Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 5973285, 11 pages
https://doi.org/10.1155/2018/5973285

Research Article

Oblivious Transfer via Lossy Encryption from Lattice-Based Cryptography

Zengpeng Li,[1] **Can Xiang**,[2] **and Chengyu Wang**[3]

[1]*College of Computer Science and Technology, Qingdao University, Qingdao 266071, China*
[2]*College of Mathematics and Informatics, South China Agricultural University, Guangzhou 510642, China*
[3]*School of CyberSpace Security, Beijing University of Posts and Telecommunications, Beijing 100871, China*

Correspondence should be addressed to Can Xiang; xiangcan1987@sina.com

Received 9 April 2018; Accepted 10 July 2018; Published 2 September 2018

Academic Editor: Jian Shen

Authentication is the first defence line to prevent malicious entities to access *smart mobile devices* (or SMD). Essentially, there exist many available cryptographic primitives to design authentication protocols. *Oblivious transfer* (OT) protocol is one of the important cryptographic primitives to design authentication protocols. The first lattice-based OT framework under universal composability (UC) model was designed by dual mode encryption and promoted us to find an alternative efficient scheme. We note that "lossy encryption" scheme is an extension of the dual mode encryption and can be used to design UC-secure OT protocol, but the investigations of OT via lossy encryption over the lattice are absent. Hence, in order to obtain an efficient authentication protocol by improving the performance of the UC-secure OT protocol, in this paper, we first design a multibit lossy encryption under the decisional learning with errors (LWE) assumption and then design a new variant of UC-secure OT protocol for authenticated protocol via lossy encryption scheme. Additionally, our OT protocol is secure against semihonest (static) adversaries in the common reference string (CRS) model and within the UC framework.

1. Introduction

Oblivious transfer (OT) is an important cryptographic primitive which can be used for designing secure multiparty computing and privacy-preserving schemes, such as authenticated key exchange and password-based authentication key exchange (PAKE) [1]. Apparently, authentication is the first line of defence to prevent unauthorized access from illegitimate entities (including both devices and users). Very recently, the issues of privacy-preserving for *smart mobile devices* (SMD), *Internet of things* (IoTs), *wireless sensor networks* (WSNs), and *cloud storage auditing* are arousing the public attention [2–4]. In this case, it is becoming more important to protect the private information for mobile computing environments by utilizing the technique of authentication [5, 6]. However, the literature which focuses on the problem of how to use the basic cryptographic primitive to design authentication protocols is relatively few and the research of improving the performance of the primitives is fewer. Thus, to design the efficient authentication protocols in the future, this paper focuses on how to improve the performance of universal composable- (UC-) secure OT protocol via lattice-based cryptography.

To our knowledge, OT protocol was originally proposed by Rabin [7]. Since then various cryptography schemes and protocols are designed by using OT, e.g., [1, 8]. Informally speaking, there exist two players (the sender **S** and the receiver **R**) in OT protocol. On the one hand, the sender can send two (or more) values to the receiver. However, the sender does not know which value will be received by the receiver, and the receiver only knows the received value and remains oblivious to the other values. In a word, they are oblivious to other's true behaviour.

Importantly, Gertner et al. [9] pointed out the relationship between the public key encryption (PKE) scheme and the OT protocol at FOCS'00. In some indistinguishability against chosen plaintext attack (IND-CPA) secure PKE schemes, if the public key generated by the key generation algorithm is

indistinguishable from the public key sampled from a uniform distribution, then we can use the PKE scheme to design an OT protocol [9]. In this setting, Peikert-Vaikuntanathan-Waters (PVW) [10] has constructed an *efficient, universally composability and generally realizable* OT via "dual mode encryption" under *worst-case* lattice assumption (LWE) at CRYPTO'08. Loosely speaking, there are two types of public key in "dual mode encryption". One type is injective keys; it is real public key and behaves normally. The other one is "lossy" or "messy" key, it is lossy public key, and it loses some information of the plaintext. Moreover, there exist two important properties for the dual mode encryption. The first one is statistically close; namely, the distributions of ciphertext for any two plaintexts under a lossy key are statistically close. The second one computationally indistinguishable; namely, the injective key is computationally indistinguishable from the lossy key. (Importantly, no efficient adversary can tell the difference between normal keys and lossy keys.)

Along with this line, the notation of "*lossy encryption*" was proposed by Bellare, Hofheinz, and Yilek (BHY) [11] on EUROCRYPT'09. Actually, the lossy encryption is an extension of the meaningful/meaningless encryption [12] and dual mode encryption [10]. In a nutshell, a "lossy" (or "messy" in [10]) cryptosystem is one which also has two modes according to two types of public keys. Concretely, **(1)** In the normal mode, the ciphertext is generated by encrypting the plaintext under an injective key. **(2)** In the lossy (or "messy") mode, the ciphertext is independent of the plaintext. Actually, the operability property was proposed by [11]; they basically can open a ciphertext generated under a lossy key for any plaintext by adopting a possibly inefficient algorithm. Meanwhile, the injective key is computationally indistinguishable from the lossy key. Actually, our work is along this line and we embark on this question:

> *How to design a (string)* OT *protocol via multibit lattice-based "lossy encryption" rather than "dual mode encryption"?*

To solve this issue, we note that, after the polynomial time solvers in the nonclassical quantum computation model was pointed out by Shor [13] for discrete logarithm and integer factorization, most researchers seek to find the various alternative computational assumption; thus lattice-based (e.g., learning with errors, LWE) cryptography draws attention. Over the last decade, lattice has emerged as a very attractive foundation separately for cryptography. Specially, Regev scheme [14] and Gentry-Peikert-Vaikutanathan (GPV, a.k.a., dual Regev) scheme [15] are important lattice-based schemes to remain secure even against quantum computer attacks.

From the above observations and inspired by the work of Peikert et al. [10], we still work along this line and construct a multibit LWE-based lossy encryption scheme which has two types of public keys.

1.1. Our Contributions and Techniques Overview. Although many would consider OT protocol a breakthrough for multiparty computation, nowadays, OT protocols are plagued by several well-known pain-points among which performance (string OT) and security (postquantum attacks) are perhaps the most visible and most often debated points. However, most existing OT protocols adopt a variety of standard number-theoretic assumptions; only a few works focus on designing the protocol under *worst-case* lattice assumption such as [10, 16, 17]. Here we fill in some of the missing details in the high-level description.

(1) We use the lossy encryption scheme to replace the dual mode encryption, then we design the OT protocol via lossy encryption over the lattice. More concretely, we note that, Peikert et al. [10] proposed the framework of OT protocol by using dual mode cryptography. Actually, the lossy encryption is an extension of the meaningful/meaningless encryption [12] and dual mode encryption [10] and has the obvious property of two types of public keys. The crux of this issue is how to obtain multibit lossy encryption scheme. In this paper, we construct *lossy encryption* via multibit Gentry-Peikert-Vaikuntanathan (GPV, a.k.a., dual Regev scheme), i.e., MGPV scheme. In particular, the public key in MGPV with many LWE instances rather than a simple matrix of LWE instance.

(2) Moreover, we design a multibit public key encryption scheme (i.e., MGPV scheme) by following the methodology of Li et al. [18]. Actually, the semantically secure multibit public key encryption scheme via subset sum problem (SSP) proposed by Lyubashevsky et al. [16] and multiple secrets Gentry-Sahai-Waters scheme via LWE assumption proposed by Li et al. [19] promoted us to explore the functions of the public key with a sequence of LWE instances. Importantly, the public matrix $\mathbf{A}$ contains many LWE instances, each one is used to protect the secret key. In this setting, the decrypter can decrypt the plaintext in a bit-by-bit manner.

(3) Lastly, we attempt to explore the potential application of our UC-secure OT protocol for PAKE in SMDs. To our knowledge, PAKE is an important tool to design authentication protocol, which can help SMDs to enable adequate user authentication and prevent unauthorized use of an unattended and lost, etc. Inspired by the OT-based PAKE [1], extending our multibit OT protocol via lossy encryption to multibit PAKE is a natural result. This solution is aimed at helping us apply SMDs for authentication and other security services.

1.2. Paper Organization. In Section 2 we formally define and present some related notations. In Section 3 we describe our multibit encryption scheme (hereafter MGPV) via LWE assumption. In Section 4 we describe our lossy encryption scheme (hereafter LE) via MGPV scheme. In Section 5 we describe a oblivious transfer protocol via the constructed LE scheme. In Section 6, we explore the potential application PAKE in SMDs. Finally, we give a conclusion in Section 7.

2. Preliminaries

Below, we introduce some necessary notations.

2.1. Notation. Throughout our paper, vectors is denoted by bold lower-case letters, e.g., $\mathbf{a}$, and matrices were denoted by upper-case letters, e.g., $\mathbf{A}$. The matrix $\mathbf{B} = \{\mathbf{b}_1, \cdots, \mathbf{b}_n\} \in \mathbb{R}^n$ contains n linearly independent vectors. The basis $\mathbf{B}$ can be used to generate the n-dimensional lattice Λ as follows:

$$\Lambda = L(\mathbf{B}) = \left\{ \mathbf{Bc} = \sum_{i \in [n]} c_i \cdot \mathbf{b}_i : \mathbf{c} \in \mathbb{Z}_q^n \right\}. \quad (1)$$

Below we give a variant of leftover hash lemma.

