Proof of Plaintext Knowledge for the Ajtai-Dwork Cryptosystem

Shafi Goldwasser[1,2], Dmitriy Kharchenko[2]

[1] CSAIL, Massachusetts Institute of Technology. Cambridge, MA 02139, USA.
[2] Department of Computer Science and Applied Mathematics,
Weizmann Institute of Science, Rehovot 76100, Israel.

Abstract. Ajtai and Dwork proposed a public-key encryption scheme in 1996 which they proved secure under the assumption that the unique shortest vector problem is hard in the worst case. This cryptosystem and its extension by Regev are the only one known for which security can be proved under a worst case assumption, and as such present a particularly interesting case to study.
In this paper, we show statistical zero-knowledge protocols for statements of the form "plaintext m corresponds to ciphertext c" and "ciphertext c and c' decrypt to the same value" for the Ajtai-Dwork cryptosystem. We then show a interactive zero-knowledge proof of plaintext knowledge (PPK) for the Ajtai-Dwork cryptosystem, based directly on the security of the cryptosystem rather than resorting to general interactive zero-knowledge constructions. The witness for these proofs is the randomness used in the encryption.
Key Words: Lattices, Verifiable Encryption, Ajtai-Dwork Cryptosystem, Worst Case Complexity Assumption, Proof of Plaintext Knowledge.

1 Introduction

There is much to celebrate in the progress made by cryptography on many fronts: rigorous definitions of security of natural cryptographic tasks, constructions of schemes achieving security based on general assumptions, new and seemingly contradictory possibilities such as zero-knowledge proofs and secure multi-party computations.

Still, during all this time, the implementations of this progress or rather the assumptions that underly all implementations, remain almost exclusively the intractability of factoring integers and of computing discrete logarithms which go back to the original papers of [9, 25] (often even stronger versions of these assumptions are utilized to gain better efficiency, such as higher quadratic residuosity, DDH, Strong-RSA). There are a couple of exceptions: computational problems over Elliptic Curves and computational problems over Integer Lattices. Whereas the computational problems over Elliptic curves do not seem to be inherintely harder than the analogous problems over finite fields, the use of computational problems over lattices seem to present a new frontier. Due to the pioneering work of Ajtai[], these problems certainly show the greatest promise from a *theoretical treatment* point of view.

In particular, in 1996, Ajtai and Dwork proposed [1] a public-key cryptosystem which is secure under the assumption that the unique shortest vector problem in integer lattices is hard in the worst case. The Ajtai-Dwork cryptosystem (and its extension by Regev [24]) are the only known public-key cryptosystems with the property that breaking a random instance of it is as hard as solving the worst-case instance of problem on which the system security is based. As such it present a particularly interesting and unique system to study from a complexity theoretic point of view.

Much study has been dedicated to the number theory based encryption systems (e.g. Cramer-Shoup, Paillier, RSA), showing how to incorporate them efficiently into larger protocols (e.g. designated confirmer signatures, e-cash protocols), extending their basic functionality (e.g. threshold decryption, verifiable encryption, group encryption, key-escrow versions), and extending them to achieve stronger security definitions (e.g. chosen cipher-text security, interactive encryption with efficient proofs of plaintext knowledge).

In contrast, the work on AD cryptosystems has been restricted to attempting cryptanalysis of the original scheme([23], showing chosen cipher text attacks [19], and proofs tightening the worst case versus average security reductions [24]. To date, there has been no protocol work involving the usage of AD encryption.

We can only speculate why this study is missing. Possibly, since the mathematics underlying the AD systems seemingly does not lend itself to simple treatment as in the case of the number theoretic schemes. Possibly, because AD is viewed largely of interest as a theoretical case study rather than one envisioned useful within other application. Or, perhaps, because it is a secondary order concern which naturally will follows the basic study of security. In any case, as by enlarge all existing number theoretic cryptosystems stand and fall together whereas the security of AD seems unrelated and could hold even if the former does not, we feel it is time to begin such treatment. Certainly, we will only be able develop intuition about the usability of this system, by attempting to do so. We initiate this study in this paper.

We begin with investigating very simple questions, which seem fundamental to many applications of public-key encryption schemes.

- First, we show how AD can be augmented to be a verifiable encryption scheme, by providing statistical zero knowledge proofs for basic statements about the plaintext of AD ciphertexts, such as 'ciphertexts c and c' decrypt to the same plaintext' and 'ciphertext c decrypts to plaintext m'. The witness for these proofs is the randomness used in the encryption.
- Second, we show a zero-knowledge interactive proof of plaintext knowledge for AD ciphertexts. Again the witness for this proof is the randomness used in the encryption. The construction is simple and direct, and does not utilize general ZK interactive proof constructions or general tools such as the existence of one-way functions. Rather it exploits the statistical zero knowledge protocols constructed above to prove statements which arise within the

interactive proof of plaintext knowledge. The computational zero knowledge property is proved assuming the security of the AD cryptosystem itself. The existence of a zero-knowledge interactive proof of plaintext knowledge, establishes in turn an interactive encryption variant of AD cryptosystem which is CCA1 secure ([13, 16]) costing reasonable overhead beyond the complexity of AD encryption itself. In contrast Hall, Goldberg, Schneier [19] showed that the secret key of the AD cryptosystem can be recovered using a CCA1 attack.

