \mathsf{LPN}_\varepsilon^{\mathbf{x}} : b = 0] = \frac{1}{2}$.

Setting $\varepsilon = \frac{1}{2}$ in Lemma 6 and using Fact 2 yields the following corollary:

Corollary 8. $\Pr[(\hat{\mathbf{a}}, \hat{b}) \leftarrow \mathsf{LPN}_{1/2}^{\mathbf{x}}] = 2^{-n-1}$. *Equivalently,* $\mathsf{LPN}_{1/2}^{\mathbf{x}} \equiv \mathsf{U}_n \times \mathsf{U}_1$.

Finally, given any LPN distribution for any fixed key $\mathbf{x}$, we can produce an LPN distribution with a random key $\mathbf{z}$ and the same ε:

Corollary 9. *Let* $\mathbf{y} \xleftarrow{\$} \mathbb{F}_2^n$ *and let* $\mathbf{z} = \mathbf{x} \oplus \mathbf{y}$. *Then for* $(\mathbf{a}, b) \xleftarrow{\$} \mathsf{LPN}_\varepsilon^{\mathbf{x}}$, *the following distributions are equivalent:* $\mathsf{LPN}_\varepsilon^{\mathbf{x}} \oplus (\mathbf{0}^n, \mathbf{a}^\top \mathbf{y}) \equiv \mathsf{LPN}_\varepsilon^{\mathbf{z}}$.

Proof.

$$\begin{aligned}\Pr[(\hat{\mathbf{a}}, \hat{b}) \leftarrow \mathsf{LPN}_\varepsilon^{\mathbf{x}} \oplus (\mathbf{0}^n, \hat{\mathbf{a}}^\top \mathbf{y})] &= \Pr[(\hat{\mathbf{a}}, \hat{b} \oplus \hat{\mathbf{a}}^\top \mathbf{y}) \leftarrow \mathsf{LPN}_\varepsilon^{\mathbf{x}}] \\ &= 2^{-n}(\hat{\mathbf{a}}^\top \mathbf{x} \oplus \hat{\mathbf{a}}^\top \mathbf{y} \oplus \hat{b} \oplus \overline{\varepsilon}) \\ &= \Pr[(\hat{\mathbf{a}}, \hat{b}) \leftarrow \mathsf{LPN}_\varepsilon^{\mathbf{x} \oplus \mathbf{y}}]\end{aligned}$$

□

Corollary 9 says that we can "duplicate" an LPN distribution in some sense: we can use some of its samples as is, from the original distribution, and at the same time use the remaining samples as if they came from an entirely different LPN distribution with the same ε. Furthermore, if $\mathbf{x}$ is uniformly random, then so is $\mathbf{z}$.

4.3 Learning Halfspaces with Noise (LHN)

Next, we introduce LHN and $\overline{\mathsf{LHN}}$. In Lemma 13 we prove their equivalence and find a formula for their distribution. Finally, we establish a connection between hardness of $\mathsf{LHN}_{\rho,\varepsilon}^{\mathbf{x}}$ and $\mathsf{LPN}_\varepsilon^{\mathbf{x}}$.

function $\mathsf{LHN}_{\rho,\varepsilon}^{\mathbf{x}}$
  return $\mathsf{LHN}_\rho(\mathsf{LPN}_\varepsilon^{\mathbf{x}})$

function $\mathsf{LHN}_\rho(\mathsf{Samp})$
  $i = 0$
  $\tilde{b} \xleftarrow{\$} \mathsf{Ber}_\rho$
  repeat
    $(\mathbf{a}^{(i)}, b_i) \xleftarrow{\$} \mathsf{Samp}$
    $i \leftarrow i + 1$
  until $b_i = \tilde{b}$
  return $(\mathbf{a}^{(i)}, b_i)$

function $\Delta_{\rho,\varepsilon}(\hat{b}|b)$
  return $\dfrac{(\hat{b} \oplus \overline{\rho})(b \oplus \hat{b} \oplus \overline{\varepsilon})}{b \oplus \rho \oplus \overline{\varepsilon}}$

function $\overline{\mathsf{LHN}}_{\rho,\varepsilon}^{\mathbf{x}}$
  $(\mathbf{a}, b) \xleftarrow{\$} \mathsf{LHN}_{\varepsilon \oplus \rho, 0}^{\mathbf{x}}$
  $\delta = \Delta_{\rho,\varepsilon}(1|\mathbf{a}^\top \mathbf{x})$
  $\tilde{b} \xleftarrow{\$} \mathsf{Ber}_\delta$ ▷ $\Pr[\tilde{b} = \hat{b}] = \Delta_{\rho,\varepsilon}(\hat{b}|\mathbf{a}^\top \mathbf{x})$
  return $(\mathbf{a}, \tilde{b})$

Algorithm 4: LHN

The algorithm $\mathsf{LHN}_{\rho,\varepsilon}^{\mathbf{x}}$, shown in Algorithm 4.3, is constructed from the oracle $\mathsf{LPN}_\varepsilon^{\mathbf{x}}$. $\mathsf{LHN}_{\rho,\varepsilon}^{\mathbf{x}}$ first uses its own randomness to draw $\tilde{b} \xleftarrow{\$} \mathsf{Ber}_\rho$. Next, for $i \geq 0$ it repeatedly obtains $(\mathbf{a}^{(i)}, b_i) \xleftarrow{\$} \mathsf{LPN}_\varepsilon^{\mathbf{x}}$. The algorithm waits until $b_i = \tilde{b}$, and then outputs $(\mathbf{a}^{(i)}, b_i)$. $\overline{\mathsf{LHN}}_{\rho,\varepsilon}^{\mathbf{x}}$ is an alternative method which first chooses $(\mathbf{a}, b) \xleftarrow{\$} \mathsf{LHN}_{\varepsilon \oplus \rho, 0}$, discards b, and then samples $\tilde{b}$. Both algorithms run in expected polynomial time. Note that $\Delta_{\varepsilon,\varepsilon}(0|1) = \Delta_{\varepsilon,\varepsilon}(1|1) = \frac{1}{2}$. This implies $\mathsf{Ber}_{\Delta_{\varepsilon,\varepsilon}(b,1)} = \mathsf{Ber}_{\frac{1}{2}} = \mathsf{U}_1$, which will make $\overline{\mathsf{LHN}}_{\varepsilon,\varepsilon}$ useful in the security proof of HB^{N}.

Outline. For $(\mathbf{a}, b) \leftarrow \mathsf{LHN}_{\rho,\varepsilon}^{\mathbf{x}}$, we characterize the distribution of $\mathbf{a}$ in Lemma 10. The special case $\varepsilon = \frac{1}{2}$ is covered in Corollary 14. We prove in Corollary 11 that $(\mathbf{a}, b) \xleftarrow{\$} \overline{\mathsf{LHN}}_{\rho,\varepsilon}$ yields the same distribution for $\mathbf{a}$. Having proven that the distributions of the first n output bits are the same, in Lemma 13 we prove that the full distributions of $(\mathbf{a}, b)$ are also equivalent.

Lemma 10. *For $\mathbf{x} \neq 0$, $(\mathbf{a}, b) \xleftarrow{\$} \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon}$ produces the following distribution of $\mathbf{a}$:*

$$\Pr[(\mathbf{a}, b) \xleftarrow{\$} \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon} : \hat{\mathbf{a}} = \mathbf{a}] = (\hat{\mathbf{a}}^\top \mathbf{x} \oplus \rho \oplus \overline{\varepsilon}) 2^{-n+1} = \begin{cases} 2^{-n+1}(\rho \oplus \overline{\varepsilon}), & \text{if } \hat{\mathbf{a}}^\top \mathbf{x} = 0 \\ 2^{-n+1}(\rho \oplus \varepsilon), & \text{if } \hat{\mathbf{a}}^\top \mathbf{x} = 1 \end{cases}$$

Proof of Lemma 10. The algorithm $\mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon}$ progresses through a series of rounds, starting with round 0, until $\tilde{b}_i = b_i$. In each round, $\mathsf{LHN}_{\rho,\varepsilon}$ samples $(\mathbf{a}^{(i)}, b_i) \xleftarrow{\$} \mathsf{LPN}^{\mathbf{x}}_{\varepsilon}$. To model its distribution, we define a series of events. Let R be the event that $\tilde{b} = 0$. Let $S^{(i)}$ be the event that the algorithm terminates by returning $(\mathbf{a}^{(i)}, b_i)$ in round i, for $i \geq 0$. Finally, let $T^{(i)}_{(\hat{\mathbf{a}},\hat{b})}$ be the event that $(\hat{\mathbf{a}}, \hat{b}) \xleftarrow{\$} \mathsf{LPN}^{\mathbf{x}}_{\varepsilon}$ in round i. It follows that

$$\Pr[(\mathbf{a}, b) \xleftarrow{\$} \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon} : \mathbf{a} = \hat{\mathbf{a}}] = \sum_{i=0}^{\infty} \left(\Pr[S^{(i)}] \prod_{j<i} \Pr[\neg S^{(j)}] \right) \left(\Pr[R] \cdot \Pr[T^{(i)}_{(\hat{\mathbf{a}},0)}] + \Pr[\neg R] \cdot \Pr[T^{(i)}_{(\hat{\mathbf{a}},1)}] \right) \tag{4}$$

$$= \sum_{i=0}^{\infty} \left(\frac{1}{2} \right)^{i+1} \left((1-\rho) \Pr[T^{(i)}_{(\hat{\mathbf{a}},0)}] + \rho \Pr[T^{(i)}_{(\hat{\mathbf{a}},1)}] \right) \tag{5}$$

$$= (1-\rho) \Pr[T^{(0)}_{(\hat{\mathbf{a}},0)}] + \rho \Pr[T^{(0)}_{(\hat{\mathbf{a}},1)}] \tag{6}$$

$$= \begin{cases} ((1-\rho)(1-\varepsilon) + \rho\varepsilon) 2^{-n+1} & \text{if } \hat{\mathbf{a}}^\top \mathbf{x} = 0 \\ ((1-\rho)\varepsilon + \rho(1-\varepsilon)) 2^{-n+1} & \text{if } \hat{\mathbf{a}}^\top \mathbf{x} = 1 \end{cases} \tag{7}$$

$$= \begin{cases} (\rho \oplus \overline{\varepsilon}) 2^{-n+1} & \text{if } \hat{\mathbf{a}}^\top \mathbf{x} = 0 \\ (\rho \oplus \varepsilon) 2^{-n+1} & \text{if } \hat{\mathbf{a}}^\top \mathbf{x} = 1 \end{cases}$$

$$= (\hat{\mathbf{a}}^\top \mathbf{x} \oplus \rho \oplus \overline{\varepsilon}) 2^{-n+1}$$

Equation 4 follows from summing over all $i \geq 0$ and all bits $\hat{b} \in \mathbb{F}_2$ the probability that $\mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon}$ terminates in round i with output $(\hat{\mathbf{a}}, \hat{b})$. Equation 5 follows from Corollary 7 and $\Pr[R] = 1 - \rho$. Equation 6 follows from the geometric series formula and from $\Pr[T^{(i)}_{(\hat{\mathbf{a}},\hat{b})}] = \Pr[T^{(0)}_{(\hat{\mathbf{a}},\hat{b})}]$. Equation 7 follows from Lemma 6. □

Since $\overline{\mathsf{LHN}}_{\rho,\varepsilon}$ calls $\mathsf{LHN}_{\rho\oplus\varepsilon,0}$, we obtain the following corollary of Lemma 10.