Lemma 1 (see [20] Lemma 2.1). *We first denote the statistical distance between the distribution $\mathbf{A}$ and $\mathbf{B}$ by $\Delta(\mathbf{A}, \mathbf{B})$. If the parameters satisfy the following conditions, i.e., $\lambda \in \mathbb{Z}$, $n \in \mathbb{N}$, $q \in \mathbb{N}$, $m \geq n \log q + 2\lambda$, $\mathbf{A}$ is a uniform random matrix over $\mathbb{Z}_q^{m \times n}$, $\mathbf{r} \xleftarrow{R} \{0,1\}^m$ and $\mathbf{y} \xleftarrow{R} \mathbb{Z}_q^n$. Then we have that*

$$\Delta\left(\left(\mathbf{A}, \mathbf{A}^T \cdot \mathbf{r}\right), (\mathbf{A}, \mathbf{y})\right) \leq 2^{-\lambda} \quad (2)$$

2.2. Gaussian Distribution. We denote the truncated discrete Gaussian distribution over $\mathbb{Z}^m$ with parameter σ by $D_{\mathbb{Z}_q^m, \sigma}$ and let $D_{\mathbb{Z}^m, \sigma}$ be $\sqrt{m} \cdot \sigma$-bounded.

Remark 2. If define $D_\sigma \longleftarrow \mathbb{Z} \Longrightarrow e \longleftarrow D_\sigma, |e| \leq \sigma$, then $D_\sigma^m \longleftarrow \mathbb{Z}^m \Longrightarrow \mathbf{e} \longleftarrow D_\sigma^m, |e| \leq \sqrt{m} \cdot \sigma$. Throughout the paper, we suppose $\sigma \geq 2\sqrt{m}$. Therefore, if $\mathbf{e} \longleftarrow D_\sigma^m$ then we have, on average, that $\|\mathbf{e}\| \approx \sqrt{m} \cdot \sigma$.

2.3. Learning with Errors

Definition 3 (LWE distribution). The LWE distribution $\mathcal{A}_{\mathbf{s},\chi}$ over $\mathbb{Z}_q^n \times \mathbb{Z}_q$ is sampled by sampling $\mathbf{a} \in \mathbb{Z}_q^n$ uniformly, choosing $e \longleftarrow \chi$, and outputting $(\mathbf{a}, b = \langle \mathbf{s}, \mathbf{a} \rangle + e \pmod q)$ for a secret vector $\mathbf{s} \in \mathbb{Z}_q^n$.

Below we describe the decision version.

Definition 4 (decision-$\mathsf{LWE}_{n,q,\chi,m}$). Sampled m samples $(\mathbf{a}_i, b_i)$ independently over $\mathbb{Z}_q^n \times \mathbb{Z}_q$. For every sample $(\mathbf{a}_i, b_i)$ the following distributions are indistinguishable (with nonnegligible advantage). (1) $(\mathbf{a}_i, b_i) \longleftarrow \mathcal{A}_{\mathbf{s},\chi}$, (2) $(\mathbf{a}_i, b_i)$ sampled from the uniform distribution.

2.4. Inhomogeneous Short Integer Solution. In this subsection, we review the Inhomogeneous Short Integer Solution (ISIS) problem as follows.

Definition 5 (ISIS). Given an integer q, a public matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, a vector $\mathbf{b} \in \mathbb{Z}_q^n$, and a real β, then find an integer vector $\mathbf{e} \in \mathbb{Z}^m$ such that $\mathbf{Ae} = \mathbf{b} \pmod q$ and $\|\mathbf{e}\|_2 \leq \beta$.

2.5. Lossy Encryption

Definition 6 ((perfectly) lossy encryption [21]). An encryption scheme $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ is called "lossy" if there exists a probabilistic polynomial time (PPT) algorithm $\mathsf{KeyGen}_{\mathsf{lossy}}$ that takes 1^λ as input and outputs pk_{lossy} such that

(i) the distribution pk_{lossy} is computationally indistinguishable from a public key pk generated by $\mathsf{Gen}(1^\lambda)$;

(ii) for every two equal-length messages m_0 and m_1, the distributions $\mathsf{Enc}(pk_{\mathsf{lossy}}, m_0)$ and $\mathsf{Enc}(pk_{\mathsf{lossy}}, m_1)$ are identically distributed for every $pk_{\mathsf{lossy}} \longleftarrow \mathsf{KeyGen}_{\mathsf{lossy}}(1^\lambda)$.

Remark 7. If an encryption scheme is lossy then it is semantically secure.

It is given by a tuple of PPT algorithms

$$\left\{\mathsf{KeyGen}_{\mathsf{real}}, \mathsf{KeyGen}_{\mathsf{lossy}}, \mathsf{Enc}, \mathsf{Dec}\right\}. \quad (3)$$

The details are as follows:

(i) $\mathsf{KeyGen}_{\mathsf{real}}(1^\lambda, \mathsf{inj})$ takes as input a security parameter λ and outputs either the real public key along with the secret key (pk_{real}, sk) or the injective key.

(ii) $\mathsf{KeyGen}_{\mathsf{lossy}}(1^\lambda, \mathsf{lossy})$ takes as input λ and outputs a lossy public key and $\perp$ instead of sk, i.e., $(pk_{\mathsf{lossy}}, \perp)$.

(iii) $\mathsf{Enc}(pk, m)$ takes as input either pk_{real} or pk_{lossy} and message m and outputs a ciphertext C.

(iv) $\mathsf{Dec}(sk, C)$ takes as input a secret key sk and a ciphertext C and outputs either a message m or $\perp$.

Lemma 8 (see [15]). *If consider all but a $2q^{-n}$ fraction of all matrix $\mathbf{A}$ over $\mathbb{Z}_q^{m \times n}$ along with any $s \geq \omega(\sqrt{\log m})$ and $m \geq 2n \log q$, then the distribution of $\mathbf{u}^T = \mathbf{e}^T \cdot \mathbf{A} \pmod q$ is statistically close to uniform over $\mathbf{Z}_q^n$, where $\mathbf{e} \sim \mathcal{D}_{\mathbb{Z},s}^m$.*

Lemma 9 (see [22]). *We first denote the distribution $\mathcal{D}$ over $\mathbb{Z}_q^n$ with min-entropy k. If $\mathbf{C} \longleftarrow \mathbb{Z}_q^{l \times n}$ is a uniform matrix and $\mathbf{s} \in \mathbb{Z}_q^{n \times n}$ is sampled from the distribution $\mathcal{D}$ for any $\varepsilon > 0$ and $l \leq (k - 2\log(1/\varepsilon) - O(1))/\log q$, then the joint distribution of $(\mathbf{C}, \mathbf{C} \cdot \mathbf{s})$ is ε-close to the uniform distribution over $\mathbb{Z}_q^{l \times n} \times \mathbb{Z}_q^l$.*

Lemma 10 (see [22]). *Consider a distribution "Lossy" for $\overline{\mathbf{A}} \longleftarrow Lossy \approx_c \mathbf{U} \longleftarrow \mathbb{Z}_q^{m \times n}$. If given $\mathbf{s} \longleftarrow \mathbb{Z}_q^n$ and $\mathbf{e} \longleftarrow \mathcal{A}_{\mathbb{Z},\beta \cdot q}^{m \times n}$, there exists $\widetilde{\mathbf{H}}_\infty^\varepsilon(\mathbf{s} \mid \overline{\mathbf{A}}, \overline{\mathbf{A}} \cdot \mathbf{s} + \mathbf{x}) \geq n$ for $\varepsilon = negl(\lambda)$. Then the Lossy distribution is as follows:*

(i) *Choose $\mathbf{D} \longleftarrow \mathbb{Z}_q^{m \times k}$, $\mathbf{C} \longleftarrow \mathbb{Z}_q^{k \times n}$, and $\mathbf{Z} \longleftarrow \mathcal{D}_{\alpha q}^{m \times n}$, where $\alpha/\beta = negl(\lambda)$ and $k \cdot \log q \leq n - 2\lambda + 2$.*

(ii) *Let $\overline{\mathbf{A}} = \mathbf{DC} + \mathbf{Z}$.*

(iii) *Output $\overline{\mathbf{A}}$.*

2.6. The Universal Composability (UC) Framework. The UC framework first defines a PPT environment machine $\mathcal{Z}$ and then uses the machine to oversee the execution of a protocol in one of two worlds. The detailed description of the executions was presented by Canetti [23], and there exist two world ensembles $\mathsf{EXEC}_{\pi,\mathcal{A},\mathcal{Z}}$ and $\mathsf{IDEAL}_{F,\mathcal{S},\mathcal{Z}}$ for real world and ideal world, respectively.

Definition 11 (see [10] Def.2.1). If there exists a simulator $\mathcal{S}$ for any adversary $\mathcal{A}$ such that for all environments $\mathcal{Z}$

$$\mathsf{IDEAL}_{F,\mathcal{S},\mathcal{Z}} \overset{c}{\approx} \mathsf{EXEC}_{\pi,\mathcal{A},\mathcal{Z}}, \tag{4}$$

then we can say that a protocol π is UC-realize a functionality F.