Previously, computational zero knowledge protocols for all the statements we prove were only known by utilizing general ZK interactive proofs for NP [17].

Throughout our work, instead of using the original Ajtai-Dwork construction which has non-zero decryption error probability, we use the decryption-error-free variant of Goldreich, Goldwasser and Halevi [15]. The semantic security of the modified cryptosystem holds under under the same assumption as the original cryptosystem. We refer to it as the AD cryptosystem throughout.

We make technical use of two prior works. The work of Micciancio and Vadhan[21] which shows a statistical zero-knowledge protocol with efficient provers for approximate versions of the SVP and CVP problems where the witness is a short vector in a lattice (or a point close to the target in the CVP case). And the work of Nguyen and Stern [23] which show how to use a CVP oracle to cryptanalyze the AD cryptosystem. Although Nguyen and Stern's work was aimed at cryptanalysis and showed that AD cryptosystem is no harder to break than the CVP problem, we use it as a positive result, using it as a tool to generate 'good instances' of an AD public key and ciphertexts for our verifiable encryption protocols for which our protocols will work. This continues the traditional pattern of research on lattices in cryptography, where progress on lattice research is used on one hand to cryptanalyze existing schemes and on the other hand to provide security proofs for lattice based cryptographic schemes.

We proceed to elaborate on related work and concepts, and our results in some detail.

1.1 Related Results and Conepts

VERIFIABLE ENCRYPTION. *Verifiable encryption* was introduced by Stadler in [26] in the context of publicly verifiable secret sharing, and in more general form by Asokan, Shoup and Waidner in [2] for the purpose of fair exchange of digital signatures. In the verifiable encryption setting, there are three parties. A party who generates the secret/public key pair (PK, SK), an encryptor which we refer to as the prover who creates a ciphertext of some plaintext, and a verifier who on input a public-key and a ciphertext verifies some application-driven properties of the plaintext. Verifiable encryption is defined with respect to some binary relation R defined on plaintext messages. Informally, a *verifiable encryption* with respect to relation R is a zero-knowledge protocol which, on public inputs ciphertext c, δ, and PK allows a prover to convince a verifier that the ciphertext c is an encryption of a message m with public key PK such that $(m, \delta) \in R$ (as in [4]). The prover uses the randomness which was used to generate the ciphertext c as auxilary input.

By using zero knowledge interactive proofs for NP [17], it is clearly possible to turn all known encryption schemes into verifiable encryption schemes for *any* $R \in NP$. However, for specific relations R of interest we may be able to get much more efficient protocols, with stronger security properties (e.g. statistical vs. computational zero-knowledge). For example, in recent work of Camenisch and Shoup [5], they propose a modification of the Cramer-Shop cryptosystem [7] based on the Paillier's decision composite residuosity assumption, for which they show an efficient verifiable encryption scheme for the relation $R = \{(m, (\delta, \gamma)) | \gamma^m = \delta\}$. Namely, they demonstrate efficient statistical ZK proofs on input a public key, ciphertext c (of the modified encryption scheme), and γ, δ pair, that c is the encryption of an m for which $\gamma^m = \delta$.

PLAINTEXT PROOFS OF KNOWLEDGE Given an instance of a public-key encryption scheme with public key pk, a proof of plaintext knowledge(PPK) allows an encryptor (or prover) to prove knowledge of the plaintext m of some ciphertext $C \in E_{pk}(m)$ to a receiver. A proof of plaintext knowledge should guarantee that no additional knowledge about m is revealed to the receiver or an eavesdropper. Customarily, this requirement is captured by requiring the plaintext proof of knowledge to be a zero-knowledge proof.

For the Rabin, RSA, Goldwasser-Micali, Paillier, El-Gamal encryption schemes, well known 3-round zero-knowledge public-coin proofs of knowledge protocols (often referred to as Σ protocols) can be easily adapted to achieve efficient PPKs.

When both the sender and the receiver are on-line, interactive public-key encryption protocols may be used. Starting with an underlying semantically secure public-key encryption scheme which has a zero-knowledge proof of plaintext knowledge, the sender of the ciphertext c in addition engages in a proof of plaintext knowledge with the receiver. The result is a CCA1 secure public-key encryption scheme [13, 16]. Utilizing efficient PPKs for specific number theoretic based semantically secure public-key encryption schemes such as the Blum-Goldwasser, Paillier, and El Gamal scheme, thus yields efficient CCA1 secure interactive public-key encryption variants of these schemes. Better yet, Katz[20] shows how design efficient interactive *non-malleable* proofs of plaintext knowledge for the RSA, Rabin, Paillier, and El-Gamal encryption schemes. Using these, one obtains efficient CCA2 secure interactive public-key encryption variants of the underlying schemes.