Corollary 11. $\forall \hat{\mathbf{a}}, \Pr[(\mathbf{a}, b) \xleftarrow{\$} \overline{\mathsf{LHN}}^{\mathbf{x}}_{\rho,\varepsilon} : \mathbf{a} = \hat{\mathbf{a}}] = \Pr[(\mathbf{a}, b) \xleftarrow{\$} \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon} : \mathbf{a} = \hat{\mathbf{a}}]$.

Proof. Since $\overline{\mathsf{LHN}}^{\mathbf{x}}_{\rho,\varepsilon}$ obtains $\mathbf{a}$ from $\mathsf{LHN}^{\mathbf{x}}_{\rho\oplus\varepsilon,0}$, it follows that

$$\begin{aligned} \Pr[(\mathbf{a}, b) \xleftarrow{\$} \overline{\mathsf{LHN}}^{\mathbf{x}}_{\rho,\varepsilon} : \mathbf{a} = \hat{\mathbf{a}}] &= \Pr[(\mathbf{a}, b) \xleftarrow{\$} \mathsf{LHN}^{\mathbf{x}}_{\rho\oplus\varepsilon,0} : \mathbf{a} = \hat{\mathbf{a}}] \\ &= (\hat{\mathbf{a}}^\top \mathbf{x} \oplus \rho \oplus \varepsilon \oplus \overline{0}) 2^{-n+1} \\ &= (\hat{\mathbf{a}}^\top \mathbf{x} \oplus \rho \oplus \varepsilon \oplus 1) 2^{-n+1} && (8) \\ &= (\hat{\mathbf{a}}^\top \mathbf{x} \oplus \rho \oplus \overline{\varepsilon}) 2^{-n+1} && (9) \\ &= \Pr[(\mathbf{a}, b) \xleftarrow{\$} \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon} : \mathbf{a} = \hat{\mathbf{a}}] \end{aligned}$$

Equation 8 and Equation 9 follow from $\overline{x} = 1 \oplus x$. □

Lemma 12. $\forall \hat{b}$, $\Pr[(\hat{\mathbf{a}}, \hat{b}) \leftarrow \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon}] = \Pr[\hat{b} \leftarrow \mathsf{Ber}_\rho] \cdot \Pr[(\hat{\mathbf{a}}, \hat{b}) \leftarrow \mathsf{LHN}^{\mathbf{x}}_{\hat{b},\varepsilon}]$.

Proof. Note that Equation 6 expresses $\Pr[(\hat{\mathbf{a}}, \hat{b}) \xleftarrow{\$} \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon} : \mathbf{a} = \hat{\mathbf{a}}]$ as the sum of two terms, each of which expresses the probability that $(\hat{\mathbf{a}}, \hat{b}) \leftarrow \mathsf{LPN}^{\mathbf{x}}_{\varepsilon}$ for some $\hat{b}$. We rewrite Equation 6, indexing over $\hat{b}$, obtaining

$$\begin{aligned} \Pr[(\hat{\mathbf{a}}, \hat{b}) \xleftarrow{\$} \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon}] &= (\hat{b} \oplus \overline{\rho}) \Pr[T^{(0)}_{(\hat{\mathbf{a}},\hat{b})}] + (\hat{b} \oplus \rho) \Pr[T^{(0)}_{(\hat{\mathbf{a}},\overline{\hat{b}})}] \\ &= (\hat{b} \oplus \overline{\rho}) \Pr[T^{(0)}_{(\hat{\mathbf{a}},\hat{b})}] && (10) \\ &= \Pr[\hat{b} \leftarrow \mathsf{Ber}_\rho] \cdot \Pr[(\hat{\mathbf{a}}, \hat{b}) \leftarrow \mathsf{LHN}^{\mathbf{x}}_{\hat{b},\varepsilon}] && (11) \end{aligned}$$

Equation 10 follows from $\Pr[(\hat{\mathbf{a}}, \overline{\hat{b}}) \leftarrow \mathsf{LHN}^{\mathbf{x}}_{\hat{b},\varepsilon} = \Pr[\overline{\hat{b}}] \xleftarrow{\$} \mathsf{LPN}^{\mathbf{x}}_{\hat{b}} = 0$ for $\hat{b} \in \mathbb{F}_2$. Equation 11 follows from Lemma 10. □

Finally, we show the full equivalence of LHN and $\overline{\mathsf{LHN}}$.

Lemma 13. $\overline{\mathsf{LHN}}^{\mathbf{x}}_{\rho,\varepsilon} \equiv \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon}$. *In particular,* $\forall(\hat{\mathbf{a}}, \hat{b})$,

$$\Pr[(\hat{\mathbf{a}}, \hat{b}) \leftarrow \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon}] = \Pr[(\hat{\mathbf{a}}, \hat{b}) \leftarrow \overline{\mathsf{LHN}}^{\mathbf{x}}_{\rho,\varepsilon}] = (\hat{b} \oplus \overline{\rho})(\hat{b} \oplus \hat{\mathbf{a}}^\top \mathbf{x} \oplus \overline{\varepsilon}) 2^{-n+1}.$$

Proof of Lemma 13. For $\overline{\mathsf{LHN}}^{\mathbf{x}}_{\rho,\varepsilon}$, recall the definition of $\Delta_{\rho,\varepsilon}(\hat{b}|b)$ from Algorithm 4.3. It follows that

$$\begin{aligned}
\Pr[(\hat{\mathbf{a}}, \hat{b}) \xleftarrow{\$} \overline{\mathsf{LHN}}_{\rho,\varepsilon}] &= \Delta_{\rho,\varepsilon}(\hat{b}|\hat{\mathbf{a}}^\top \mathbf{x}) \cdot \Pr[(\mathbf{a}, b) \xleftarrow{\$} \overline{\mathsf{LHN}}_{\rho,\varepsilon} : \mathbf{a} = \hat{\mathbf{a}}] && (12)\\
&= \Delta_{\rho,\varepsilon}(\hat{b}|\hat{\mathbf{a}}^\top \mathbf{x}) \cdot (\hat{\mathbf{a}}^\top \mathbf{x} \oplus \rho \oplus \overline{\varepsilon}) 2^{-n+1} && (13)\\
&= \left(\frac{(\hat{b} \oplus \overline{\rho})(\hat{b} \oplus \hat{\mathbf{a}}^\top \mathbf{x} \oplus \overline{\varepsilon})}{\hat{\mathbf{a}}^\top \mathbf{x} \oplus \rho \oplus \overline{\varepsilon}} \right) \cdot (\hat{\mathbf{a}}^\top \mathbf{x} \oplus \rho \oplus \overline{\varepsilon}) 2^{-n+1} && (14)\\
&= (\hat{b} \oplus \overline{\rho})(\hat{b} \oplus \hat{\mathbf{a}}^\top \mathbf{x} \oplus \overline{\varepsilon}) 2^{-n+1}.\\
&= \Pr[\hat{b} \leftarrow \mathsf{Ber}_\rho] \cdot \Pr[(\mathbf{a}, b) \leftarrow \mathsf{LHN}^{\mathbf{x}}_{\hat{b},\varepsilon}] && (15)\\
&= \Pr[(\hat{\mathbf{a}}, \hat{b}) \leftarrow \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon}] && (16)
\end{aligned}$$

Equation 12 follows from the definition of $\overline{\mathsf{LHN}}_{\rho,\varepsilon}$ in Algorithm 4.3. Equation 13 follows from Lemma 10. Equation 14 follows from the definition of $\Delta_{\rho,\varepsilon}$ in Algorithm 4.3. Equation 15 follows from Fact 2 and Lemma 10. Equation 16 follows from Lemma 12. □

Since $\forall x, x \oplus \frac{1}{2} = \frac{1}{2}$, we obtain the following corollary.

Corollary 14. *For* $\mathbf{x} \neq 0$, $\Pr[(\hat{\mathbf{a}}, \hat{b}) \xleftarrow{\$} \mathsf{LHN}^{\mathbf{x}}_{\rho,\frac{1}{2}}] = (b \oplus \overline{\rho}) 2^{-n}$.

Hardness of LHN. Hardness of LHN can be defined using an indistinguishability game. More formally, the advantage of an algorithm $\mathcal{A}$ is defined using a game in which the adversary attempts to guess whether the oracle is $\mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon}$ or $\mathsf{U}_n \times \mathsf{Ber}_\varepsilon$, which is equivalent, by Corollary 14, to $\mathsf{LHN}^{\mathbf{x}}_{\rho,\frac{1}{2}}$.

$$\mathsf{Adv}^{\mathsf{LHN}}_{\mathcal{A}}(\rho, \varepsilon, n) = \left| \Pr\left[\begin{array}{l} \mathbf{x} \xleftarrow{\$} \mathsf{KG}, b \xleftarrow{\$} \mathbb{F}_2, \\ \mathcal{O}_b = \begin{cases} \mathsf{LHN}^{\mathbf{x}}_{\rho,\frac{1}{2}} & \text{if } b = 0 \\ \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon}, & \text{if } b = 1 \end{cases}, : \hat{b} = b \\ \hat{b} \xleftarrow{\$} \mathcal{A}^{\mathcal{O}_b}() \end{array} \right] - \frac{1}{2} \right| \tag{17}$$

Hardness of $\mathsf{LHN}(\rho, \varepsilon, n)$ and hardness of $\mathsf{LPN}(\varepsilon, n)$ are directly related:

Lemma 15. *For any* ρ, ε, *if there exists a probabilistic polynomial time adversary* $\mathcal{A}$ *achieving* $\mathsf{Adv}^{\mathsf{LHN}}_{\mathcal{A}}(\rho, \varepsilon, n) \geq \delta$, *then there exists a probabilistic polynomial time adversary* $\mathcal{B}$ *for which* $\mathsf{Adv}^{\mathsf{LPN}}_{\mathcal{B}}(\varepsilon, n) \geq \delta$.