3. Multibit GPV Scheme

This paper aims to obtain an efficient multibit OT protocol via the multibit lossy encryption. But before designing the multibit lossy encryption scheme, we first present how to obtain the building block multibit GPV scheme of the lossy encryption scheme. Below, we follow the multibit FHE framework of Li et al. [22], and we first develop the multibit GPV (MGPV) scheme. Notably, most of existing LWE-based encryption schemes focus on how to enrich the functions of the single-bit encryption that was originally proposed by Regev [14], such as chosen-plaintext-attacker (CPA) secure schemes [10, 24], chosen-ciphertext-attacker (CCA) secure schemes [25, 26], fully homomorphic encryption (FHE) schemes [20], and oblivious transfer [10, 17].

Notably several recent works (such as [10, 27]) have formally shown the properties of multibit encryption. Fortunately, a multibit Regev [14] scheme was provided by Peikert et al. [10], which is called "pack ciphertext method" and was used as a crux tool to construct multibit FHE schemes [27]. Similarly, many works extended the work of Gentry-Peikert-Vaikutanathan (GPV) to the multibit scheme in the same way [18]. Along with this line, Lindner and Peikert [24] considered a new scheme under the multibit setting, where it is possible to encrypt multiple bits at one-time and makes PKE even more efficient. However, all of the mentioned schemes are constructed by a straightforward concatenation method with the inefficient performance. An important question is raised naturally.

> *Is it possible to explore a new method to design the multiple bits* GPV *encryption under the* LWE *assumption instead of the method of straightforward concatenation?*

We formally explore this important question in this section and we believe that the multibit GPV based on the public key with a sequence of LWE instances might offer many advantages over other approaches. The main ideas behind our method to design the MGPV scheme is described in following sections.

3.1. MGPV *Scheme.* Below we describe the MGPV scheme and its properties.

(i) $\mathsf{params} \longleftarrow \mathsf{MGPV.Setup}(1^\lambda)$:

(1) Take λ as input and output the common parameter $\mathsf{params} = (n, q, \chi, m, t)$, and let $l = \lfloor \log q \rfloor + 1$. We remark that this Setup algorithm is identical to the GPV [15] scheme except that we let a parameter t be the number of secret keys.

(ii) $(pk, sk) \longleftarrow \mathsf{MGPV.KeyGen}(\mathsf{params})$:

(1) Sample $\mathbf{e}_i \longleftarrow \chi^{n\times 1}$, $i \in [t]$ and output $sk_i := \mathbf{e}_i' \longleftarrow (\mathbf{I}_i, -\mathbf{e}_i^T)^T = (0, \cdots, 1, \cdots, 0 \mid -e_{i,1}, \cdots, -e_{i,n}) \in \chi^{n+t}$, and the i-th position is 1.

(2) Choose a matrix $\mathbf{B} \longleftarrow \mathbb{Z}_q^{m\times n}$ and compute $\mathbf{u}_i = \mathbf{B}\mathbf{e}_i \in \mathbb{Z}_q^{m\times 1}$, then we set $\mathbf{A} = [\mathbf{u}_1 \mid \cdots \mid \mathbf{u}_t \mid \mathbf{B}] \in \mathbb{Z}_q^{m\times(n+t)}$.

(3) Output $pk \longleftarrow \mathbf{A}$ and $sk \longleftarrow \{\mathbf{e}_1', \cdots, \mathbf{e}_t'\}$.

(iii) $\mathbf{c} \longleftarrow \mathsf{MGPV.Enc}(\mathsf{params}, pk, \mathbf{m})$:

(1) Set $\mathbf{m} := (m_1, \cdots, m_t)$, $m_i \in \{0, 1\}$ and define $\mathbf{m}' := (\mathbf{m} \mid \mathbf{0}) \in \mathbb{Z}_q^{1\times(n+t)}$.

(2) Sample $\mathbf{x}^T = (\mathbf{x}_1^T \mid \mathbf{x}_2^T) = (x_{1,1}, \cdots, x_{1,t} \mid x_{2,1}, \cdots, x_{2,n}) = (\mathbf{x}_1^T \longleftarrow \chi^{1\times t}, \mathbf{x}_2^T \longleftarrow \chi^{1\times n}) \in D_{\mathbb{Z}^{1\times(n+t)}}$, then choose $\mathbf{r} \longleftarrow \mathbb{Z}_q^{m\times 1}$.

(3) Compute $\mathbf{c} = \mathbf{A}^T \cdot \mathbf{r} + \lfloor q/2 \rfloor \cdot \mathbf{m}^T + \mathbf{x} \in \mathbb{Z}_q^{(n+t)\times 1}$, where the size of ciphertext is $O((n+t)\log^2 q)$.

(iv) $\mathbf{m}' \longleftarrow \mathsf{MGPV.Dec}(\mathsf{params}, sk, \mathbf{c})$:

(1) In order to make the reader understand the structure of the secret ky matrix $\mathbf{S} = (sk_1, \cdots, sk_t) \in \{0, 1\}^{(n+t)\times t}$, the detailed form of the matrix is as follows:

$$\mathbf{S} = \left(\mathbf{e}_1', \cdots, \mathbf{e}_t'\right) = \left(\begin{array}{c|c|c} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \\ \hline -e_{1,1} & \cdots & -e_{t,1} \\ \vdots & \ddots & \vdots \\ -e_{1,n} & \cdots & -e_{t,n} \end{array}\right). \tag{5}$$

(2) Then compute and output

$$\begin{aligned} \langle \mathbf{c}, \mathbf{S} \rangle &= \left\langle \mathbf{A}^T \cdot \mathbf{r} + \left\lfloor \frac{q}{2} \right\rfloor \cdot \mathbf{m}^T + \mathbf{x}^T, \mathbf{S} \right\rangle \\ &= \left\lfloor \frac{q}{2} \right\rfloor \cdot \mathbf{m}^T \cdot \mathbf{S} + \mathbf{x}^T \cdot \mathbf{S} \\ &= \left\lfloor \frac{q}{2} \right\rfloor \cdot \mathbf{m}' + \left(x_{1,1} - \mathbf{x}_2\mathbf{e}_1, \cdots, x_{1,t} - \mathbf{x}_2\mathbf{e}_t\right) \\ &\quad \left(\mathrm{mod}\, q\right). \end{aligned} \tag{6}$$

We note that the magnitude of the vector $(x_{1,1} - \mathbf{x}_2\mathbf{e}_1, \cdots, x_{1,t} - \mathbf{x}_2\mathbf{e}_t)$ can be regarded as the form of $t \cdot |(x_{1,i} + \mathbf{x}_2^T \cdot \mathbf{e}_i)|$ for $i \in [t]$. If $\|\langle \mathbf{c}, \mathbf{S}\rangle\| \leq t(n+1)B < q/4$, then set $m_i = 1$ and otherwise set $m_i = 0$. Output $\mathbf{m}' = (m_1, \cdots, m_t)$.

Remark 12. We stress that the ciphertext can be decrypted in a bit-by-bit manner. Once we have the secret key matrix $\mathbf{S}$, we can choose the i-th column of $\mathbf{S}$ to recover the i-th bit of the plaintext. In more detail,

(1) we use i-th column vector $\mathbf{s}_i$ from $\mathbf{S}$ to get the i-th position bit of message;

(2) compute and output $\langle \mathbf{c}, \mathbf{s}\rangle = \langle \mathbf{A}^T\mathbf{r} + \lfloor q/2 \rfloor \mathbf{m} + \mathbf{x}^T, \mathbf{s}\rangle = \lfloor q/2 \rfloor m_i + (x_{1,i} + \mathbf{x}_2^T \cdot \mathbf{e}_i) \pmod q$.

If $\|\langle \mathbf{c}, \mathbf{S}\rangle\| \leq (n+1)B < q/4$, then set $m_i = 1$ and otherwise set $m_i = 0$. Output m'.

3.2. Correctness. In this subsection, we analyze the magnitude of the noise.

Lemma 13 (correctness). *Consider the decryption algorithm decrypts in a bit-by-bit manner. If the ciphertext is* $\mathbf{c} = \mathbf{A}^T \cdot \mathbf{r} + \lfloor q/2 \rfloor \cdot \mathbf{m}^T + \mathbf{x}^T \pmod q \in \mathbb{Z}_q^{(m+t)\times 1}$ *under the i-th column secret key* $sk_i = \mathbf{e}_i' \in \mathbb{Z}_q^{(m+t)\times 1}$*, then we have that*

$$\begin{aligned} \langle \mathbf{c}, \mathbf{e}_i \rangle &= \left\lfloor \frac{q}{2} \right\rfloor \cdot m_i + \left(x_{1,i} + \mathbf{x}_2^T \cdot \mathbf{e}_i\right) \\ &= \left\lfloor \frac{q}{2} \right\rfloor \cdot m_i + error \pmod q, \end{aligned} \tag{7}$$

with $|error| < E \leq \lfloor q/2 \rfloor/2$*. Hence, for the secret key matrix* $sk = \mathbf{S} \in \mathbb{Z}_q^{(m+t)\times t}$*, we get the following result:*

$$\langle \mathbf{c}, \mathbf{S}\rangle = \left\lfloor \frac{q}{2} \right\rfloor \cdot \mathbf{m}' + \mathbf{x}^T \cdot \mathbf{S} = \left\lfloor \frac{q}{2} \right\rfloor \cdot \mathbf{m}' + t \cdot error \pmod q, \tag{8}$$

with $|t \cdot error| < t \cdot E \leq \lfloor q/2 \rfloor/2$*. Hence, there exists* $\mathbf{m}' \longleftarrow \mathsf{Dec}(sk, \mathbf{m})$.