Naturally, if one-way functions exist, PPKs can be achieved using completeness results [17] for interactive zero-knowledge proofs for NP, proofs of knowledge for NP[12], and non-malleable interactive zero knowledge PPK for NP[8]. However, these general constructions are prohibitively inefficient as they require as a preliminary step polynomial time reductions to instances of NP-complete problems.

For the Ajtai-Dwork cryptosystem, these general completeness constructions of PPK were the only one knows prior to our work.

Finally, we note that in contrast to the interactive case, known constructions of *non-interactive* zero-knowledge proofs (NIZK) [8] for NP languages (which are a central tool in constructing CCA2 secure non-interactive public-key encryption given semantically secure public-key encryption algorithms) require trapdoor permutations. The intractability assumption on which the security of the Ajtai-Dwork cryptosystem is based, however, is not known to imply the existence of trapdoor permutations. It remains a central open problem to find a non-interactive CCA2 secure public-key encryption algorithm (efficient or otherwise) based on the AD-cryptosystem assumption.

LATTICE TOOLS. Our work uses as tools the results of [21] and [23]. In [21] Micciancio and Vadhan provide a zero-knowledge proof system for the GapCVP_γ problem for $\gamma = \Omega(\sqrt{\frac{n}{\log(n)}})$ where n is the dimension of the lattice. An instance of the GapCVP_γ is a triple consisting of a lattice L, a vector x and a value t. An instance is a YES instance if the distance between the vector x and the lattice L is less than t. If the distance is greater than γt the instance is a NO instance. In the proof zero-knowledge system of Micciancio and Vadhan [21] a prover proves to a verifier that an instance of the GapCVP_γ is a YES instance. If the instance is NO instance, the verifier rejects with high probability.

Nguyen and Stern showed in [23] how to use a CVP oracle to distinguish between ciphertexts of '0' and '1' of the Ajtai-Dwork cryptosystem (with decryption errors). For a random public key and a random ciphertext of the Ajtai-Dwork cryptosystem, Nguyen and Stern construct some lattice L and some vector x. They show that for ciphertexts of '0' the distance between the lattice L and the vector x is likely to be small, whereas for ciphertexts of '1' the distance is likely to be large.

1.2 Our Results in Detail

VERIFIABLE ENCRYPTION FOR THE AD CRYPTOSYSTEM. The first result of this paper is the design of statistical zero-knowledge protocol for proving that ciphertexts decrypt to given plaintexts for the AD public key cryptosystems. Namely, on public inputs ciphertext c, δ, and public-key PK a verifiable encryption scheme for the *equivalence relation* $R = \{(m, \delta) | m = \delta\}$.

The encryption method of Ajtai and Dwork is bit-by-bit. Thus, to prove statement of the form "c is the ciphertext corresponding to m" it suffices to construct two zero-knowledge protocols: one to prove that a ciphertext decrypts to '0' and the other is to prove that a ciphertext decrypts to '1'. We construct two separate but in principle similar protocols for these tasks.

Ciphertexts of the AD cryptosystem are vectors in some public key dependent domain. The decryption algorithm decrypts every vector of the domain to '0' or '1', but not all vectors can be obtained by encrypting '0' or '1'. We say that a ciphertext is *legal* if it can be legally obtained by running encryption algorithm. The protocol for proving that a ciphertext c decrypts to 'b' (for $b \in \{0,1\}$ respectively) has the following properties of completeness and soundness: if c is a legal ciphertext of 'b', then the verifier always accepts; if the decryption of c is not 'b' (regardless whether c is a legal encryption of 'b' or not), then the verifier rejects with high probability. Thus, completeness holds only for c's which were obtained legally by applying the encryption algorithm, whereas soundness of the protocols holds for any input c from the prescribed domain.

We remark that the completeness of the protocols we present here requires some technical condition to hold for the public-key and the input ciphertext on which it is applied. Luckily, theorems proved in [23] show that with good probability, random public-keys produced by the AD key generation algorithm and random ciphertexts produced by the AD encryption algorithm obey these technical conditions. Moreover, it is easy to check if these conditions hold for a given public-key at key generation time, and for a given ciphertext at encryption time (using the randomness used by the algorithm to generate the ciphertext). Thus, we modify the AD key generation algorithm and encryption algorithm to ensure that all legally generated public-keys and ciphertext obey the desired conditions. We emphasize that the soundness of our protocols hold for all ciphertexts and public keys, regardless of whether they obey the said conditions.

The idea behind the protocol for proving that a ciphertext decrypts to '0' is as follows. We show a transformation of AD public-keys and ciphertexts to instances of the GapCVP_γ problem, such that (1) a legal AD public key and legal AD ciphertext which decrypts to '0', transforms to a YES instance of the GapCVP_γ; and (2) any AD public key and any ciphertext which decrypts to '1' transforms to a NO instance of the GapCVP_γ. On common input, a public key and a ciphertext, the prover and verifier transform it to the appropriate instance of GapCVP_γ and run the Micciancio and Vadhan [21] zero-knowledge protocol for proving that the constructed instance is a YES instance. The value of $\gamma = \Omega(\sqrt{\frac{n}{\log(n)}})$ where n is polynomially related to the value of the security parameter. The same approach is used to design the protocol proving that a ciphertext decrypts to '1'.