Proof of Lemma 15. Let $\mathcal{B}^{\mathcal{O}} = \mathcal{A}^{\mathsf{LHN}_\rho(\mathcal{O})}$. That is, $\mathcal{B}$ runs $\mathcal{A}$ and gives $\mathcal{A}$ access to an oracle LHN_ρ applied to $\mathcal{B}$'s oracle $\mathcal{O}$. Since $\mathsf{LHN}_\rho(\mathsf{LPN}^{\mathbf{x}}_\varepsilon) \equiv \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon}$ and $\mathsf{LHN}_\rho(\mathsf{LPN}^{\mathbf{x}}_{1/2}) \equiv \mathsf{U}_n \times \mathsf{Ber}_\rho$ by Corollary 8, $\mathsf{Adv}^{\mathsf{LPN}}_{\mathcal{B}}(\varepsilon, n)$ can be expressed as

$$\begin{aligned}
\mathsf{Adv}^{\mathsf{LPN}}_{\mathcal{B}}(\varepsilon, n) &= \left| \Pr\left[\begin{array}{l} \mathbf{x} \xleftarrow{\$} \mathsf{KG}, b \xleftarrow{\$} \mathbb{F}_2, \\ \mathcal{O}_b = \begin{cases} \mathsf{LHN}^{\mathbf{x}}_{\rho,\frac{1}{2}} & \text{if } b = 0 \\ \mathsf{LHN}^{\mathbf{x}}_{\rho,\varepsilon}, & \text{if } b = 1 \end{cases}, : \hat{b} = b \\ \hat{b} \xleftarrow{\$} \mathcal{B}^{\mathcal{O}_b}() \end{array} \right] - \frac{1}{2} \right| \\
&= \mathsf{Adv}^{\mathsf{LHN}}_{\mathcal{A}}(\rho, \varepsilon, n).
\end{aligned}$$

□

We will not need the reverse direction, but it is possible to show that LHN for an n-bit secret is at least as hard as LPN with a secret of length $n-1$ using Subspace LWE [Pie10]. Thus, LHN and LPN are essentially equivalent up to a 1 bit change in secret length.

5 Proof of Man-in-the-Middle-security

Let $S_{\mathsf{HB^N}}$ be the event that the Reader accepts in the challenge phase of bilinear $\mathsf{HB^N}$, and similarly define $T_{\mathsf{HB^N}}$ for symmetric $\mathsf{HB^N}$. For any efficient adversary $\mathcal{A}$, we define the adversary's advantage, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{HB^N}} = \Pr[S_{\mathsf{HB^N}}]$, $\mathsf{Adv}_{\mathcal{A}}'^{\mathsf{HB^N}} = \Pr[T_{\mathsf{HB^N}}]$. Our main result will be the following.

Theorem 16. *For any efficient adversary $\mathcal{A}$,*

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{HB^N}} \leq 2\varepsilon_{\mathsf{LPN}} + \mathsf{negl}$$
$$\mathsf{Adv}_{\mathcal{A}}'^{\mathsf{HB^N}} \leq 2\varepsilon_{\mathsf{LPN}} + \mathsf{negl}$$

The proof of Theorem 16 will consist of several steps. First, in Section 5.1, we establish an initial indistinguishability result, Theorem 17. Roughly speaking, Theorem 17 shows that any efficient adversary cannot distinguish between "nearby" keys $\mathbf{X}_0, \mathbf{X}_1$. Next, in Section 5.2, we build on Theorem 17 to establish Theorem 25, which uses a sequence of games to show that the adversary cannot even distinguish between "faraway" keys $\mathbf{X}_0$ and $\mathbf{X}_m$. Then, in Section 5.3, we provide an alternative construction of the final game in the sequence, and present Theorem 23, which states that the adversary has negligible advantage in the final game. The proof of Theorem 23 will require Theorem 26, a technical result for understanding products of matrices, which we state and prove in Section 5.4. Finally, in Section 5.5, we prove Theorem 23 and Theorem 16.

5.1 Indistinguishability for nearby keys

Our first main result, Theorem 17, establishes that no efficient adversary $\mathcal{A}$ interacting with a Tag and Reader $\mathcal{T}, \mathcal{R}$ can distinguish (in Phase II) whether the parties share a uniformly random secret key $\mathbf{X}_0 \xleftarrow{\$} \mathbb{F}_2^{n\times n}$ or a nearby secret key $\mathbf{X}_1$ which differs by a rank 1 matrix, i.e. $\mathbf{X}_1 = \mathbf{X}_0 \oplus \mathbf{r}\mathbf{s}^\top$ for $\mathbf{r}, \mathbf{s} \xleftarrow{\$} \mathbb{F}_2^n$.

The proof uses the sequence of games technique. For the bilinear protocol, in A_0, $\mathcal{A}$ interacts in Phase I with $\mathcal{T}_\varepsilon^{\mathbf{X}_0}, \mathcal{R}_\varepsilon^{\mathbf{X}_0}$, which correctly execute the protocol using a uniformly random key $\mathbf{X}_0 \xleftarrow{\$} \mathbb{F}_2^{n\times n}$, and in Phase II with $\mathcal{R}_\varepsilon^{\mathbf{X}_0}$. In A_5, $\mathcal{A}$ interacts with $\mathcal{T}_\varepsilon^{\mathbf{X}_1}, \mathcal{R}_\varepsilon^{\mathbf{X}_1}$ in Phase I, and then with $\mathcal{T}_\varepsilon^{\mathbf{X}_0}$ in Phase II. In the sequence of intermediate games, $\mathcal{T}, \mathcal{R}$ initially share secret $\mathbf{X}_0$ but attempt to simulate the protocol for secret $\mathbf{X}_1$ in an attempt to force $\mathcal{A}$ to reveal information about $\mathbf{X}_0$ and hence $\mathbf{r}$.

Theorem 17 establishes that these two games, and also their symmetric variants, are computationally indistinguishable from each other. Let S_Γ be the event that the Reader accepts in the challenge phase of Game Γ. For any efficient adversary $\mathcal{A}$ and any game Γ, we define the advantage $\mathsf{Adv}_{\mathcal{A}}^{\Gamma} = \Pr[S_\Gamma]$. All games are shown in Figure 8.

Theorem 17. *For any efficient adversary $\mathcal{A}$,*

$$|\mathsf{Adv}_{\mathcal{A}}^{A_0} - \mathsf{Adv}_{\mathcal{A}}^{A_5}| \leq 2\varepsilon_{\mathsf{LPN}},$$
$$|\mathsf{Adv}_{\mathcal{A}}^{B_0} - \mathsf{Adv}_{\mathcal{A}}^{B_5}| \leq 2\varepsilon_{\mathsf{LPN}}.$$

Transitions between games. The proof of Theorem 17 is built from a sequence of games with three types of transitions. Two types of transitions are syntactically equivalent ways to rewrite the distributions, while one type of transition is computationally indistinguishable based on LPN. The three types are as follows.

Changing Sampling of $\mathbf{b}^{(i)}, \mathbf{a}^{(i)}, \mathbf{a}^{*(i)}$. The transitions between Games A_0-A_1 and Games A_4-A_5, and similarly Games B_0-B_1 and Games B_4-B_5, change how $\mathbf{a}^{(i)}, \mathbf{b}^{(i)}, \mathbf{a}^{*(i)}$ are sampled, using Lemma 15.

Rewriting the error term. In Games A_1-A_2 and Games A_3-A_4 we use Lemma 13 to convert back and forth between representing $\mathbf{a}^{(i)\top}\mathbf{r}$ and $\mathbf{b}^{(i)\top}\mathbf{s}$ as a function of $\mathbf{f}_i$ and $\mathbf{e}_i$, respectively; and representing $\mathbf{f}_i$ and $\mathbf{e}_i$ as functions of $\mathbf{a}^{(i)\top}\mathbf{r}, \mathbf{b}^{(i)\top}\mathbf{s}$ respectively. Likewise, in Games B_1-B_2 and Games B_3-B_4, we apply Lemma 13 to $\mathbf{a}^{(i)\top}\mathbf{r}$ and $\mathbf{b}^{(i)\top}\mathbf{r}$.