Proof. Consider the following parameters $\mathbf{x}^T = (\mathbf{x}_1^T \longleftarrow \chi^{1\times t}, \mathbf{x}_2^T \longleftarrow \chi^{1\times m})$ and $\forall x_i \longleftarrow \chi$, $|x_i| \leq B$ (where $B \ll q$). Thus, we can get

$$\begin{aligned} \left\langle \mathbf{c}, \mathbf{e}_i' \right\rangle &= \mathbf{r}^T \cdot \mathbf{A}^T \cdot \mathbf{e}_i' + \left\lfloor \frac{q}{2} \right\rfloor \mathbf{m}^T \cdot \mathbf{e}_i' + \mathbf{x}^T \cdot \mathbf{e}_i' \\ &= 0 + \left\lfloor \frac{q}{2} \right\rfloor \cdot m_i + \left(x_{1,i} + \mathbf{x}_2^T \cdot \mathbf{e}_i\right) \\ &= \left\lfloor \frac{q}{2} \right\rfloor \cdot m_i + error_D \pmod q, \end{aligned} \tag{9}$$

with $\|error_D\| \leq \|x_{1,i}\| + \|\mathbf{x}_2^T \cdot \mathbf{e}_i\| \leq B + mB \leq E_D$; the norm of $x_{1,i} + \mathbf{x}_2^T \cdot \mathbf{e}_i$ is bounded by $(m+1)B$, where E_D is denoted as the norm of error elements.

Hence, we can easily obtain the result $\|t \cdot error_D\| \leq t \cdot E_D$ for $\langle \mathbf{c}, \mathbf{S}\rangle = \lfloor q/2 \rfloor \cdot \mathbf{m}' + t \cdot error_D \pmod q$. □

3.3. Security

Theorem 14. *Regarding the following two distributions* $\mathcal{X}$ *and* $\mathcal{Y}$*,*

(i) *the distribution* $\mathcal{X}$ *is denoted as matrices* $[\mathbf{u}_1 \mid \cdots \mid \mathbf{u}_t \mid \mathbf{B}]$ *on* $m \times (t+n)$*, where* $\mathbf{B} \in \mathbb{Z}_q^{m\times n}$ *is a uniform matrix for all* $1 \leq i \leq t$*,* $\mathbf{u}_i = \mathbf{B}\mathbf{e}_i \pmod q$*, and* $\mathbf{e}_i$ *is sampled from* χ^n*.*

(ii) *the distribution* $\mathcal{Y}$ *is denoted as the uniform on* $\mathbb{Z}_q^{m\times(t+n)}$*.*

If the (n, q, χ, m)-ISIS *assumption is hard for the parameters* $m > n \in \mathbb{N}$*,* $q \in \mathbb{N}$*,* $\chi \longleftarrow \mathbb{Z}$*, and* $t = O(\log(n))$ *being an integer, then the distribution* $\mathcal{X}$ *is computationally indistinguishable from* $\mathcal{Y}$*.*

The following theorem formalizes the key result used to show the security of MGPV scheme. We show the scheme is *IND-CPA* secure by using Theorem 14.

Theorem 15. *If the* ISIS *assumption and* LWE *assumption hold for the parameters* params $= (n, q, \chi, m, t)$*, then the* MGPV *scheme is IND-CPA-secure.*

Proof. The high-level proof is as follows:

(i) Firstly, armed with the ISIS assumption, the matrix $\mathbf{A} = [\mathbf{u}_1, \cdots, \mathbf{u}_t \mid \mathbf{B}] \in \mathbb{Z}_q^{n\times(m+t)}$ is computationally indistinguishable from a uniform random matrix by applying the Theorem 14.

(ii) Secondly, the matrix $\mathbf{Ar} + \mathbf{e}$ is indistinguishable from uniform under the LWE assumption and the leftover hash lemma.

This concludes the proof of the theorem. □

3.4. Oblivious Transfer via MGPV. In this subsection, we instantiate the OT protocol under the LWE assumption that provides security for the sender against an honest-but-curious receiver and security for the receiver against a cheating sender.

OT protocol contains two phases, as shown in Figure 1, the initialization phase (i.e., Setup) and the transfer phase (i.e., Transfer) [10].

(i) The Setup phase: the *sender* $\mathbf{S}$ owns 2 elements $\mathbf{m}_0$ and $\mathbf{m}_1$. The *receiver* samples a choice bit $b \in \{0, 1\}$.

(ii) The Transfer phase:

(1) At the beginning of each transfer, the *receiver* $\mathbf{R}$ has an input choice bit b, and he invokes the KeyGen$(\cdot)$ algorithm and outputs a pair (pk_b, sk_b), then he draws a vector as pk_{1-b} from the distribution $\mathbb{Z}_q^{n\times(n+t)}$, then $\mathbf{R}$ sends the pair (pk_0, pk_1) to the *sender* $\mathbf{S}$.

(2) Upon receiving the pair (pk_0, pk_1), the *sender* $\mathbf{S}$ inputs 2 elements $\mathbf{m}_0$ and $\mathbf{m}_1$ and invokes the Enc$(\cdot)$ algorithm to encrypt them under the

Receiver $\mathbf{R}(1^\lambda, b)$		*Sender* $\mathbf{S}$
$(pk_b, sk_b) \leftarrow \mathsf{KeyGen}(1^\lambda, b)$;		
Set $pk_{1-b} \leftarrow \mathbb{Z}_q^{n\times(n+t)}$	$\xrightarrow{(pk_0, pk_1)}$	*Draw* $\mathbf{m}_0, \mathbf{m}_1 \in \{0,1\}^t$
		$\mathbf{c}_0 \leftarrow \mathsf{Enc}(pk_0, \mathbf{m}_0)$
	$\xleftarrow{(\mathbf{c}_0, \mathbf{c}_1)}$	$\mathbf{c}_1 \leftarrow \mathsf{Enc}(pk_1, \mathbf{m}_1)$
$\mathbf{m}_b \leftarrow \mathsf{Dec}(sk_b, \mathbf{c}_b)$		

* $\mathscr{F}_{CRS}^{\mathscr{D}}$ runs with parties $P_1, \cdots, P_n$. The common reference string functionality $\mathscr{F}_{CRS}^{\mathscr{D}}$ [10]

Figure 1: Function $\mathscr{F}_{CRS}^{\mathscr{D}}$, * [10].

pk_0, pk_1, respectively, then outputs the ciphertext $\mathbf{c}_0$ and $\mathbf{c}_1$ and sends back to the receiver $\mathbf{R}$.

(3) Upon receiving the pair $\mathbf{c}_0, \mathbf{c}_1$, the receiver invokes the $\mathsf{Dec}(\cdot)$ algorithm and outputs $\mathbf{m}_b$.

3.5. Security. The OT protocol is a simple application of MGPV scheme.

Theorem 16 (see [16] Theorem 5.2). *We say that the OT protocol is secure for the receiver if the* $\mathsf{LWE}_{n,q,m,\chi}$ *problem is hard.*

Proof. The above security proof is simple, so we omit further details and recommend the reader to find further details from the proof of Theorem 5.2 in Lyubashevsky et al. at TCC'10 [16]. □

Theorem 17 (see [16] Theorem 5.3). *We say that the above OT protocol is secure for the sender against an honest-but-curious receiver, if the* $\mathsf{LWE}_{n,q,m,\chi}$ *assumption is hard for the input message length t of the sender.*

Proof. The detailed proof can be found from the Theorem 5.3 in Lyubashevsky et al. at TCC'10 [16]. □

4. Lossy Encryption (LE Scheme)

The notation of "lossy encryption" was proposed by Bellare-Hofheinz-Yilek (BHY) [11]. Actually, the lossy encryption is an extension of the meaningful/meaningless encryption [12] and dual mode encryption [10]. At a high level, a "lossy" (or "messy" in [10]) cryptosystem is one which also has two modes according to two types of public keys. Concretely, **(1)** in the normal mode, the ciphertext is generated by encrypting the plaintext under an injective key. **(2)** In the lossy (or "messy") mode, the ciphertext is independent of the plaintext. Actually, the operability property was proposed by [11]; they basically allow a possibly inefficient algorithm to open a ciphertext generated under a lossy key to any plaintext. Meanwhile, the injective key is computationally indistinguishable from the lossy key.

4.1. Multibit Lossy Encryption Scheme. Gentry et al. [15] proposed the dual Regev scheme to design the identity-based encryption (IBE) with the random oracle. Then, Agrawal et al. [28] used it to design the IBE scheme in the standard model. In this paper, we construct the LWE-based lossy encryption from multiple bits GPV. However, the process of encryption is different from GPV. In our construction, we only sample the noise vector $\mathbf{e}$ one-time rather than twice $\mathbf{AS}$ in the GPV scheme. The concrete construction is as follows:

(i) $(\mathsf{crs}, \mathsf{params}) \longleftarrow \mathsf{Setup}(1^\lambda)$:

(1) Set $m \geq 2n \cdot \log q$ and secure parameter λ. Since Lemma 10, we set $k \cdot \log q \leq n - 2 \cdot \lambda + 2$, $l \leq (k - 2\log(1/\varepsilon) - O(1))/\log q$, $q \geq 5rm$, $r \geq \omega(\sqrt{\log m})$, $\beta \leq 1/(r\sqrt{m} \cdot \omega(\sqrt{\log m}))$, $\beta \cdot q > O(2 \cdot \sqrt{n})$, and $\alpha/\beta = negl(\lambda)$. To satisfy these requirements, q should be superpolynomial of the secure parameter λ, moreover, t as described in MGPV scheme.