The second result of this paper is the design of a verifiable encryption scheme on inputs PK and ciphertext c for the *encrypted equivalence* relation $R_1 = \{(m, c')|c'$ is a legal AD encryption with public key PK of $m\}$. Again, as the AD cryptosystem is bit-by-bit, it will suffice to construct a statistical zero-knowledge protocol to prove that given two ciphertexts c and c', encryped with public key PK, decrypt to the same bit. The prover's auxilary inputs are the random bits used by the encryption algorithm to generate c and c'.

We take advantage of the observation that if c and c' are legal AD ciphertexts of the same bit under the same AD public-key PK, then with high probability $\overline{c} = (c + c') \bmod P(w_1, \ldots, w_n))$ decrypts to '0' (where $P(w_1, \ldots, w_n)$ is the parallelepiped spanned by the w_i's specified in the public key PK, see section 2.2). Thus, the prover need only prove is that $\overline{c}$ decrypts to '0' , using the statistical zero-knowledge protocol above for proving that AD ciphertext decrypts to '0'. If c is a legal ciphertext which decrypts to the same bit as c' the prover will succeed, whereas for any c which does not decrypt to the same bit as c' the prover will fail with high probability. Due to lack of space in this extended abstract further treatment of this result is omitted.

ZK Proofs of Plaintext Knowledge for AD Cryptosystem. We provide a **direct** (without using general results about NP in Zero-knowledge) zero-knowledge interactive proof of knowledge of the plaintext(PPK) for the AD cryptosystem.

As AD cryptosystem is a bit-by-bit encryption scheme, it suffices to describe how to prove on input public key PK, and ciphertext c of a single-bit plaintext b that the prover 'knows' b.

We prove that if c and c' are legal encryptions of b and b' respectively under AD public key PK, then with high probability $c + c' \mod P(w_1 \dots w_n)$ decrypts to $b \oplus b'$. The proof of plaintext knowledge for the AD cryptosystem follows naturally. On input (PK, c) where c is an encryption of b, the prover sends the verifier a random encryption c' of a random bit b'. The verifier then asks the prover to either prove that it knows the decryption of c' or to prove that it knows a decryption of $c + c' \mod P(w_1 \dots w_n)$. The former can be done simply by revealing the randomness used to encrypt c' and the latter can be done by proving in statistical zero-knowledge that $c + c'$ decrypts to $b \oplus b'$ using the statistical zero knowledge protocols designed in the first part of this work.

We prove that the resulting protocol is computational zero-knowledge under the same worst case intractability ISVP assumption of the AD cryptosystem.

Assumption ISVP: (Infeasibility of Shortest Vector Problem): There is no polynomial time algorithm, which given an arbitrary basis for an n-dimensional lattice which has a "unique $poly(n)$-shortest" vector, finds the shortest non-zero vector in the lattice. By "unique $poly(n)$-shortest" vector we mean that any vector in the lattice of length at most "poly(n)" times bigger than the shortest vector, is parallel to the shortest vector.

Combining the zero-knowledge PPK protocol with the AD cryptosystem, where the sender/encryptor (along with sending the ciphertext) interactively proves to the receiver that he knows the plaintext, yields automatically an interactive encryption scheme which is CCA1 secure based on ISVP. Previousy, Hall, Goldberg, Schneier [19] show how to completely recover the secret key of AD cryptosystem under a CCA1 attack.[3]

We believe that addressing the smaller problem of zero-knowledge PPK for AD cryptosystem as we have done here, is a promising first step in the pursuit of an CCA2 secure lattice based public-key encryption scheme, possibly first in an interactive setting by extending our protocol to be non-malleable.

2 Preliminaries

2.1 Notations

We let $x \in_R S$ denote choosing x at random with uniform probability in set S.

Given a parallelepiped $P = P(w_1, \dots, w_n)$ and a vector v, we *reduce* v *modulo* P by obtaining a vector $v' \in P$ so that $v' = v + \sum_i c_i w_i$, where the c_i are all integers. We denote it by $v' = v \bmod P$.

All distances in this paper, are the Euclidean distances in $\mathbb{R}^n$. Let $dist(v_1, v_2)$ denote the distance between vectors v_1 and v_2 in $\mathbb{R}^n$, and $\mathrm{dist}(v, S)$ denote the distance between vector v and a set S in $\mathbb{R}^n$.

[3] Their work explicitly addresses the [15] variant with eliminated decryption.

Let $v_1, \ldots, v_m$ be linearly independent vectors in $\mathbb{R}^n$. An *m-dimensional lattice* with the basis $\{v_1, \ldots, v_m\}$ is the set of all integer linear combinations of v_i's, $\{\sum_{i=1}^m a_i v_i \ : \ a_i \in \mathbb{Z}\}$.