Switching $\mathbf{X}_0$ and $\mathbf{X}_1$. In Games A_2-A_3 and in Games B_2-B_3, we use Corollary 19 to perform the actual switch. Throughout the sequence of games, we first sample $\mathbf{X}_0 \xleftarrow{\$} \mathbb{F}_2^{n\times n}$ at random, then sample $\mathbf{r}, \mathbf{s}$, and compute $\mathbf{X}_1 = \mathbf{X}_0 \oplus \mathbf{r}\mathbf{s}^\top$. $\mathbf{X}_0$ is used as the key in the challenge phase, and $\mathbf{X}$ is used in the attack phase. For the first few games, $\mathbf{X} = \mathbf{X}_0$, and then we switch to $\mathbf{X} = \mathbf{X}_1$. For all games, Figure 8 lists changes between games, while the following are common to all games:

$$\mathbf{z}_i \xleftarrow{\$} \mathsf{wz}^{\mathbf{X}}(\mathbf{a}'^{(i)}, \mathbf{b}^{(i)}, \mathbf{e}_i)$$
$$\mathbf{w}_i \xleftarrow{\$} \mathsf{wz}^{\mathbf{X}}(\mathbf{a}^{(i)}, \mathbf{b}'^{(i)}, \mathbf{f}_i)$$
$$\mathcal{R}^* = \mathcal{R}^{\mathbf{X}_0} \Leftrightarrow \begin{cases} \mathbf{w}_i^* \xleftarrow{\$} \mathsf{wz}^{\mathbf{X}_0}(\mathbf{a}^{*(i)}, \mathbf{b}^{*(i)}, \mathbf{f}_i^*) \\ (\mathbf{a}^{*(i)}, \mathbf{f}_i^*) \xleftarrow{\$} (\mathsf{U}_n, \mathsf{Ber}_\varepsilon) \end{cases}$$

Game	$\mathbf{X}$	$(\mathbf{a}^{(i)}, \mathbf{f}_i) \xleftarrow{\$} \mathsf{Ra}()$	$(\mathbf{b}^{(i)}, \mathbf{e}_i) \xleftarrow{\$} \mathsf{Tb}()$
A_0	$\mathbf{X}_0$	$\mathsf{LHN}^{\mathbf{r}}_{\varepsilon,\frac{1}{2}}$	$\mathsf{LHN}^{\mathbf{s}}_{\varepsilon,\frac{1}{2}}$
A_1	$\mathbf{X}_0$	$\mathsf{LHN}^{\mathbf{r}}_{\varepsilon,\varepsilon}$	$\mathsf{LHN}^{\mathbf{s}}_{\varepsilon,\varepsilon}$
A_2	$\mathbf{X}_0$	$\overline{\mathsf{LHN}}^{\mathbf{r}}_{\varepsilon,\varepsilon}$	$\overline{\mathsf{LHN}}^{\mathbf{s}}_{\varepsilon,\varepsilon}$
A_3	$\mathbf{X}_1$	$\overline{\mathsf{LHN}}^{\mathbf{r}}_{\varepsilon,\varepsilon}$	$\overline{\mathsf{LHN}}^{\mathbf{s}}_{\varepsilon,\varepsilon}$
A_4	$\mathbf{X}_1$	$\mathsf{LHN}^{\mathbf{r}}_{\varepsilon,\varepsilon}$	$\mathsf{LHN}^{\mathbf{s}}_{\varepsilon,\varepsilon}$
A_5	$\mathbf{X}_1$	$\mathsf{LHN}^{\mathbf{r}}_{\varepsilon,\frac{1}{2}}$	$\mathsf{LHN}^{\mathbf{s}}_{\varepsilon,\frac{1}{2}}$

(a) Non-symmetric

Game	$\mathbf{X}$	$\mathsf{Ra}() = \mathsf{Tb}()$
B_0	$\mathbf{X}_0$	$\mathsf{LHN}^{\mathbf{r}}_{\varepsilon,\frac{1}{2}}$
B_1	$\mathbf{X}_0$	$\mathsf{LHN}^{\mathbf{r}}_{\varepsilon,\varepsilon}$
B_2	$\mathbf{X}_0$	$\overline{\mathsf{LHN}}^{\mathbf{r}}_{\varepsilon,\varepsilon}$
B_3	$\mathbf{X}_1$	$\overline{\mathsf{LHN}}^{\mathbf{r}}_{\varepsilon,\varepsilon}$
B_4	$\mathbf{X}_1$	$\mathsf{LHN}^{\mathbf{r}}_{\varepsilon,\varepsilon}$
B_5	$\mathbf{X}_1$	$\mathsf{LHN}^{\mathbf{r}}_{\varepsilon,\frac{1}{2}}$

(b) Symmetric

Figure 8: Summary of Games

Theorem 17 will follow from Corollary 18, Corollary 19, and Corollary 21, which we now state and prove.

Corollary 18. *The following games are equivalent:*

$$|\Pr[S_{A_1}] - \Pr[S_{A_2}]| = |\Pr[S_{A_3}] - \Pr[S_{A_4}]| = 0,$$
$$|\Pr[S_{B_1}] - \Pr[S_{B_2}]| = |\Pr[S_{B_3}] - \Pr[S_{B_4}]| = 0.$$

Proof. For the non-symmetric case, we apply Lemma 13 with $(\rho, \mathbf{x}) = (\varepsilon, \mathbf{r})$ to $\mathsf{Ra}()$, and with $(\rho, \mathbf{x}) = (\varepsilon, \mathbf{s})$ to $\mathsf{Tb}()$. As a result, we find that $\mathsf{Ra}()$ and $\mathsf{Tb}()$ produce identical distributions in Games A_1-A_2 and Games A_3-A_4. Similarly, applying Lemma 13 with $(\rho, \mathbf{x}) = (\varepsilon, \mathbf{r})$ to both $\mathsf{Ra}()$ and $\mathsf{Tb}()$ yields equivalence of Games B_1-B_2 and Games B_3-B_4. □

Next we show that the games involving swapping keys are equivalent:

Corollary 19. $|\Pr[S_{A_2}] - \Pr[S_{A_3}]| = 0$, $|\Pr[S_{B_2}] - \Pr[S_{B_3}]| = 0$

The proof of Corollary 19 requires a technical lemma.

Lemma 20. *Given any* $\hat{\mathbf{b}} \in \mathbb{F}_2^n, \mathbf{X}_0 \in \mathbb{F}_2^{n\times n}$, *let* $\mathbf{x} \xleftarrow{\$} \mathbb{F}_2^n$, $\mathbf{X}_1 \xleftarrow{\$} \mathbf{X}_0 \oplus \mathbf{x}\mathbf{y}^\top$ *and* $(\mathbf{a}, e) \xleftarrow{\$} \mathsf{LHN}(\mathbf{x})$. *Then for any* $\mathbf{y} \in \mathbb{F}_2^n$, $\mathsf{wz}^{\mathbf{X}_0}(\mathbf{a}, \hat{\mathbf{b}}, e) \equiv \mathsf{wz}^{\mathbf{X}_1}(\mathbf{a}, \hat{\mathbf{b}}, e)$.

Proof.

$$\begin{aligned}
\mathsf{wz}^{\mathbf{X}_1}(\mathbf{a},\mathbf{b}',e) &= \mathbf{a}^\top\mathbf{X}_1\mathbf{b}' \oplus \mathsf{Ber}_{\Delta_{\varepsilon,\varepsilon}(1|\mathbf{a}^\top\mathbf{x})} \\
&= \mathbf{a}^\top\mathbf{X}_0\mathbf{b}' \oplus \mathbf{a}^\top\mathbf{x}\mathbf{y}^\top\mathbf{b}' \oplus \mathsf{Ber}_{\Delta_{\varepsilon,\varepsilon}(1|\mathbf{a}^\top\mathbf{x})} \\
&= \mathbf{a}^\top\mathbf{X}_0\mathbf{b}' \oplus \begin{cases} \mathsf{Ber}_{\Delta_{\varepsilon,\varepsilon}(1|0)} & \text{if } \mathbf{a}^\top\mathbf{x}=0 \\ \mathsf{Ber}_{\Delta_{\varepsilon,\varepsilon}(1|1)} \oplus \mathbf{y}^\top\mathbf{b}' & \text{if } \mathbf{a}^\top\mathbf{x}=1 \end{cases} \\
&= \mathbf{a}^\top\mathbf{X}_0\mathbf{b}' \oplus \begin{cases} \mathsf{Ber}_{\Delta_{\varepsilon,\varepsilon}(1|0)} & \text{if } \mathbf{a}^\top\mathbf{x}=0 \\ \mathsf{U}_1 & \text{if } \mathbf{a}^\top\mathbf{x}=1 \end{cases} \qquad (18) \\
&= \mathbf{a}^\top\mathbf{X}_0\mathbf{b}' \oplus \begin{cases} \mathsf{Ber}_{\Delta_{\varepsilon,\varepsilon}(1|0)} & \text{if } \mathbf{a}^\top\mathbf{x}=0 \\ \mathsf{Ber}_{\Delta_{\varepsilon,\varepsilon}(1|1)} & \text{if } \mathbf{a}^\top\mathbf{x}=1 \end{cases} \\
&= \mathbf{a}^\top\mathbf{X}_0\mathbf{b}' \oplus \mathsf{Ber}_{\Delta_{\varepsilon,\varepsilon}(1|\mathbf{a}^\top\mathbf{x})} \\
&= \mathsf{wz}^{\mathbf{X}_0}(\mathbf{a},\mathbf{b}',e)
\end{aligned}$$

Equation 18 follows from Fact 2 and $\Delta_{\varepsilon,\varepsilon}(b|1) = \frac{1}{2}$. □

The lemma uses the LHN distribution [3] to annihilate the adversary's contribution $\mathbf{y}^\top\mathbf{b}'$ to $\mathbf{w}$ corresponding to $\mathbf{x}\mathbf{y}^\top$. Corollary 19 then follows from several applications of Lemma 20.

Proof of Corollary 19. For Games A_2-A_3, we apply the lemma for $\mathbf{X}_0$ with $(\mathbf{x},\mathbf{y}) = (\mathbf{r},\mathbf{s})$ and $(\mathbf{a},\mathbf{b}') = (\mathbf{a}^{(i)},\mathbf{b}'^{(i)})$, which yields equivalence for $\mathsf{Rw}^{\mathbf{X}}(\mathbf{a}^{(i)},\mathbf{b}'^{(i)},\mathbf{f}_i) = \mathsf{wz}(\mathbf{a}^{(i)},\mathbf{b}'^{(i)},\mathbf{f}_i)$. Next we apply the lemma for $\mathbf{X}_0^\top$ with $(\mathbf{x},\mathbf{y}) = (\mathbf{s},\mathbf{r})$ and $(\mathbf{a},\mathbf{b}') = (\mathbf{b}^{(i)},\mathbf{a}'^{(i)})$, which yields equivalence for $\mathsf{Tz}^{\mathbf{X}}(\mathbf{a}'^{(i)},\mathbf{b}^{(i)},\mathbf{e}_i) = \mathsf{wz}^{\mathbf{X}^\top}(\mathbf{b}^{(i)},\mathbf{a}'^{(i)},\mathbf{e}_i)$. For Games B_2-B_3, the result follows from applying Lemma 20 for $\mathbf{X}_0 = \mathbf{X}_0^\top$ with $(\mathbf{x},\mathbf{y}) = (\mathbf{r},\mathbf{r})$ and $(\mathbf{a},\mathbf{b}')$ equal to $(\mathbf{a}^{(i)},\mathbf{b}'^{(i)})$ and $(\mathbf{b}^{(i)},\mathbf{a}'^{(i)})$ respectively. □

Finally, we consider the remaining games in Corollary 21.