(2) Output $\mathsf{params} := (m, n, q, \chi, k, l, t)$ and $\mathsf{crs} := \mathbf{B}$, where $\mathbf{B} \longleftarrow \mathbb{Z}_q^{m\times n}$.

(ii) $(pk_{\mathsf{real}}, sk) \longleftarrow \mathsf{KeyGen}_{\mathsf{real}}(\mathsf{params})$:

(1) For $i \in [t]$, $\mathbf{e}_i \longleftarrow \chi^{n\times 1}$, then we have that $\mathbf{u}_i := \mathbf{A}\mathbf{e}_i \in \mathbb{Z}_q^{m\times 1}$. Compose all $\mathbf{e}_i$ together, then we have that $\mathbf{E} = [\mathbf{e}_1, \cdots, \mathbf{e}_t] \in \mathscr{D}_{\mathbb{Z},r}^{n\times t}$.

(2) Hence, $pk_{\mathsf{real}} := \mathbf{A} = [\mathbf{u}_1, \cdots, \mathbf{u}_t \mid \mathbf{B}] \in \mathbb{Z}_q^{m\times(n+t)}$.

(3) Output $sk := \mathbf{S} = [\mathbf{I} \mid \mathbf{E}]$ as described above.

(iii) $(pk_{\mathsf{lossy}}, \perp) \longleftarrow \mathsf{KeyGen}_{\mathsf{lossy}}(\mathsf{params})$:

(1) Choose $\mathbf{D} \longleftarrow \mathbb{Z}_q^{m\times k}$, $\mathbf{C} \longleftarrow \mathbb{Z}_q^{k\times n}$, $\mathbf{Z} \longleftarrow \mathscr{D}_{\mathbb{Z},\alpha\cdot q}^{m\times n}$, and $\mathbf{U} \in \mathbb{Z}_q^{m\times t} := [\mathbf{u}_i', \ldots, \mathbf{u}_t']$, where $\mathbf{u}_i' \longleftarrow \mathbb{Z}_q^{m\times 1}$.

(2) Output $sk := \perp$, $pk_{\mathsf{lossy}} := (\mathbf{u}_1', \cdots, \mathbf{u}_t' \mid \mathbf{DC} + \mathbf{Z})$.

(iv) $\mathbf{c} \longleftarrow \mathsf{Enc}(pk, \mathbf{m})$:

(1) Denote $\mathbf{m}' = [\mathbf{m} \mid \mathbf{0}] \in \{0,1\}^{1\times(n+t)}$.

(2) Choose random vectors $\mathbf{r} \longleftarrow \mathbf{Z}_q^{m\times 1}$, $\mathbf{x} \longleftarrow \mathscr{D}_{\mathbb{Z},\beta\cdot q}^{(n+t)\times l}$.

(3) Compute and output ciphertexts: $\mathbf{c} := \mathbf{A}^T \cdot \mathbf{r} + \lfloor q/2 \rfloor \mathbf{m}'^T + \mathbf{x} \in \mathbb{Z}_q^{(n+t)\times 1}$.

(v) $\mathbf{m}' \longleftarrow \mathsf{Dec}(sk, \mathbf{c})$: Compute and output: $\langle \mathbf{c}, \mathbf{S} \rangle = \lfloor q/2 \rfloor \mathbf{m} + \mathbf{x}^T \cdot \mathbf{S}$.

In order to construct the oblivious transfer, we need to design a verification algorithm $\mathsf{Verify}(\cdot)$ for the sender $\mathbf{S}$, who is similar to the $\mathsf{FindMessy}(\cdot)$ in [10], and will use the $\mathsf{Verify}(\cdot)$ to verify that the public key pk_σ from the receiver $\mathbf{R}$ is pk_{real} or pk_{lossy}, in more detail:

(i) $b \longleftarrow \mathsf{Verify}(pk_\sigma, \sigma)$: for $\sigma \in \{0,1\} := \{\mathsf{real}, \mathsf{lossy}\}$, the key generation takes a chosen decryptable branch $\sigma \in \{0,1\}$ as a parameter, and the resulting secret key sk_σ corresponds to branch σ of public key pk_σ. Then, we use $b \in \{0,1\}$ to distinguish the two messages. Actually, messages encrypted on branch $b = \sigma$ can be decrypted using sk_σ, while those on the other branch cannot.

Below, we show that this scheme fulfills the properties of lossy encryption.

Proposition 18. *Correctness on Real Keys. For all (pk_{real}, sk) generated by $\mathsf{KeyGen}_{\mathsf{real}}(1^\lambda)$ and all message $\mathbf{m}$,*

$$\begin{aligned}\mathsf{Dec}\left(sk, \mathsf{Enc}\left(pk_{\mathrm{real}}, \mathbf{m}\right)\right) &= \mathsf{Dec}\left(\mathbf{S}, \mathsf{Enc}\left(\mathbf{A}, \mathbf{m}'\right)\right)\\ &= \mathsf{Dec}\left(\mathbf{S}, \left(\mathbf{A}^T\mathbf{r} + \left\lfloor \frac{q}{2} \right\rfloor \mathbf{m}'^T + \mathbf{x}\right)\right)\\ &= \left\lfloor \frac{q}{2} \right\rfloor \cdot \mathbf{m} + error\end{aligned} \tag{10}$$

the algorithm $\mathsf{Dec}(\cdot)$ will get the correct message with overwhelming probability.

We need to remark that, considering the parameters $q \geq 5rm$ and $\beta \leq 1/(r\sqrt{m} \cdot \omega(\sqrt{\log q}))$ which were denoted in [15]. Then $\mathsf{Dec}(sk, \mathbf{c})$ decrypts correctly with overwhelming probability (over the random choices of $\mathsf{KeyGen}_{\mathsf{real}}(1^\lambda)$ and $\mathsf{Enc}(pk, \mathbf{m})$).

Proposition 19. *Lossiness of Encryption with Lossy Keys. In more detail*

$$\begin{aligned}\mathsf{Enc}\left(pk_{\mathsf{lossy}}, \mathbf{m}\right) &= \mathsf{Enc}\left(\underline{(\mathbf{u}_1, \ldots, \mathbf{u}_t \mid \mathbf{DC} + \mathbf{Z})}, \mathbf{m}'\right)\\ &= \underline{(\mathbf{u}_1, \ldots, \mathbf{u}_t \mid \mathbf{DC} + \mathbf{Z})}^T \cdot \mathbf{r} + \left\lfloor \frac{q}{2} \right\rfloor\\ &\quad \cdot \mathbf{m}' + \mathbf{x} \pmod q\end{aligned} \tag{11}$$

Parse public key $\mathbf{A}$ as $pk1 := (\mathbf{u}_1, \ldots, \mathbf{u}_t)$ and $pk2 := \mathbf{B}$, by Lemma 10, $\widetilde{\mathbf{H}}_\infty^\varepsilon(\mathbf{r} \mid \overline{\mathbf{B}}, \overline{\mathbf{B}} \cdot \mathbf{r} + \mathbf{x}) \geq m$ since $m \leq (\lambda - 2\log(1/\varepsilon) - O(1))/\log q$, and by Lemma 9, given $(\mathbf{DC}+\mathbf{Z})\cdot\mathbf{r}+\mathbf{x}$, $\underline{(\mathbf{u}_1, \ldots, \mathbf{u}_t)}^T \cdot \mathbf{r}$ is ε-close to $\mathcal{U}(\mathbb{Z}_q^{t\times 1})$. When $\varepsilon = negl(\lambda)$, $\underline{(\mathbf{u}_1, \ldots, \mathbf{u}_t)}^T \cdot \mathbf{r} \approx_s \mathcal{U}(\mathbb{Z}_q^{t\times 1})$ given $(\mathbf{DC} + \mathbf{Z}) \cdot \mathbf{r} + \mathbf{x}$. Therefore, $\forall \mathbf{m} \in \mathcal{M}$, given $(\mathbf{DC} + \mathbf{Z})\mathbf{r} + \mathbf{x}$, i.e.,

$$\underline{(\mathbf{u}_1, \ldots, \mathbf{u}_t)}^T \cdot \mathbf{r} + \left\lfloor \frac{q}{2} \right\rfloor \cdot \mathbf{m} \approx_s \mathcal{U}\left(\mathbb{Z}_q^{t\times 1}\right) \tag{12}$$

for any lossy keys pk_{lossy} generated by $\mathsf{KeyGen}_{\mathsf{lossy}}(1^\lambda)$ and any two messages $\mathbf{m}_0 \neq \mathbf{m}_1$, holds

$$\mathsf{Enc}\left(pk_{\mathsf{lossy}}, \mathbf{m}_0\right) \approx_s \mathsf{Enc}\left(pk_{\mathsf{lossy}}, \mathbf{m}_1\right) \tag{13}$$