For linearly independent vectors $w_1, \ldots, w_n$ in $\mathbb{R}^n$ the *parallelepiped* spanned by w_i's is the set

$$P(w_1, \ldots, w_n) = \left\{ \sum_{i=1}^n a_i w_i \ : \ a_i \in [0,1) \right\}.$$

The *width* of the parallelepiped $P(w_1, \ldots, w_n)$ is the maximum over i of distances between w_i and the subspace spanned by other w_i's.

For every $v \in \mathbb{R}^n$ there is only one $v' \in P(w_1, \ldots, w_n)$ such that $v - v' = \sum_{i=1}^n a_i w_i$ for some integers $a_1, \ldots, a_n$. We denote this by $v' = v \mod P(w_1, \ldots, w_n)$. Note, that we can consider n to be dimension of the lattice L. We can always consider a lattice to be enclosed in a subspace spanned by it's basis vectors.

For interactive protocols involving two parties A (the prover) and B (the verifier), we let the notation $(A(a), B(b))(x)$ be the random variable denoting whether B accepts or rejects common input x following an execution of the protocol where B has private private input b and A has private input a.

2.2 The Ajtai-Dwork Cryptosystem with Eliminated Decryption Errors

Let the security parameter be denoted by n.

In order to simplify the construction we present the scheme in terms of real numbers, but we always mean numbers with some fixed finite precision. We need to define several parameters which will be used throughout the paper. For a security parameter n let $m = n^3$, $\rho_n = 2^{n \log n}$. We denote by B_n the n-dimensional cube of side-length ρ_n. We also denote by S_n the n-dimensional ball of radius n^{-8}.

The errorless Ajati-Dwork cryptosystem [15] consists of three algorithms $(\mathcal{K}, \mathcal{E}, \mathcal{D})$, where $\mathcal{K}$ is a key generation algorithm, $\mathcal{E}$ is an encryption algorithm, and $\mathcal{D}$ is a decryption.

The encryption algorithm encrypts strings in a bit-by-bit fashion and thus in this paper we shall assume henceforth that all messages are single bits.

Key Generating algorithm $\mathcal{K}$ on input 1^n:

The private key SK = vector u chosen at random from the n-dimensional unit ball.

The public key $PK = \{w_1, \ldots, w_n, v_1, \ldots, v_m, k\}$, where $v_1, \ldots, v_m, w_1, \ldots, w_n$ are vectors in $\mathbb{R}^n$ generated as follows.

v's: For $i = 1 \ldots n$ (1) Pick vector a_i at random from the set $\{x \in B_n : \langle x, u \rangle \in \mathbb{Z}\}$; (2) For $j = 1, \ldots, n$ select δ_j at random in S_n; (3) Output $v_i = a + \sum_{j=1}^n \delta_j$.

w's: The vectors $w_1, \ldots, w_n$ are obtained according to the same procedure as vectors $v_1, \ldots, v_m$, subject to the additional constraint that the width of the parallelepiped $P(w_1, \ldots, w_n)$ is at least $n^{-2}\rho_n$. Remark: It is shown in [1] that the width of $P(w_1, \ldots, w_n)$ will be large enough with probability at least $1 - n^{-1/2}$.

k: Choose k at random from the set of $\{i : \langle a_i, u\rangle \text{ is an odd integer}\}$. We note that such an index exists with probability $1 - 2^{-\Omega(m)}$.

We let $(SK, PK) \in \mathcal{K}(1^n)$ denote picking a pair of keys according to generating algorithm $\mathcal{K}$ on input 1^n, and call such pair an *instance* of AD cryptosystem. In various definitions and theorems in this paper, given an instance (SK, PK) of the AD cryptosystem, we often refer directly to components of PK and SK as u,$v_1, \ldots, v_n$ etc.

At times our algorithms may take as input keys $K = \{w_1, \ldots, w_n, v_1, \ldots, v_m, k\}$ which may not have been generated by $\mathcal{K}$, in which case we refer to them as AD public-key's.

Encryption algorithm $\mathcal{E}$ on input public key PK and message bit b:

Choose $r = r_1, \ldots, r_m$, $r_i \in_R \{0, 1\}$.

If $b =$ '0', set ciphertext $c = \sum_{i=1}^{m} r_i v_i \mod P(w_1, \ldots, w_n)$.

If $b =$ '1', set ciphertext $c = (\frac{v_k}{2} + \sum_{i=1}^{m} r_i v_i) \mod P(w_1, \ldots, w_n)$.

Denote ciphertext c obtained by encrypting b under public key PK using randomness r, as $c = \mathcal{E}_{pk}(b; r)$.

Decryption algorithm $\mathcal{D}$ on input ciphertext c and secret key u:

If $\operatorname{dist}(\langle c, u\rangle, \mathbb{Z}) < \frac{1}{4}$, output '0', otherwise output '1'.

We let $\mathcal{D}_{SK}(c) = b$, denote the event that c decrypts to b, under secret key SK.

Note that the cryptosystem $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ is errorless. Namely, a legal encryption of '0' will always be decrypted as '0' and analogously an encryption of '1' is always decrypted as '1'.

2.3 Generating Good Public-keys and Ciphertexts

We note that completeness of the protocols we design in this paper, will only hold for public-keys and ciphertexts which obey certain 'good' technical conditions defined below.