Corollary 21. *The following games are computationally indistinguishable:*

$$\begin{aligned}
|\Pr[S_{A_0}] - \Pr[S_{A_1}]|, |\Pr[S_{A_4}] - \Pr[S_{A_5}]| &\le \varepsilon_{\mathsf{LPN}}, \\
|\Pr[S_{B_0}] - \Pr[S_{B_1}]|, |\Pr[S_{B_4}] - \Pr[S_{B_5}]| &\le \varepsilon_{\mathsf{LPN}}.
\end{aligned}$$

Corollary 21 will follow from Lemma 22.

Lemma 22. *Given a challenge oracle* $\mathcal{O}_b = \begin{cases} \mathsf{LPN}^{\mathbf{r}}_{1/2} & \text{if } b=0 \\ \mathsf{LPN}^{\mathbf{r}}_{\varepsilon}, & \text{if } b=1 \end{cases}$*, we can construct two separate challenge oracles,*

$$(\mathcal{O}_b^{(1)},\mathcal{O}_b^{(2)}) = \begin{cases} (\mathsf{LPN}^{\mathbf{r}}_{1/2},\mathsf{LPN}^{\mathbf{s}}_{1/2}) & \text{if } b=0 \\ (\mathsf{LPN}^{\mathbf{r}}_{\varepsilon},\mathsf{LPN}^{\mathbf{s}}_{\varepsilon}) & \text{if } b=1 \end{cases}$$

Proof. The lemma follows from applying Corollary 9 once for $b=0$ and once for $b=1$, setting $\mathbf{x}=\mathbf{r},\mathbf{z}=\mathbf{s}$ in both cases. □

Proof of Corollary 21. We define four games, as shown in Figure 9. In each game $\alpha_{\Gamma_1,\Gamma_2}$, the adversary $\mathcal{A}$ interacts with the tag $\mathcal{T}_{\Gamma_1,\Gamma_2}$, reader $\mathcal{R}_{\Gamma_1,\Gamma_2}$, and Phase II reader $\mathcal{R}^*_{\Gamma_1,\Gamma_2}$. Let $\mathcal{T}^{\mathbf{X}}_{\Gamma_1,\Gamma_2} = (\mathsf{Tb}(),\mathsf{Tz}^{\mathbf{X}}(\cdot,\cdot,\cdot))$, $\mathcal{R}^{\mathbf{X}}_{\Gamma_1,\Gamma_2} = (\mathsf{Ra}(),\mathsf{Rw}^{\mathbf{X}}(\cdot,\cdot,\cdot))$, where $\mathsf{Rw}(\cdot,\cdot,\cdot) = \mathsf{Tz}(\cdot,\cdot,\cdot) = \mathsf{wz}(\cdot,\cdot,\cdot)$, and $\mathbf{X},\mathsf{Ra}(),\mathsf{Tb}()$ are as specified in Figure 9. For all $\alpha_{\Gamma_1,\Gamma_2},\beta_{\Gamma_1,\Gamma_2}$ respectively, let $\mathcal{R}^*_{\Gamma_1,\Gamma_2} = \mathcal{R}^{*\mathbf{X}_0}$ be identical to the reader in A_0, B_0 respectively.

[3] This key step is actually the raison d'être of LHN.

Game	$\mathbf{X}$	$\mathsf{Ra}()$	$\mathsf{Tb}()$
$\alpha_{\mathrm{A}_0,\mathrm{A}_1}$	$\mathbf{X}_1 \xleftarrow{\$} \mathsf{KeyGen}_{\mathsf{HB}^{\mathsf{N}}}(1^n) \oplus \mathbf{r}\mathbf{s}^\top$	$\mathcal{O}_b^{(1)}$	$\mathcal{O}_b^{(2)}$
$\alpha_{\mathrm{A}_4,\mathrm{A}_5}$	$\mathbf{X}_0 \xleftarrow{\$} \mathsf{KeyGen}_{\mathsf{HB}^{\mathsf{N}}}(1^n)$	$\mathcal{O}_{\bar{b}}^{(1)}$	$\mathcal{O}_{\bar{b}}^{(2)}$
$\beta_{\mathrm{B}_0,\mathrm{B}_1}$	$\mathbf{X}_1 \xleftarrow{\$} \mathsf{KeyGen.Symm}_{\mathsf{HB}^{\mathsf{N}}}(1^n) \oplus \mathbf{r}\mathbf{r}^\top$	$\mathsf{LHN}_\varepsilon(\mathcal{O}_b)$	$\mathsf{LHN}_\varepsilon(\mathcal{O}_b)$
$\beta_{\mathrm{B}_4,\mathrm{B}_5}$	$\mathbf{X}_0 \xleftarrow{\$} \mathsf{KeyGen.Symm}_{\mathsf{HB}^{\mathsf{N}}}(1^n)$	$\mathsf{LHN}_\varepsilon(\mathcal{O}_{\bar{b}})$	$\mathsf{LHN}_\varepsilon(\mathcal{O}_{\bar{b}})$

Figure 9: Interpolating Games

We define the generic advantage of an adversary against a tag $\mathcal{T}$, reader $\mathcal{R}$, and challenge phase reader $\mathcal{R}^*$:

$$\mathsf{Adv}_{\mathcal{A}}^{(\mathcal{T},\mathcal{R},\mathcal{R}^*)} = \left| \Pr \left[\begin{array}{l} \mathbf{X} \xleftarrow{\$} \mathsf{KeyGen}, \\ s \xleftarrow{\$} \mathcal{A}_1^{\mathcal{T}^{\mathbf{X}},\mathcal{R}^{\mathbf{X}}}(1^n), \\ \mathbf{A}^* \xleftarrow{\$} \mathcal{R}_1^{*\mathbf{X}}, \\ (\mathbf{z}^*, \mathbf{B}^*) \xleftarrow{\$} \mathcal{A}_2(s, \mathbf{A}^*) \\ \mathbf{w}^* \xleftarrow{\$} \mathcal{R}_2^{*\mathbf{X}}(\mathbf{B}^*) \end{array} : \begin{array}{l} \forall i \in [k], \hat{\mathbf{B}}\mathbf{e}^{(i)} \neq \mathbf{0}^n, \\ |\mathbf{z}^* \oplus \mathbf{w}^*| \leq k \cdot \mathsf{u}_{\mathsf{HB}^{\mathsf{N}}}(\varepsilon) \end{array} \right] \right|$$

Let $\mathcal{T}_\Gamma, \mathcal{R}_\Gamma, \mathcal{R}_\Gamma^*$ be the Tag, Reader, and Phase II Reader in Game Γ. Consider the adversary $\mathcal{B}_{\Gamma_1,\Gamma_2}$ defined as follows. $\mathcal{B}_{\Gamma_1,\Gamma_2}(\mathcal{O})$ constructs game $\alpha_{\Gamma_1,\Gamma_2}$ from its oracle via Lemma 22. It runs $\mathcal{A}$, returning 1 if $\mathcal{A}$ is accepted (i.e. $|\mathbf{z}^* \oplus \mathbf{w}^*| \leq k \cdot \mathsf{u}_{\mathsf{HB}^{\mathsf{N}}}(\varepsilon)$), and 0 otherwise. Then by Lemma 22,

$$\begin{aligned} \varepsilon_{\mathsf{LPN}} &\geq \mathsf{Adv}_{\mathcal{B}}^{\mathsf{LPN}} \\ &= |\mathsf{Adv}_{\mathcal{A}}^{(\mathcal{T}_{\Gamma_1},\mathcal{R}_{\Gamma_1},\mathcal{R}^*_{\Gamma_1})} - \mathsf{Adv}_{\mathcal{A}}^{(\mathcal{T}_{\Gamma_2},\mathcal{R}_{\Gamma_2},\mathcal{R}^*_{\Gamma_2})}| \\ &= |\mathsf{Adv}_{\mathcal{A}}^{\Gamma_1} - \mathsf{Adv}_{\mathcal{A}}^{\Gamma_2}| \end{aligned}$$

The same argument establishes that $\varepsilon_{\mathsf{LPN}} \geq |\mathsf{Adv}_{\mathcal{A}}^{\Gamma_1} - \mathsf{Adv}_{\mathcal{A}}^{\Gamma_2}|$ for the $\beta_{\Gamma_1,\Gamma_2}$, except that we replace Lemma 22 by Lemma 15 for the construction of $\mathsf{Ra}()$ for $\mathcal{T}$ and $\mathsf{Tb}()$ for $\mathcal{R}$. □

Proof of Theorem 17. It follows from the triangle inequality and the above corollaries that

$$|\mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_0} - \mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_5}| \leq \sum_{i\in[5]} |\mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_{i-1}} - \mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_i}| \leq \varepsilon_{\mathsf{LPN}} + 0 + 0 + 0 + \varepsilon_{\mathsf{LPN}} = 2\varepsilon_{\mathsf{LPN}},$$

$$|\mathsf{Adv}_{\mathcal{A}}^{\mathrm{B}_0} - \mathsf{Adv}_{\mathcal{A}}^{\mathrm{B}_5}| \leq \sum_{i\in[5]} |\mathsf{Adv}_{\mathcal{A}}^{\mathrm{B}_{i-1}} - \mathsf{Adv}_{\mathcal{A}}^{\mathrm{B}_i}| \leq \varepsilon_{\mathsf{LPN}} + 0 + 0 + 0 + \varepsilon_{\mathsf{LPN}} = 2\varepsilon_{\mathsf{LPN}},$$ □

5.2 Indistinguishability for $\mathbf{X}_0$ and $\mathbf{X}_0 \oplus \sum_{j=1}^m \mathbf{r}^{(j)}\mathbf{s}^{(j)\top}$

Next, we use Theorem 17 repeatedly to prove Theorem 25, which states that the adversary cannot distinguish between $\mathbf{X}_0$ or $\mathbf{X}_0 \oplus \sum_{i=1}^m \mathbf{r}^{(i)}\mathbf{s}^{(i)^\top}$ with non-negligible advantage, for any polynomial m. We define two new sequences of games $\mathrm{A}_{i,j}$ and $\mathrm{B}_{i,j}$, where j remains the same as in Figure 8, and $i \in \{0, \ldots, m\}$. The only change is in the key generation step. For all games, we generate $\mathbf{X}_0 \in \mathbb{F}_2^{n\times n}$ and $\forall j \in [m], \mathbf{r}^{(j)}, \mathbf{s}^{(j)} \xleftarrow{\$} \mathbb{F}_2^n$. For $\ell \in [m]$, we define $\mathbf{X}_\ell = \mathbf{X}_0 \oplus \sum_{j\leq\ell} \mathbf{r}^{(j)}\mathbf{s}^{(j)^\top}$.