Proposition 20. *Indistinguishability between Real Public Key and Lossy Public Key. pk_{real} is $(\underline{\mathbf{u}_1, \ldots, \mathbf{u}_t} \mid \mathbf{B})$, and pk_{lossy} is $(\underline{\mathbf{u}'_1, \ldots, \mathbf{u}'_t} \mid (\mathbf{DC} + \mathbf{Z}))$. Since $m \geq 2n\log q$, by Lemma 8, sample $\mathbf{U}_1 \longleftarrow \mathbb{Z}_q^{m\times t}$ and $\mathbf{U}_2 \longleftarrow \mathbb{Z}_q^{m\times n}$.*

$$\left(\underline{\mathbf{B}\cdot\mathbf{e}_1, \ldots, \mathbf{B}\cdot\mathbf{e}_t} \mid \mathbf{B}\right) \approx_s (\mathbf{U}_1, \mathbf{U}_2) \tag{14}$$

Under the hardness of LWE, $(\mathbf{U}_1, \mathbf{U}_2) \approx_c (\underline{\mathbf{u}_1, \ldots, \mathbf{u}_t} \mid \mathbf{DC} + \mathbf{Z})$,

$$\therefore \left(\underline{\mathbf{u}_1, \ldots, \mathbf{u}_t} \mid \mathbf{DC} + \mathbf{Z}\right) \approx_c \left(\underline{\mathbf{B}\cdot\mathbf{e}_1, \ldots, \mathbf{B}\cdot\mathbf{e}_t} \mid \mathbf{B}\right). \tag{15}$$

i.e., pk_{real} and pk_{lossy} are computationally indistinguishable.

5. OT via Lossy Encryption

In this section, inspired by David et al. [21] UC-secure OT protocol via lossy encryption using the McEliece assumption over code-based cryptography, we present an UC-secure OT protocol via lattice-based lossy encryption using LWE and ISIS assumption.

5.1. Our Construction: UC-Secure OT for Ideal Functionalities. Before describing our construction, we first denote the ideal functionalities $\mathcal{F}_{CRS}^{\mathcal{D}}$ and $\mathcal{F}_{\mathsf{OT}}$. In more detail, the CRS functionality $\mathcal{F}_{CRS}^{\mathcal{D}}$ outputs a string with a fixed distribution as depicted in Algorithm 1.

As shown in Algorithm 2, the two-party functionality OT contains a sender $\mathbf{S}$ with input x_0, x_1 and receiver $\mathbf{R}$ with an input $\sigma \in \{0,1\}$. Importantly, the OT functionality $\mathcal{F}_{\mathsf{OT}}$ captures requirement of OT specification.

5.2. Our Construction: OT from Lossy Encryption. Below we describe our main contribution, a various OT protocol from lossy encryption in Figure 2.

Simulating the communication with $\mathcal{Z}$: the simulator $\mathcal{S}$ writes every input value from $\mathcal{Z}$ into the input tape of the adversary $\mathcal{A}$. $\mathcal{S}$ copies every output value written by $\mathcal{A}$ to his output tape. The environment $\mathcal{Z}$ can read the output tap.

Simulating R is corrupted: $\mathcal{S}$ simulates the view of the receiver without considering which mode of the protocol and does the following: running the Setup algorithm in messy mode and letting $(\mathsf{crs}, t) \longleftarrow \mathsf{Setup}^{\mathsf{Lossy}}(1^\lambda)$. If the parties query the $\mathcal{F}_{CRS}^{\mathsf{mode}}$, then it obtains the feedback $(\mathsf{sid}, \mathsf{crs})$.

(i) Once the adversary $\mathcal{A}$ generates a message $(\mathsf{sid}, \mathsf{ssid}, pk_{\mathsf{real}}, pk_{\mathsf{lossy}})$, $\mathcal{S}$ extracts the choice bit b of the corrupted receiver and lets $b \longleftarrow \{0,1\}$, then $\mathcal{S}$ sends the command $(\mathsf{sid}, \mathsf{ssid}, \mathsf{receiver}, 1-b)$ to the $\mathcal{F}_{\mathsf{OT}}$, then $\mathcal{F}_{\mathsf{OT}}$ returns the output $(\mathsf{sid}, \mathsf{ssid}, \mathbf{m}_{1-b})$ to $\mathcal{S}$, and $\mathcal{S}$ then stores it along with b.

(ii) Once the dummy simulator $\mathbf{S}$ is activated by the command $(\mathsf{sid}, \mathsf{ssid})$, $\mathcal{S}$ then simulates the $\mathbf{S}$'s behaviour, and looks up the corresponding bit b for $\mathbf{m}_b$ and $\mathbf{m}_{1-b}$ and then computes $\mathbf{c}_b \longleftarrow \mathsf{Enc}(pk, \mathbf{0}^t)$ and $\mathbf{c}_{1-b} \longleftarrow \mathsf{Enc}(pk, \mathbf{m}_{1-b})$ and sends $(\mathsf{sid}, \mathsf{ssid}, \mathbf{c}_0, \mathbf{c}_1)$ to $\mathcal{A}$.

$\mathcal{F}_{CRS}^{\mathcal{D}}$ is parameterized by an algorithm $\mathcal{D}$, and $\mathcal{F}_{CRS}^{\mathcal{D}}$ can interact with parities $P_1, \cdots, P_n$.
(i) Upon receiving a command (sid, P_i, P_j) from the party P_i, first let $\mathsf{crs} \longleftarrow \mathcal{D}(1^\lambda)$, then send the message $(\mathsf{sid}, \mathsf{crs})$ to P_i and send the message (crs, P_i, P_j) to the adversary;
(ii) Upon receiving a command (sid, P_i, P_j) from the party P_j (and only P_j), then send P_j and the adversary the message $(\mathsf{sid}, \mathsf{crs})$, and halt.

Algorithm 1: The CRS functionality $\mathcal{F}_{CRS}^{\mathcal{D}}$ from [10].

$\mathcal{F}_{\mathsf{OT}}$ interacts with a receiver **R** and a sender **S**.
(i) Upon receiving a command $(\mathsf{sid}, \mathsf{sender}, x_0, x_1)$ from **S**, store the pair (x_0, x_1) for $x_i \in \{0,1\}^l$. (Notably, the length of the string l is fixed and all parties know);
(ii) Upon receiving a command $(\mathsf{sid}, \mathsf{receiver}, \sigma)$ from **R**, then check if $(\mathsf{sid}, \mathsf{sender}, \cdots)$ was previously sent and send the message (sid, x_σ) to **R**, and send the adversary $\mathcal{S}$ the message (sid) and halt. Otherwise, send nothing to **R**.

Algorithm 2: The oblivious transfer functionality $\mathcal{F}_{\mathsf{OT}}$.

Simulating S is corrupted: $\mathcal{S}$ does the following without considering which mode of the protocol: running the real (injective) mode KeyGen algorithm and letting $(\mathsf{crs}) \longleftarrow \mathsf{Setup}(1^\lambda)$. If the parties query the ideal functionality $\mathcal{F}_{CRS}^{mode}$, then $\mathcal{F}_{CRS}^{mode}$ returns $(\mathsf{sid}, \mathsf{crs})$ to them.

(i) Once the dummy **R** is activated on by the command $(\mathsf{sid}, \mathsf{ssid})$, $\mathcal{S}$ then simulates the behaviour of **R** and computes $(pk_{\mathsf{real}}, sk_{\mathsf{real}}) \longleftarrow \mathsf{KeyGen}(1^\lambda, inj)$ and $(pk_{\mathsf{lossy}}, sk_{\mathsf{lossy}}) \longleftarrow \mathsf{KeyGen}(1^\lambda)$, and then $\mathcal{S}$ sends $(\mathsf{sid}, \mathsf{ssid}, pk_{\mathsf{real}}, pk_{\mathsf{lossy}})$ to $\mathcal{A}$ and stores $(\mathsf{sid}, \mathsf{ssid}, pk_{\mathsf{real}}, sk_{\mathsf{real}}, sk_{\mathsf{lossy}})$.
(ii) When $\mathcal{A}$ replies with a message $(\mathsf{sid}, \mathsf{ssid}, \mathbf{c}_0, \mathbf{c}_1)$, the $\mathcal{S}$ looks up the corresponding $(pk_{\mathsf{real}}, sk_{\mathsf{real}})$ and $(pk_{\mathsf{lossy}}, sk_{\mathsf{lossy}})$, computes $\mathbf{m}_b \longleftarrow \mathsf{Dec}(sk_b, \mathbf{c}_b)$ for each $b \in \{0,1\}$, and returns $(\mathsf{sid}, \mathsf{ssid}, \mathsf{sender}, \mathbf{m}_0, \mathbf{m}_1)$ to $\widehat{F}_{OT}$.

Simulating the remaining cases: once both parties are corrupted by the adversary, then $\mathcal{S}$ runs $\mathcal{A}$. More concretely, $\mathcal{S}$ internally runs the **S** on input $(\mathsf{sid}, \mathsf{ssid}, \mathbf{m}_0 = \mathbf{0}^t, \mathbf{m}_1 = \mathbf{0}^t)$; meanwhile, it runs the honest **R** on input $(\mathsf{sid}, \mathsf{ssid}, \sigma = 0)$ and honest no matter which party is corrupted. When the corresponding dummy party is activated in the ideal execution, **S** activates the appropriate algorithm and delivers $\mathcal{A}$ all messages between its internal **R** and **S**.