By theorems proved by Nguyen and Stern in [23] (for the purposes of cryptanalysis of AD cryptosystem), it follows that such good public-keys and ciphertexts will come up with high probability in the natural course of running the generating algorithm $\mathcal{K}$ and encryption algorithm $\mathcal{E}$. Moreover, the parties who run $\mathcal{K}$ and $\mathcal{E}$ can check that the outputs are good, and if not repeat the process till a good output is computed.

We will thus modify the definition of algorithms $\mathcal{K}$ (for key generation) and $\mathcal{E}$ (for encryption) to to ensure they always output public-keys and ciphertexts which are good.

Formally,

Definition 21 *Let $\varepsilon \in (0,1)$. We say that a public key $PK = \{w_1, \dots, w_n, v_1, \dots, v_m, k\}$ where $v_1, \dots, v_m, w_1, \dots, w_n$ are vectors in $\mathbb{R}^n$ of AD is ε-good if*

$$E\left[\sum_{j=1}^{n}\left\langle \sum_{i=1}^{m}(b_i v_i), w_j^{\perp}\right\rangle^2\right] \le \frac{n^4 \rho_n^2}{2\varepsilon}, \tag{2.1}$$

where $w_j^{\perp}$ is a unit vector orthogonal to the hyperplane spanned by other w_j's. Expectation is taken over independent uniform choices of $b_1, \dots, b_m$ from $\{0,1\}$.

Claim 22 *[23] For sufficiently large n, for any $\varepsilon \in (0,1)$, a public key PK of AD picked at random according to the key generating protocol of section (2.2) is ε-good with probability at least $1-\varepsilon$.*

Definition 23 *Let $\varepsilon, \varepsilon_1 \in (0,1)$, and PK be an ε-good public key of AD. We say that a ciphertext c of '0' is $(\varepsilon, \varepsilon_1)-$good if for a_i, b_i's such that $c = \sum_{i=1}^{m} b_i v_i + \sum_{i=1}^{n} a_i w_i$*

$$\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}c \\ 0 \end{pmatrix}, B_{PK}(a_1, \dots, a_n, b_1, \dots, b_m)^t\right) \le \sqrt{1 + \frac{1}{2\varepsilon\varepsilon_1}} n^4 \tag{2.2}$$

Claim 24 *[23] For sufficiently large n, for any ε, $\varepsilon_1 \in (0,1)$ and an ε-good public key PK of AD the following holds: a random ciphertext c of '0' is $(\varepsilon, \varepsilon_1)$-good with probability at least $1-\varepsilon_1$. Probability is taken over random bits used by the encryption algorithm $\mathcal{E}$ to encrypt c.*

Definition 25 *Let $\varepsilon, \varepsilon_1 \in (0,1)$ and PK be an ε-good public key of AD. We say that a ciphertext c of '1' is $(\varepsilon, \varepsilon_1)-$good if and only $(c - \frac{v_k}{2}) \mod P(w_1, \dots, w_n)$ is a $(\varepsilon, \varepsilon_1)$-good ciphertext of '0'.*

Since, a random ciphertext c of '1' $(c - \frac{v_k}{2}) \mod P(w_1, \dots, w_n)$ is distributed as a random ciphertext of '0', we automatically get an analogous claim for random ciphertexts of '1'.

Claim 26 *For sufficiently large n, for any $\varepsilon, \varepsilon_1 \in (0,1)$ and for an ε-good public key PK of AD the following holds: a random ciphertext c of '1' is $(\varepsilon, \varepsilon_1)$-good with probability at least $1-\varepsilon_1$. Probability is taken over random bits used by the encryption algorithm to encrypt c.*

2.4 Modified AD Key Generation and Encryption Algorithms

We modify $\mathcal{K}$ and $\mathcal{E}$ to enforce the output of $\mathcal{K}$ to be ε-good and the output of $\mathcal{E}$ to be $(\varepsilon, \varepsilon_1)$-good.

For the protocols of section 3 we need ε, $\varepsilon_1 \in (0,1)$ to satisfy

$$\sqrt{1+\frac{1}{2\varepsilon\varepsilon_1}} \leq \left(\frac{1}{4}-\frac{2}{n^2}\right)\frac{n\sqrt{\log(n+n^3)}}{3\sqrt{2}}, \tag{2.3}$$

For the protocol of section 4 we need $\varepsilon, \varepsilon_1 \in (0,1)$ to satisfy

$$\sqrt{1+\frac{1}{2\varepsilon\varepsilon_1}} \leq \left(\frac{1}{4}-\frac{2}{n^2}\right)\frac{n\sqrt{\log(n+n^3)}}{12\sqrt{2}}. \tag{2.4}$$

Modified Key Generating algorithm $\mathcal{K}'$ on input 1^n:

Repeat
Let $(SK, PK) \in_R \mathcal{K}(1^n)$
Until $E\left[\sum_{j=1}^{n}\left\langle\sum_{i=1}^{m}(b_i v_i), w_j^{\perp}\right\rangle^2\right] \leq \frac{n^4\rho_n^2}{2\varepsilon}$ (where $PK = \{w_1, \ldots, w_n, v_1, \ldots, v_m, k\}$)
Output (SK, PK)

We let $(SK, PK) \in \mathcal{K}'(1^n)$ denote generating instance (SK, PK) according to key generation algorithm $\mathcal{K}'(1^n)$.