In Game $\mathrm{A}_{i,j}$, if Game $\mathrm{A}_{0,j}$ had key $\mathbf{X}_t$, we now use $\mathbf{X}_{t+i}$. That is, $\mathbf{X}_{t+i}$ is used in the challenge phase, while $\mathbf{X}_0$ is used in the attack phase. We shall prove that Game $\mathrm{A}_{0,0}$ is indistinguishable from Game $\mathrm{A}_{5,m}$, and similarly Game $\mathrm{B}_{0,0}$ is indistinguishable from Game $\mathrm{B}_{5,m}$.

Theorem 23. *For all efficient $\mathcal{A}$,*

$$|\mathsf{Adv}_{\mathcal{A}}^{A_{0,0}} - \mathsf{Adv}_{\mathcal{A}}^{A_{5,m}}| \leq 2\varepsilon_{\mathsf{LPN}},$$
$$|\mathsf{Adv}_{\mathcal{A}}^{B_{0,0}} - \mathsf{Adv}_{\mathcal{A}}^{B_{5,m}}| \leq 2\varepsilon_{\mathsf{LPN}}.$$

First, we need another result on indistinguishability of games.

Corollary 24. *For all $i \in [m]$,*

$$|\mathsf{Adv}_{\mathcal{A}}^{A_{2,i-1}} - \mathsf{Adv}_{\mathcal{A}}^{A_{3,i}}| = 0,$$
$$|\mathsf{Adv}_{\mathcal{A}}^{B_{2,i-1}} - \mathsf{Adv}_{\mathcal{A}}^{B_{3,i}}| = 0,$$

Proof of Corollary 24. $\forall i \in [m]$, Game $\mathrm{A}_{2,i-1}$ and Game $\mathrm{A}_{3,i}$ both use $\mathbf{X}_i$ as the challenge key. The only difference is the labeling of the secrets: Game $\mathrm{A}_{2,i-1}$ sets $\mathsf{Ra}() = \overline{\mathsf{LHN}}_{\varepsilon,\varepsilon}(\mathbf{r}^{(i)})$, while Game $\mathrm{A}_{3,i}$ uses $\mathsf{Ra}() = \overline{\mathsf{LHN}}_{\varepsilon,\varepsilon}(\mathbf{r}^{(i-1)})$. Since $\mathbf{r}^{(i)}, \mathbf{r}^{(i-1)}$ are independent of $\mathbf{X}_i$, it follows that Game $\mathrm{A}_{2,i} \equiv$ Game $\mathrm{A}_{3,i-1}$. By the same argument, Game $\mathrm{B}_{2,i} \equiv$ Game $\mathrm{B}_{3,i-1}$. □

Proof of Theorem 23. By the triangle inequality,

$$\begin{aligned}
|\mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_{0,0}} - \mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_{5,m}}| &\leq |\mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_{0,0}} - \mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_{2,0}}| + \left(\sum_{i\in[m]} |\mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_{2,i-1}} - \mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_{3,i}}|\right) + |\mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_{3,m}} - \mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_{5,m}}| \\
&\leq 2\varepsilon_{\mathsf{LPN}} + \sum_{i\in[m]} |\mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_{2,i-1}} - \mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_{3,i-1}}| + \sum_{i\in[m]} |\mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_{3,i-1}} - \mathsf{Adv}_{\mathcal{A}}^{\mathrm{A}_{2,i}}| \qquad (19) \\
&= 2\varepsilon_{\mathsf{LPN}}. \qquad (20)
\end{aligned}$$

Equation 19 follows from Corollary 18 and Corollary 21. Equation 20 follows from Corollary 19 and Corollary 24. A similar argument establishes the second inequality. □

5.3 An Alternative Construction of $\mathbf{A}_{5,m}$ and $\mathbf{B}_{5,m}$

In the original construction, $\mathbf{X}_0 \xleftarrow{\$} \mathbb{F}_2^{n\times n}$ and $\forall j \in [m]$, $\mathbf{r}^{(j)}$ (and $\mathbf{s}^{(j)}$ in Game $\mathrm{A}_{5,m}$) are randomly chosen from $\mathbb{F}_2^n$. Consider $\mathbf{x}^{(j)}, \mathbf{y}^{(j)}$ for $j \in [m]$. In the bilinear case, we set $\mathbf{x}^{(j)} = \mathbf{r}^{(j)}$ and $\mathbf{y}^{(j)} = \mathbf{s}^{(j)}$. In the symmetric case, we set $\mathbf{x}^{(j)} = \mathbf{r}^{(j)}$ and $\mathbf{y}^{(j)} = \mathbf{r}^{(j)}$. For both cases, $\mathbf{X}_m$ is constructed from these values by $\mathbf{X}_m = \mathbf{X}_0 \oplus \sum_{j=1}^m \mathbf{x}^{(j)}\mathbf{y}^{(j)\top}$. The adversary then interacts with $\mathcal{T}^{\mathbf{X}_m}, \mathcal{R}^{\mathbf{X}_m}$ in Phase I, and $\mathcal{R}^{*\mathbf{X}_0}$ in Phase II.

Alternatively, we can build the final game by choosing $\mathbf{X}_m \xleftarrow{\$} \mathbb{F}_2^{n\times n}$ and letting $\mathbf{X}_0 = \mathbf{X}_m \oplus \sum_{j=1}^m \mathbf{x}^{(j)}\mathbf{y}^{(j)\top}$. For all $j \in [m]$, we can defer the random choice of $\mathbf{x}^{(j)}, \mathbf{y}^{(j)}$ until after Phase I, since now they are not used at all in Phase I.

Let $\mathbf{R}, \mathbf{S}$ be such that $\forall j \in [m], \mathbf{R}\mathbf{e}^{(i)} = \mathbf{r}^{(j)}, \mathbf{S}\mathbf{e}^{(j)} = \mathbf{s}^{(j)}$. The final game now looks as follows:

$$\mathsf{Adv}^{\mathrm{A}_{5,m}}_{\mathcal{A}} = \left| \Pr \left[\begin{array}{l} \mathbf{X}_m \xleftarrow{\$} \mathsf{KeyGen}_{\mathsf{HB^N}}(1^n), \\ s \xleftarrow{\$} \mathcal{A}_1^{\mathcal{T}^{\mathbf{X}_m}, \mathcal{R}^{\mathbf{X}_m}}(1^n), \\ \hat{\mathbf{A}} \xleftarrow{\$} \mathbb{F}_2^{n \times k}, \\ (\mathbf{z}^*, \hat{\mathbf{B}}) \xleftarrow{\$} \mathcal{A}_2(s, \hat{\mathbf{A}}), \\ \mathbf{R}, \mathbf{S} \xleftarrow{\$} \mathbb{F}_2^{n \times m}, \\ \mathbf{f} \xleftarrow{\$} \mathsf{Ber}_\varepsilon^n, \\ \mathbf{w}^* = \mathsf{diag}(\hat{\mathbf{A}}^\top \mathbf{X}_m^\top \hat{\mathbf{B}}) \oplus \mathsf{diag}(\hat{\mathbf{A}}^\top \mathbf{R} \mathbf{S}^\top \hat{\mathbf{B}}) \oplus \mathbf{f} \end{array} : \begin{array}{l} \forall i \in [k], \hat{\mathbf{B}}\mathbf{e}^{(i)} \neq \mathbf{0}^n, \\ |\mathbf{z}^* \oplus \mathbf{w}^*| \leq k \cdot \mathsf{u}_{\mathsf{HB^N}}(\varepsilon) \end{array} \right] \right| \tag{21}$$

$$\mathsf{Adv}^{\mathrm{B}_{5,m}}_{\mathcal{A}} = \left| \Pr \left[\begin{array}{l} \mathbf{X}_m \xleftarrow{\$} \mathsf{KeyGen.Symm}_{\mathsf{HB^N}}(1^n), \\ s \xleftarrow{\$} \mathcal{A}_1^{\mathcal{T}^{\mathbf{X}_m}, \mathcal{R}^{\mathbf{X}_m}}(1^n), \\ \hat{\mathbf{A}} \xleftarrow{\$} \mathbb{F}_2^{n \times k}, \\ (\mathbf{z}^*, \hat{\mathbf{B}}) \xleftarrow{\$} \mathcal{A}_2(s, \hat{\mathbf{A}}), \\ \mathbf{R} \xleftarrow{\$} \mathbb{F}_2^{n \times m}, \mathbf{S} = \mathbf{R}, \\ \mathbf{f} \xleftarrow{\$} \mathsf{Ber}_\varepsilon^n, \\ \mathbf{w}^* = \mathsf{diag}(\hat{\mathbf{A}}^\top \mathbf{X}_m^\top \hat{\mathbf{B}}) \oplus \mathsf{diag}(\hat{\mathbf{A}}^\top \mathbf{R} \mathbf{S}^\top \hat{\mathbf{B}}) \oplus \mathbf{f} \end{array} : \begin{array}{l} \forall i \in [k], \hat{\mathbf{B}}\mathbf{e}^{(i)} \neq \mathbf{0}^n, \\ |\mathbf{z}^* \oplus \mathbf{w}^*| \leq k \cdot \mathsf{u}_{\mathsf{HB^N}}(\varepsilon) \end{array} \right] \right| \tag{22}$$

In Section 5.5, we use Equation 21 and Equation 22 to prove the following theorem, which states that $\mathcal{A}$ has negligible advantage in the final games:

Theorem 25. *For all probabilistic polynomial time $\mathcal{A}$,*

$$\mathsf{Adv}^{A_{5,m}}_{\mathcal{A}} \leq \mathsf{negl},$$
$$\mathsf{Adv}^{B_{5,m}}_{\mathcal{A}} \leq \mathsf{negl}.$$

Theorem 25 will follow if we can prove that $\mathbf{w}^*$ is almost uniformly distributed, since no adversary can guess a uniformly distributed value. In order to do this, we first develop some technical tools in Section 5.4 to address the $\mathsf{diag}(\hat{\mathbf{A}}^\top \mathbf{R} \mathbf{S}^\top \hat{\mathbf{B}})$ component of the sum.