Caim. If $\mathcal{A}$ corrupts **R** in an execution of $\mathsf{LE}^{\mathsf{lossy}}$, i.e., **S** in lossy mode, then we have

$$\mathsf{IDEAL}_{\widehat{\mathcal{F}}_{\mathsf{OT}}, \mathcal{S}, \mathcal{Z}} \approx_s \mathsf{EXEC}_{\mathsf{LE}^{\mathsf{lossy}}, \mathcal{A}, \mathcal{Z}} \tag{16}$$

Proof. Below we give a formal proof, in more detail:

(i) The real world execution can be viewed as the proceed of the following game.
  (a) Firstly, obtain crs by invoking the algorithm $\mathsf{Setup}_1^{\mathsf{lossy}}(1^\lambda)$.
  (b) Secondly, the environment $\mathcal{Z}$ can schedule subsessions arbitrarily. Notably, in each subsession,
    (1) $\mathcal{Z}$ can choose an arbitrary message $(\mathbf{m}_0, \mathbf{m}_1)$ for the honest sender **S**;
    (2) the honest sender **S** sends the ciphertext $\mathbf{c}_b \longleftarrow \mathsf{Enc}(pk, \mathbf{0}^t)$ for each $b \in \{0,1\}$ to $\mathcal{Z}$.

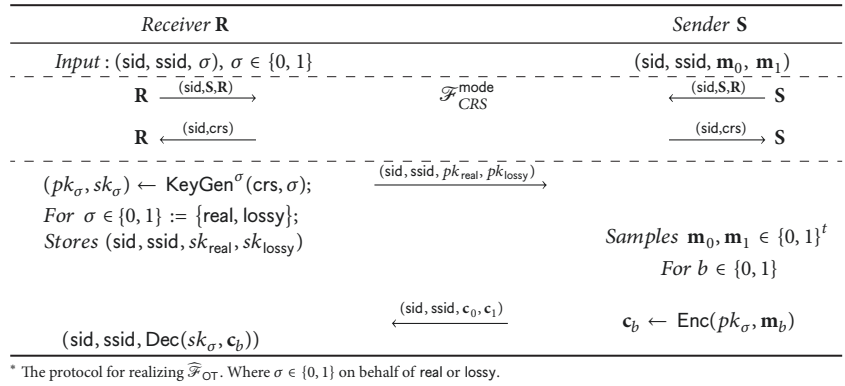
(ii) The ideal world execution can be viewed as the proceed of the following game:
  (a) Firstly, obtain crs by running the algorithm $\mathsf{Setup}^{\mathsf{lossy}}(1^\lambda)$.
  (b) Secondly, the environment $\mathcal{Z}$ schedules subsessions arbitrarily, in each subsession,
    (1) $\mathcal{Z}$ can input an arbitrary pk and the arbitrary input message $(\mathbf{m}_0, \mathbf{m}_1)$ for the dummy sender **S**;
    (2) $\mathcal{S}$ can run the verification algorithm $b \longleftarrow \mathsf{Verify}(\mathsf{crs}, pk_\sigma, \sigma)$ to $\mathcal{Z}$ and learn $\mathbf{m}_{1-b}$ from $\widehat{\mathcal{F}}_{\mathsf{OT}}$. It then sends $\mathcal{Z}$ the $\mathbf{c}_b \longleftarrow \mathsf{Enc}(\mathsf{crs}, pk, 1-b, \mathbf{0}^t)$ and $\mathbf{c}_{1-b} \longleftarrow \mathsf{Enc}(\mathsf{crs}, pk, 1-b, \mathbf{m}_{1-b})$.

We stress that the only difference between the ideal world execution and the real world execution is the generation of $\mathbf{c}_b$ in each subsession. But by lossy key generation in Proposition 19, hence the above two games are statistically indistinguishable. □

Claim. If, in an execution of $\mathsf{LE}^{\mathsf{real}}$ (i.e., **S** in real mode), $\mathcal{A}$ corrupts **S** then we have

$$\mathsf{IDEAL}_{\widehat{\mathcal{F}}_{\mathsf{OT}}, \mathcal{S}, \mathcal{Z}} \approx_s \mathsf{EXEC}_{\mathsf{LE}^{\mathsf{real}}, \mathcal{A}, \mathcal{Z}}. \tag{17}$$

Proof. Below we give a formal proof, in more detail:



* The protocol for realizing $\widetilde{\mathcal{F}}_{\mathrm{OT}}$. Where $\sigma \in \{0,1\}$ on behalf of real or lossy.

Figure 2: Protocol $\mathsf{LE}^{\mathsf{mode}}$ for oblivious transfer *.

(i) The real world execution can be viewed as the proceed of the following game:

(a) Firstly, $\mathsf{crs} \longleftarrow \mathsf{Setup}_1^{\mathsf{real}}(1^\lambda)$.

(b) Secondly, the environment arbitrarily schedules some number of subsessions. In each subsession,

(1) $\mathcal{Z}$ chooses an input σ for the honest $\mathbf{R}$, who generates $(pk_\sigma, sk_\sigma) \longleftarrow \mathsf{KeyGen}(\mathsf{crs}, \sigma)$, and sends pk_{real}, pk_{lossy} to $\mathcal{Z}$;

(2) then $\mathcal{D}$ proceeds arbitrarily $(\mathbf{c}_1, \mathbf{c}_2)$ to the honest $\mathbf{R}$ outputs $\mathsf{Dec}(\mathsf{crs}, sk_\sigma, \mathbf{c}_\sigma)$.

(ii) The ideal world execution can be viewed as the proceed of the following game:

(a) Firstly, $(\mathsf{crs}, t) \longleftarrow \mathsf{Setup}^{\mathsf{Dec}}(1^\lambda)$.

(b) Secondly, the environment arbitrarily schedules subsessions, in each subsession,

(1) $\mathcal{Z}$ outputs arbitrary σ which is not known to $\mathbf{S}$;

(2) $\mathcal{S}$ then runs $(pk_{\mathsf{real}}, sk_{\mathsf{real}}) \longleftarrow \mathsf{KeyGen}(\mathsf{crs}, \sigma)$ and $(pk_{\mathsf{lossy}}, sk_{\mathsf{lossy}}) \longleftarrow \mathsf{KeyGen}(\mathsf{crs}, 1-\sigma)$ and sends pk_{real}, pk_{lossy} to $\mathcal{Z}$;

(3) lastly, $\mathcal{S}$ receives the arbitrary ciphertext tuple $(\mathbf{c}_0, \mathbf{c}_1)$ from $\mathcal{Z}$.

□

Remark 21. Actually, the dummy entity $\mathbf{R}$ queries the value of $\mathsf{Dec}(\mathsf{crs}, sk_\sigma, \mathbf{c}_\sigma)$ from the ideal functionality, then the simulator $\mathcal{S}$ provides the messages $\mathbf{m}_b \longleftarrow \mathsf{Dec}(\mathsf{crs}, sk_b, \mathbf{c}_b)$.

The only difference between the two games is method of the public and secret keys. The above two games are statistically indistinguishable by the lossy key generation in Proposition 20.

Claim. There exists the following result:

$$\mathsf{EXEC}_{\pi^{\mathsf{lossy}}, \mathcal{A}, \mathcal{Z}} \approx_c \mathsf{EXEC}_{\pi^{\mathsf{real}}, \mathcal{A}, \mathcal{Z}} \tag{18}$$

for any protocol π^{mode} in the $\mathcal{F}_{CRS}^{\mathsf{mode}}$-hybrid model.

Proof. In the lossy encryption, the output of $\mathcal{F}_{CRS}^{\mathsf{lossy}}$ is computationally indistinguishable from $\mathcal{F}_{CRS}^{\mathsf{real}}$ since the indistinguishability of modes. Moreover, $\mathcal{Z}$ can run the protocol π^{mode} and can receive a polynomial number of samples from either $\mathcal{F}_{CRS}^{\mathsf{lossy}}$ or $\mathcal{F}_{CRS}^{\mathsf{real}}$. Thus, the above two executions are indistinguishable by a standard hybrid argument. □

5.3. Performance. Lattice-based cryptography has been subjecting of intense research appearing recently, bringing groundbreaking advance to the understanding of the subjacent questions. One of the main characteristics of lattice-based cryptography is worst-case to average-case reductions, which provides stronger security against quantum computer attacks. In this paper, we construct a lossy encryption scheme via a variant of multibit GPV scheme, then we construct the universal composable secure OT protocols based on LWE assumption by utilizing the lossy encryption as the building block. Below, a comparison of some related works with our scheme is provided in the Table 1.

As shown in the Table 1, we can easily obtain the following conclusion. We follow the methodology of Li et al. [18] and design a multibit public key encryption scheme, i.e., MGPV scheme. Importantly, the public matrix $\mathbf{A}$ contains many LWE instances; each one is used to protect the secret key. In this setting, the decrypter can decrypt the plaintext either in a bit-by-bit manner or in a one-time manner. Meanwhile, compared the magnitude of ciphertext of PVW scheme [10] with ours, it is easy to see the two schemes with the same magnitude of ciphertext $O(\log_q mn)$. Although the public key size of PVW depends on the parameter n and our scheme's public key size depends on m, the bit decryption of our scheme implies flexible decryption (i.e., multibit decryption), which means that our scheme is more practical in reality.