Modified Encryption algorithm $\mathcal{E}'$ on input public key PK and message bit b:

Repeat
Pick $r = r_1, \ldots, r_m$, $r_i \in_R \{0, 1\}$.
Let $c = \sum_{i=1}^{m} r_i v_i \mod P(w_1, \ldots, w_n)$.
Compute a_i's such that $c = \sum_{i=1}^{m} r_i v_i + \sum_{i=1}^{n} a_i w_i$.
Until dist $\left(\begin{pmatrix} n^6\sqrt{n}c \\ 0 \end{pmatrix}, B_{PK}(a_1, \ldots, a_n, b_1, \ldots, b_m)^t\right) \leq \sqrt{1+\frac{1}{2\varepsilon\varepsilon_1}}n^4$
Output $c + b\frac{v_k}{2} \mod P(w_1, \ldots, w_n)$.

We let $c \in \mathcal{E}'_{PK}(b)$ denote generating c by running algorithm $\mathcal{E}'$ on inputs PK and b, let $c \in \mathcal{E}'_{PK}(\cdot)$ denote c being in the domain of $\mathcal{E}'_{PK}$, and let $c = \mathcal{E}'_{PK}(b, r)$ denote generating c by running algorithm $\mathcal{E}'_{PK}$ on input b using randomness r.

2.5 Zero-knowledge Proof System for Approximate Closest Vector Problem

The protocols presented in this paper, exploit heavily the recent zero-knowledge protocol with for promise closest vector problem presented by Micciancio and Vadhan in [21].

Definition 27 *For $\gamma > 1$ instances of the promise closest vector problem GapCVP$_\gamma$ are tuples (L, t, x) where L is a lattice in $\mathbb{R}^n$ specified by its basis, $t > 0$, and vector x in $\mathbb{R}^n$.*

- *(L, t, x) is a YES instance of the GapCVP$_\gamma$ if* $\operatorname{dist}(L, x) \le t$
- *(L, t, x) is a NO instance of the GapCVP$_\gamma$ if* $\operatorname{dist}(L, x) > \gamma t$

The promise is that an instance of the GapCVP$_\gamma$ is restricted to be YES or NO instance, any other tuples are not instances of the GapCVP$_\gamma$.

In the protocol described by Micciancio and Vadhan [21] the prover proves to the verifier in zero-knowledge that a given instance of the GapCVP$_\gamma$ is a YES instance.

The protocol is statistical zero-knowledge for $\gamma = \Omega(\sqrt{\frac{n}{\log(n)}})$, where n is the dimension of the vector space containing the lattice L. Moreover, for such a γ the prover runs in polynomial time.

3 Verifiable Encryption for AD Cryptosystem

The ultimate goal of this section is to present two zero-knowledge protocols which form verifiable encryption schema for the equivalence relation. The first protocol is for proving that a ciphertext of AD decrypts to '0', and the second is for proving that a ciphertext of AD decrypts to '1'. In both protocols a common input to the prover and the verifier is a pair (PK, c) – public key of AD and a ciphertext. In addition, the prover has access to an auxiliary input consisting of random bits used to encrypt the ciphertext.

We will show a mapping from a pair (PK, c) to an instance (L, t, x) of GapCVP$_\gamma$ such that for good public keys and ciphertexts of bit '0' the pair maps to a YES instance of GapCVP$_\gamma$, whereas for any ciphertext which descrypts to '1' the pair maps to a NO instance of GapCVP$_\gamma$. Then, to prove that c decrypts to '0', simply run the ZK protocol of [21] to prove that (L, t, x) is a YES instance of GapCVP$_\gamma$. The case of ciphertext which decrypts to '1', is similarly handled.

Throughout this section n denotes the security parameter, $m = n^3$, and $\gamma = \sqrt{\frac{n+m}{\log(n+m)}}$.

3.1 Mapping AD ciphertexts to GapCVP instances

We define a mapping from pairs (PK, c) consisting of a public key and a ciphertext of AD to instances of GapCVP$_\gamma$.