5.4 A Theorem for Products of Random Matrices

Let $S_{\hat{\mathbf{A}}}$ be the event that $\mathsf{rank}(\hat{\mathbf{A}}) < k$. Let $T_{\hat{\mathbf{B}}^\top \mathbf{S}}$ be the event that $\mathsf{rank}(\hat{\mathbf{B}}^\top \mathbf{S}) = 0$. The main result of this section is the following theorem.

Theorem 26. *Given $\hat{\mathbf{A}} \xleftarrow{\$} \mathbb{F}_2^{n \times k}$ and any $\hat{\mathbf{B}} \in \mathbb{F}_2^{n \times k}$ such that $\forall i \in [m], \hat{\mathbf{B}}\mathbf{e}^{(i)} \neq \mathbf{0}^k$, let $\mathbf{R} \xleftarrow{\$} \mathbb{F}_2^{n \times m}$. For the bilinear protocol, $\mathbf{S} \xleftarrow{\$} \mathbb{F}_2^{n \times m}$, while for the symmetric protocol, $\mathbf{S} = \mathbf{R}$. Then for any $\hat{\mathbf{z}} \in \mathbb{F}_2^k$,*

(a) $\Pr[S_{\hat{\mathbf{A}}}] \leq 2^{k-n}$, $\Pr[T_{\hat{\mathbf{B}}^\top \mathbf{S}}] \leq k 2^{-m}$

(b) If events $S_{\hat{\mathbf{A}}}$ and $T_{\hat{\mathbf{B}}^\top \mathbf{S}}$ do not occur, then $\Pr[\mathsf{diag}(\hat{\mathbf{A}}^\top \mathbf{R} \mathbf{S}^\top \hat{\mathbf{B}}) = \hat{\mathbf{z}}] = 2^{-k}$

Roughly speaking, Theorem 26 states that $\hat{\mathbf{A}}$ and $\hat{\mathbf{B}}^\top \mathbf{S}$ are "degenerate" only with negligible probability, and if $\hat{\mathbf{A}}, \hat{\mathbf{B}}^\top \mathbf{S}$ are nondegenerate, then $\mathsf{diag}(\hat{\mathbf{A}}^\top \mathbf{R} \mathbf{S}^\top \hat{\mathbf{B}})$ is uniformly distributed.

Proof of Theorem 26(a). First consider $S_{\hat{\mathbf{A}}}$. We find that

$$\begin{aligned}
\Pr[\mathsf{rank}(\hat{\mathbf{A}}) < k] &= \Pr[\exists \mathbf{x} \in \mathbb{F}_2^k \setminus \{\mathbf{0}^k\} : \hat{\mathbf{A}}\mathbf{x} = \mathbf{0}^k] && (23)\\
&\leq \sum_{\mathbf{x} \in \mathbb{F}_2^k \setminus \{\mathbf{0}^k\}} \Pr[\hat{\mathbf{A}}\mathbf{x} = \mathbf{0}^k] && (24)\\
&= \sum_{\mathbf{x} \in \mathbb{F}_2^k \setminus \{\mathbf{0}^k\}} \prod_{i \in [n]} \Pr[(\mathbf{e}^{(i)\top}\hat{\mathbf{A}})\mathbf{x} = 0] && (25)\\
&= (2^k - 1) \cdot \prod_{i \in [n]} \frac{2^{k-1}}{2^k} && (26)\\
&\leq 2^{k-n}
\end{aligned}$$

Equation 23 and Equation 26 both follows from $\mathsf{rank}(\mathbf{X}) + \mathsf{rank}(\ker(\mathbf{X})) = k$, for $\mathbf{X} = \hat{\mathbf{A}}, \mathbf{x}$ respectively. Equation 24 follows from the union bound. Equation 25 follows from independence of the columns $\mathbf{e}^{(i)\top}\hat{\mathbf{A}}$ of $\hat{\mathbf{A}}$.

Next consider $T_{\hat{\mathbf{B}}^\top \mathbf{S}}$. We find that $\forall i \in [k]$,

$$\begin{aligned}
\Pr[\exists i \in [k], \mathbf{S}^\top\hat{\mathbf{B}}\mathbf{e}^{(i)} = \mathbf{0}^m] &\leq \sum_{i \in [k]} \Pr[\mathbf{S}^\top\hat{\mathbf{B}}\mathbf{e}^{(i)} = \mathbf{0}^m] && (27)\\
&= \sum_{i \in [k]} \prod_{j \in [m]} \Pr[\mathbf{e}^{(j)\top}\mathbf{S}\hat{\mathbf{B}}\mathbf{e}^{(i)} = 0] && (28)\\
&= \sum_{i \in [k]} \prod_{j \in [m]} \frac{2^{m-1}}{2^m} && (29)\\
&= 2^{-m} && (30)
\end{aligned}$$

Equation 27 follows from the union bound. Equation 28 follows from independence of the columns $\mathbf{e}^{(j)\top}\mathbf{S}$ of $\mathbf{S}$. Equation 29 follows from $\mathsf{rank}(\mathbf{X}) + \mathsf{rank}(\ker(\mathbf{X})) = m$ for $\mathbf{X} = \hat{\mathbf{B}}\mathbf{e}^{(i)}$. □

We move on to Theorem 26(b). We will need the following two lemmata.

Lemma 27. *Let* $\mathbf{R} \xleftarrow{\$} \mathbb{F}_2^{n \times m}$. *Then* $\forall \hat{\mathbf{A}} \in \mathbb{F}_2^{n \times k}$ *with* $\mathsf{rank}(\hat{\mathbf{A}}) = k \leq n$, $\forall \hat{\mathbf{Y}} \in \mathbb{F}_2^{k \times m}$, $\Pr[\hat{\mathbf{A}}^\top\mathbf{R} = \hat{\mathbf{Y}}] = 2^{-km}$.

Proof. Each column $\hat{\mathbf{Y}}\mathbf{e}^{(i)} = \hat{\mathbf{A}}^\top(\mathbf{R}\mathbf{e}^{(i)})$ is an independently random element of $\mathsf{Im}(\hat{\mathbf{A}}^\top)$. Since $\hat{\mathbf{A}}$ has full rank, $\mathsf{Im}(\hat{\mathbf{A}}^\top)$ is all of $\mathbb{F}_2^k$, so that each column is a uniformly random k-bit vector. □

Lemma 28. *Let* $\mathbf{Y} \xleftarrow{\$} \mathbb{F}_2^{k \times m}$, $\mathbf{S} \xleftarrow{\$} \mathbb{F}_2^{n \times m}$. *Given any* $\hat{\mathbf{z}} \in \mathbb{F}_2^k$ *and* $\hat{\mathbf{B}} \in \mathbb{F}_2^{n \times k}$ *so that* $\forall i \in [k], \mathbf{S}^\top\hat{\mathbf{B}}\mathbf{e}^{(i)} \neq \mathbf{0}^n$,

$$\Pr[\mathsf{diag}(\mathbf{Y}\mathbf{S}^\top\hat{\mathbf{B}}) = \hat{\mathbf{z}}] = 2^{-k}$$

Proof. For all $i \in [k]$, let $\mathbf{y}^{(i)} = \mathbf{Y}^\top\mathbf{e}^{(i)}$ and $\mathbf{x}^{(i)} = \mathbf{S}^\top\hat{\mathbf{B}}\mathbf{e}^{(i)}$. Then

$$\begin{aligned}
\Pr[\mathsf{diag}(\mathbf{Y}\mathbf{S}^\top\hat{\mathbf{B}}) = \hat{\mathbf{z}}] &= \prod_{i=1}^{k} \Pr[\mathbf{y}^{(i)\top}\mathbf{x}^{(i)} = \hat{\mathbf{z}}_i] && (31)\\
&= \prod_{i=1}^{k} \frac{|\{\mathbf{y}^{(i)} : \mathbf{y}^{(i)\top}\mathbf{x}^{(i)} = \hat{\mathbf{z}}_i\}|}{|\mathbb{F}_2^n|} && (32)\\
&= \prod_{i=1}^{k} \frac{2^{n-1}}{2^n} && (33)\\
&= 2^{-k}
\end{aligned}$$

Equation 31 follows from expressing the diagonal of the product $\mathbf{Y}\mathbf{S}^\top\hat{\mathbf{B}}$ in terms of $\mathbf{Y}$ and $\mathbf{S}^\top\hat{\mathbf{B}}$. Equation 32 follows from independence of the $\mathbf{y}^{(i)}$. Equation 33 follows from $|\ker(\mathbf{x}^{(i)})| = 2^{n-1} = |\mathbb{F}_2^n \setminus \ker(\mathbf{x}^{(i)})|$ for $\mathbf{x}^{(i)} \neq \mathbf{0}^k$. □

Theorem 26(b) now follows immediately from Lemma 27 and Lemma 28.

5.5 Final Steps

Now we have all the tools necessary to prove Theorem 25 and, finally, Theorem 16.