6. Potential Application: Password-Based Authenticated Key Exchange for Smart Mobile Devices

Nowadays, SMDs, IoTs, and WSNs within the workplace are expanding rapidly. Obliviously, these devices are becoming important tools that offer competitive advantages for the mobile workforce. But they also might be endangered by the

Table 1: The comparison of some related works with our scheme.

Scheme	assumption	message size	bit Dec	one-time Dec	applications
GPV [15]	LWE/SIS	1	✔	✘	PKE&IBE
PVW [10]	LWE	1	✘	✔	PKE&OT
LPS [16]	SSP	t	✔	✔	PKE&OT
Our scheme	LWE	t	✔	✔	PKE&OT

information they can access remotely. In this case, enabling user authentication for SMDs is the first line of defence to prevent the malicious unauthorized user.

Most of related works [5, 29] focus on how to use PAKE as the basic tool to achieve the authentication for SMDs. In particular, Wei et al. [5] proposed a PAKE protocol for wireless body area networks. He et al. [29] proposed an authentication protocol for mobile wireless networks with conditional privacy preservation. However, to our knowledge, related works of lattice-based PAKE for SMDs authentication are limited. Hence, in this section, we explore how to implement PAKE via our OT protocol. Because details of the design and implementation are beyond the scope of the discussion of this paper, thus, we just give a brief of description for the technical line as follows. In more detail, following the technical line of Canetti et al. [1], we first realize OT-based PKAE via LWE assumption instead of computational Diffie-Hellman (CDH) assumption and the hardness of factoring. Next, we can extend the PAKE protocol for privacy-preserving authentication schemes for SMDs.

7. Conclusion

In this paper, we have investigated one of the hot but hard topics in authentication of SMDs, IoTs, and WSNs. As an important building block, OT can be used for designing privacy-preserving authentication protocols. Thus, we focus on an important question how to design on an efficient UC-secure OT protocol for PAKE which can be used to achieve authentication for SMDs. However, an important question that remain is how to implement OT-based PAKE under the LWE assumption following our presented brief technical line. Meanwhile, we believe that this result enriched the postquantum OT protocols. However, it remains open to be secure against adaptive adversaries under the lossy encryption and its variants. We leave these topics for future research.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The research was supported by the National Natural Science Foundation of China (nos. 61802214 and 11701187) and the PhD Start-up Fund of the Natural Science Foundation of Guangdong Province of China (no. 2017A030310522).

References

[1] R. Canetti, D. Dachman-Soled, V. Vaikuntanathan, and H. Wee, "Efficient password authenticated key exchange via oblivious transfer," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 7293, pp. 449–466, 2012.

[2] D. Wang, N. Wang, P. Wang, and S. Qing, "Preserving privacy for free: Efficient and provably secure two-factor authentication scheme with user anonymity," *Information Sciences*, vol. 321, Article ID 11496, pp. 162–178, 2015.

[3] C. Su, B. Santoso, Y. Li, R. H. Deng, and X. Huang, "Universally Composable RFID Mutual Authentication," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 1, pp. 83–94, 2017.

[4] Y. Zhang, J. Yu, R. Hao, C. Wang, and K. Ren, "Enabling Efficient User Revocation in Identity-based Cloud Storage Auditing for Shared Big Data," *IEEE Transactions on Dependable and Secure Computing*, pp. 1-1, 2018.

[5] F. Wei, P. Vijayakumar, J. Shen, R. Zhang, and L. Li, "A provably secure password-based anonymous authentication scheme for wireless body area networks," *Computers and Electrical Engineering*, vol. 65, pp. 322–331, 2018.

[6] L. Zhang, Z. Zhang, and X. Hu, "UC-secure two-server password-based authentication protocol and its applications," in *Proceedings of the 11th ACM Asia Conference on Computer and Communications Security, ASIA CCS 2016*, pp. 153–164, June 2016.

[7] M. O. Rabin, "Probabilistic algorithms in finite fields," *SIAM Journal on Computing*, vol. 9, no. 2, pp. 273–280, 1980.

[8] J. Han, W. Susilo, Y. Mu, M. H. Au, and J. Cao, "AAC-OT: Accountable Oblivious Transfer with Access Control," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2502–2514, 2015.

[9] Y. Gertner, S. Kannan, T. Malkin, O. Reingold, and Viswanathan. M., "The relationship between public key encryption and oblivious transfer," in *Proceedings of the FOCS 2000*, pp. 325–335, IEEE Computer Society Press.

[10] C. Peikert, V. Vaikuntanathan, and B. Waters, "A framework for efficient and composable oblivious transfer," Cryptology ePrint Archive 2007/348, https://eprint.iacr.org/2007/348.

[11] M. Bellare, D. Hofheinz, and S. Yilek, "Possibility and impossibility results for encryption and commitment secure under selective opening," in *Advances in cryptology EUROCRYPT 2009*, vol. 5479, pp. 1–35, Springer, Berlin, Germany, 2009.

[12] G. Kol and M. Naor, "Cryptography and game theory: designing protocols for exchanging information," in *Theory of Cryptography: Fifth Theory of Cryptography Conference, TCC 2008, New York, USA, March 19–21, 2008. Proceedings*, vol. 4948

of *Lecture Notes in Computer Science*, pp. 320–339, Springer, Berlin, Germany, 2008.

[13] P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings of the 35th Annual Symposium on Foundations of Computer Science (SFCS '94)*, pp. 124–134, IEEE Computer Society Press, 1994.

[14] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," in *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC '05)*, pp. 84–93, ACM Press, Baltimore, Md, USA, May 2005.

[15] C. Gentry, C. Peikert, and V. Vaikuntanathan, "Trapdoors for hard lattices and new cryptographic constructions," in *Proceedings of the 14th Annual ACM Symposium on Theory of Computing (STOC '08)*, pp. 197–206, Victoria, Canada, May 2008.

[16] V. Lyubashevsky, A. Palacio, and G. Segev, "Public-key cryptographic primitives provably as secure as subset sum," in *Theory of cryptography*, vol. 5978, pp. 382–400, Springer, Berlin, Germany, 2010.

[17] B. David, R. Dowsley, and A. Nascimento, "Universally composable oblivious transfer based on a variant of LPN," in *Proceedings of the CANS 2014*, pp. 143–158, Springer.

[18] Z. Li, C. Ma, and H. Zhou, "Multi-key FHE for multi-bit messages," *Science China Information Sciences*, vol. 61, article 029101, pp. 1–3, 2018.

[19] Z. Li, S. D. Galbraith, and C. Ma, "Preventing Adaptive Key Recovery Attacks on the Gentry-Sahai-Waters Leveled Homomorphic Encryption Scheme," Cryptology ePrint Archive 2016/1146, https://eprint.iacr.org/2016/1146.

[20] Z. Brakerski and V. Vaikuntanathan, "Efficient fully homomorphic encryption from (standard) LWE," in *Proceedings of the IEEE 52nd Annual Symposium on Foundations of Computer Science (FOCS '11)*, pp. 97–106, Palm Springs, CA, USA, October 2011.

[21] B. M. David, A. C. A. Nascimento, and J. Müller-Quade, "Universally composable oblivious transfer from lossy encryption and the McEliece assumptions," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 7412, pp. 80–99, 2012.

[22] Z. Li, C. Ma, and D. Wang, "Leakage Resilient Leveled FHE on Multiple Bit Message," *IEEE Transactions on Big Data*, pp. 1-1.

[23] R. Canetti, "Universally composable security: A new paradigm for cryptographic protocols," in *Proceedings of the FOCS*, pp. 136–145, IEEE Computer Society Press, 2001.

[24] R. Lindner and C. Peikert, "Better key sizes (and attacks) for LWE-based encryption," Cryptology ePrint Archive 2010/613, https://eprint.iacr.org/2010/613.

[25] C. Peikert, "Public-key cryptosystems from the worst-case shortest vector problem: extended abstract," in *Proceedings of the STOC 2009*, pp. 333–342, ACM Press.

[26] C. Peikert and B. Waters, "Lossy trapdoor functions and their applications," in *Proceedings of the 40th Annual ACM Symposium on Theory of Computing (STOC '08)*, pp. 187–196, ACM Press, 2008.

[27] R. Hiromasa, M. Abe, and T. Okamoto, "Packing messages and optimizing bootstrapping in GSW-FHE," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E99A, no. 1, pp. 73–82, 2016.

[28] S. Agrawal, D. Boneh, and X. Boyen, "Efficient lattice (H)IBE in the standard model," in *Advances in Cryptology—EUROCRYPT 2010*, vol. 6110 of *Lecture Notes in Computer Science*, pp. 553–572, Springer, Berlin, Germany, 2010.

[29] D. He, D. Wang, Q. Xie, and K. Chen, "Anonymous handover authentication protocol for mobile wireless networks with conditional privacy preservation," *Science China Information Sciences*, vol. 60, no. 5, Article ID 052104, 2017.