Proof of Theorem 25. By Theorem 26, for any vector $\hat{\mathbf{z}} \in \mathbb{F}_2^k$ of guesses made by the adversary, if $S_{\hat{\mathbf{A}}}$,$T_{\hat{\mathbf{B}}^\top\mathbf{S}}$ do not occur, then then $\Pr[\mathbf{w} = \hat{\mathbf{z}}] = 2^{-k}$, so that

$$\begin{aligned}\mathbf{w} &= \mathsf{diag}(\hat{\mathbf{A}}^\top\mathbf{X}_m\hat{\mathbf{B}}) \oplus \mathsf{diag}(\hat{\mathbf{A}}\mathbf{R}\mathbf{S}^\top\hat{\mathbf{B}}) \oplus \mathbf{f} \\ &= \mathsf{diag}(\hat{\mathbf{A}}^\top\mathbf{X}_m\hat{\mathbf{B}}) \oplus \mathsf{Ber}^k_{\frac{1}{2}} \oplus \mathbf{f} \\ &= \mathsf{Ber}^k_{\frac{1}{2}} \end{aligned} \tag{34}$$

Equation 34 follows from Fact 2. Therefore, for a random vector $\mathbf{u} \xleftarrow{\$} \mathbb{F}_2^k$, if $S_{\hat{\mathbf{A}}}$,$T_{\hat{\mathbf{B}}^\top\mathbf{S}}$ do not occur, then

$$\begin{aligned}\Pr[|\mathbf{z} \oplus \mathbf{w}| \leq k \cdot \mathsf{u}_{\mathsf{HB}^{\mathsf{N}}}(\varepsilon)] &= \Pr[|\mathbf{z} \oplus \mathbf{u}| \leq k \cdot \mathsf{u}_{\mathsf{HB}^{\mathsf{N}}}(\varepsilon)] \\ &\leq 2^{-k\left(\left(\frac{1}{2} - (1+\delta)(\varepsilon\oplus\varepsilon)\right)\right)^2}\end{aligned} \tag{35}$$

Equation 35 follows from the well-known Chernoff bound, $\Pr[X \leq (1-\mu)\cdot X] \leq e^{-\mu^2 k}$.

Recall that with $\mathsf{u}_{\mathsf{HB}^{\mathsf{N}}}(\varepsilon) = (1+\delta)(\varepsilon \oplus \varepsilon)$, HB^{N} achieves completeness $e^{-\delta^2 k}$, i.e. an honest $\mathcal{T}$ fails with probability at most $e^{-\delta^2 k}$. If we set δ so that $\mathsf{u}_{\mathsf{HB}^{\mathsf{N}}}(\varepsilon) = (\varepsilon\oplus\varepsilon)(1+\delta) = \frac{1}{2}(1-\delta)$, we obtain the same bound of $e^{-\delta^2 k}$ for both soundness and completeness. $(\varepsilon\oplus\varepsilon)(1+\delta) = \frac{1}{2}(1-\delta)$ results in $\delta = \frac{\frac{1}{2}-(\varepsilon\oplus\varepsilon)}{\frac{1}{2}+(\varepsilon\oplus\varepsilon)} = \frac{1-4\varepsilon+4\varepsilon^2}{1+4\varepsilon-4\varepsilon^2}$. As a result, we obtain

$$\begin{aligned}\Pr[|\mathbf{z} \oplus \mathbf{w}| \leq k \cdot \mathsf{u}_{\mathsf{HB}^{\mathsf{N}}}(\varepsilon)] &\leq 2^{-k\delta^2} \\ &= 2^{-k\left(\frac{1-4\varepsilon+4\varepsilon^2}{1+4\varepsilon-4\varepsilon^2}\right)^2}\end{aligned}$$

If ε is a constant, for example, the bound is $2^{-O(k)}$, which is negligible.

□

Proof of Theorem 16.

$$\mathsf{Adv}^{\mathsf{HB}^{\mathsf{N}}}_{\mathcal{A}} \leq \mathsf{Adv}^{\mathrm{A}_{5,m}}_{\mathcal{A}} + 2\varepsilon_{\mathsf{LPN}} \tag{36}$$

$$\leq \left(2^{k-n} + k2^{-m} + 2^{-k\left(\frac{1-4\varepsilon+4\varepsilon^2}{1+4\varepsilon-4\varepsilon^2}\right)^2}\right) + 2\varepsilon_{\mathsf{LPN}} \tag{37}$$

$$= \mathsf{negl} \tag{38}$$

Equation 36 follows from Theorem 23. Equation 37 follows from applying Theorem 26. Equation 38 follows from the LPN assumption and from setting $k < n - \omega(\log n), m = n$, and $\varepsilon = \theta(1)$. □

6 Conclusion

We have introduced HB^{N}, a bilinear version of HB, and proven its security in the Man-in-the-Middle model, for both the non-symmetric and symmetric variants. Along the way, we have introduced a new notation the simplifies working with random variables over $\mathbb{F}_2$, assembled a useful collection of lemmas for working with LPN, and introduced the LHN problem. We hope that these technical tools will be useful for future work on protocols based on LPN.

References

[AL87] Dana Angluin and Philip D Laird. Learning from Noisy Examples. *Machine Learning*, 2(4):343–370, 1987.

[BC08] Julien Bringer and Herve Chabanne. Trusted-HB: a low-cost version of HB secure against man-in-the-middle attacks. *arXiv*, 2008.

[BCD06] Julien Bringer, Hervé Chabanne, and Emmanuelle Dottax. HB^{++}: a lightweight authentication protocol secure against some attacks. In *Second International Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing (SecPerU 2006)*, pages 28–33. IEEE Computer Society, 2006.

[DK08] D Duc and Kwangjo Kim. Securing HB against GRS man-in-the-middle attack. *caislab.icu.ac.kr*, 2008.

[FS09] Dmitry Frumkin and Adi Shamir. Un-Trusted-HB: Security vulnerabilities of Trusted-HB. *EPrint*, 2009.

[GRS05] Henri Gilbert, Matthew Robshaw, and Herve Sibert. Active attack against HB^{+}: a provably secure lightweight authentication protocol. *Electronics Letters*, 2005.

[GRS08a] Henri Gilbert, Matthew Robshaw, and Yannick Seurin. Good variants of HB^{+} are hard to find. In *Proc. Financial Cryptography and Data Security*, pages 156–170, 2008.

[GRS08b] Henri Gilbert, Matthew Robshaw, and Yannick Seurin. $\text{HB}^{\#}$: Increasing the security and efficiency of HB. In *Proc. EUROCRYPT*, volume 4965, pages 361–378, 2008.

[HB01] Nicholas Hopper and Manuel Blum. Secure human identification protocols. In *Proc. ASIACRYPT*, 2001.

[JW05] Ari Juels and Stephen Weis. Authenticating pervasive devices with human protocols. In *Proc. CRYPTO*, pages 293–308, 2005.

[Kea93] M. Kearns. Efficient noise-tolerant learning from statistical queries. In *Proceedings of the 25th ACM Symposium on Theory of Computing*, pages 392–401. ACM, 1993.

[KPC+11] Eike Kiltz, Krzystof Pietrzak, David Cash, Abhishek Jain, and Daniele Venturi. Efficient Authentication from Hard Learning Problems. In *Proc. Eurocrypt*, pages 7–26, 2011.

[KSS10] Jonathan Katz, Ji Sun Shin, and Adam Smith. Parallel and concurrent security of the HB and HB^{+} protocols. *Journal of Cryptology*, 23(3):402–421, 2010.

[LMM08] X Leng, K Mayes, and K Markantonakis. HB-MP+ protocol: An improvement on the HB-MP protocol. *2008 IEEE International Conference on RFID*, 2008.

[MP07] Jorge Munilla and Alberto Peinado. HB-MP: A further step in the HB-family of lightweight authentication protocols. *Computer Networks*, 2007.

[OOV08] Khaled Ouafi, Raphael Overbeck, and Serge Vaudenay. On the security of $\text{HB}^{\#}$ against a man-in-the-middle attack. *Proc. ASIACRYPT*, 2008.

[Pie10] Krzystof Pietrzak. Subspace LWE, 2010. Manuscript available at `http://homepages.cwi.nl/~pietrzak/publications/SLWE.pdf`.

A Modeling the Active Security Game

The adversary $\mathcal{A}$ can be defined as two algorithms $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$. In Phase I, $\mathcal{A}_1$ has access to the Phase I oracles $\mathcal{T}, \mathcal{R}$, and outputs its state s for input to $\mathcal{A}_2$. $\mathcal{A}_2$ submits $\mathbf{b}^{*(i)}$ to the Phase II challenger $\mathcal{R}^*$ (either in parallel or in serial) and receives $\mathbf{a}^{*(i)}$ in exchange, as shown in Figure 10. From the model, we see that the reason HB^{N} can be used in either two or three rounds is precisely because the computation of $\mathbf{b}^{(i)}$ does not depend on $\mathbf{a}'^{(i)}$,

and $\mathbf{a}^{(i)}$ does not depend on $\mathbf{b}'^{(i)}$.

Phase I	$s \xleftarrow{\$} \mathcal{A}_1^{\mathcal{T},\mathcal{R}}$,
Phase II In serial or parallel	$\mathbf{b}^{*(i)} \xleftarrow{\$} \mathcal{A}_2(s)$, $\mathbf{a}^{*(i)} \xleftarrow{\$} \mathcal{R}^*(\mathbf{b}^{*(i)})$,
Phase II: Final	$\mathbf{z}_i^* \xleftarrow{\$} \mathcal{A}_2(\mathbf{b}^{*(i)}, \mathbf{a}^{*(i)}, s)$

Two round	Three Round
$\mathbf{b}^{(i)} \xleftarrow{\$} \mathcal{T}^{\mathbf{b}}(\mathbf{a}'^{(i)})$	$\mathbf{b}^{(i)} \xleftarrow{\$} \mathcal{T}^{\mathbf{b}}()$
$\mathbf{a}^{(i)} \xleftarrow{\$} \mathcal{R}^{\mathbf{a}}()$	$\mathbf{a}^{(i)} \xleftarrow{\$} \mathcal{R}^{\mathbf{a}}(\mathbf{b}'^{(i)})$
$\mathbf{z}_i \xleftarrow{\$} \mathcal{T}^{\mathbf{z}}(\mathbf{a}'^{(i)}, \mathbf{b}^{(i)})$	$\mathbf{z}_i \xleftarrow{\$} \mathcal{T}^{\mathbf{z}}(\mathbf{a}'^{(i)}, \mathbf{b}^{(i)})$
$\mathbf{w}_i \xleftarrow{\$} \mathcal{T}^{\mathbf{w}}(\mathbf{z}'_i, \mathbf{a}^{(i)}, \mathbf{b}'^{(i)})$	$\mathbf{w}_i \xleftarrow{\$} \mathcal{T}^{\mathbf{w}}(\mathbf{z}'_i, \mathbf{a}^{(i)}, \mathbf{b}'^{(i)})$

Figure 10: Modeling the Oracles in Two and Three Rounds