Definition 31 *Let $PK = \{w_1, \ldots, w_n, v_1, \ldots, w_m, k\}$ be a public key of AD. Let c be a vector from $P((w_1, \ldots, w_n)$. Define mapping $\mathcal{F}(PK, c) = (L_{PK}, t, x_c)$ where*

$$x_c = \begin{pmatrix} n^6 \sqrt{n} c \\ 0 \end{pmatrix} \in \mathbb{R}^{n+2m}, \quad t = n^4 \sqrt{1 + \frac{1}{2\varepsilon\varepsilon_1}} \tag{3.1}$$

And L_{PK} is an $(n+m)$-dimensional lattice in $\mathbb{R}^{2n+m}$ spanned by the columns of the following matrix B_{PK},

$$B_{PK} = \begin{pmatrix} n^6\sqrt{n}w_1 & \dots & n^6\sqrt{n}w_n & n^6\sqrt{n}v_1 & \dots & n^6\sqrt{n}v_m \\ 1 & 0 & & \dots & & 0 \\ 0 & \ddots & & & & \\ \vdots & & 1 & \ddots & & \vdots \\ & & \ddots & n^2\sqrt{n} & & \\ & & & & \ddots & 0 \\ 0 & & & \dots & 0 & n^2\sqrt{n} \end{pmatrix} \tag{3.2}$$

3.2 Connection between AD ciphertexts of '0' and the GapCVP$_\gamma$ problem

We next state the theorem which forms a theoretical basis for the protocol for proving that a ciphertext decrypts to '0'. The theorem states that good public keys and ciphertexts of '0' map under $\mathcal{F}$ to a YES instance of GapCVP$_\gamma$, whereas any ciphertext which decrypts to '1', will map under $\mathcal{F}$ to a NO instance of GapCVP$_\gamma$.

Theorem 32 *For sufficiently large n,*

1. *For $(SK, PK) \in \mathcal{K}'(1^n)$ and $c \in \mathcal{E}'_{PK}(0)$, $\mathcal{F}(PK, c)$ is a YES instance of GapCVP$_\gamma$.*
2. *for any instance (SK, PK) of AD and $c \in P(w_1, \dots, w_n)$ such that $\mathcal{D}_{SK}(c) =$ '1', $\mathcal{F}(PK, c)$ is a NO instance of GapCVP$_\gamma$.*

Proof. (1) The first statement directly follows from the definition of an $(\varepsilon, \varepsilon_1)$-good ciphertext of '0'.

(2) Let $c \in P(w_1, \dots, w_n)$ be any vector which decrypts to '1'. Let $T = t\gamma$. From (2.3) it follows that

$$\frac{3T}{n^6\sqrt{n}} = 3\frac{\sqrt{1+\frac{1}{2\varepsilon\varepsilon_1}}\sqrt{n+n^3}}{n^2\sqrt{n}\sqrt{\log(n+n^3)}} < \frac{3\sqrt{1+\frac{1}{2\varepsilon\varepsilon_1}}\sqrt{2}}{n\sqrt{\log(n+n^3)}} \le \frac{1}{4} - \frac{2}{n^2} < \frac{1}{4} < \operatorname{dist}\left(\langle c, u\rangle, \mathbb{Z}\right).$$

By theorem 33 (proved below) $\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}c \\ 0 \end{pmatrix}, L_{PK}\right) \le T$ can not hold. Thus $\left(L_{PK}, t, \begin{pmatrix} n^6\sqrt{n}c \\ 0 \end{pmatrix}\right)$ is a NO instance of the GapCVP$_\gamma$.

Theorem 33 *Let $T > 0$, PK be a public key of AD, and $c \in P(w_1, \dots, w_n)$. For sufficiently large n,*

$$\text{If} \quad \operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}c \\ 0 \end{pmatrix}, L_{PK}\right) \le T \quad \text{then} \quad \operatorname{dist}\left(\langle u, c\rangle, \mathbb{Z}\right) \le \frac{3T}{n^6\sqrt{n}} \tag{3.3}$$

Proof. Let $c \in P(w_1, \ldots, w_n)$ be such that $\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}c \\ 0 \end{pmatrix}, L_{PK}\right) \leq T$, hence there are integers $a_1, \ldots, a_n, b_1, \ldots, b_m$ such that $\left\| \begin{pmatrix} n^6\sqrt{n}c \\ 0 \end{pmatrix} - B_{PK}(a_1, \ldots, a_n, b_1, \ldots, b_m)^t \right\|^2 \leq T^2$.

Observing the construction of the matrix B_{PK} (3.2) we get that for the vector $e = n^6\sqrt{n}c - n^6\sqrt{n}\left(\sum_{i=1}^n a_i w_i + \sum_{i=1}^m b_i v_i\right)$

$$\sum_{i=1}^n a_i^2 + \sum_{i=1}^m n^5 b_i^2 + \|e\|^2 \leq T^2. \tag{3.4}$$

$\|e\| \leq T$, thus $|\langle u, e\rangle| \leq T$. It follows that $\operatorname{dist}(\langle u, e\rangle, \mathbb{Z}) \leq T$.

Note that $c = \sum_{i=1}^n a_i w_i + \sum_{i=1}^m b_i v_i + \frac{e}{n^6\sqrt{n}}$, hence

$$\operatorname{dist}(\langle u, c\rangle, \mathbb{Z}) \leq \sum_{i=1}^n |a_i| \operatorname{dist}(\langle u, w_i\rangle, \mathbb{Z}) + \sum_{i=1}^m |b_i| \operatorname{dist}(\langle u, v_i\rangle, \mathbb{Z}) + \frac{T}{n^6\sqrt{n}}. \tag{3.5}$$

Let us upper bound the first term of (3.5). According to the construction of AD for all $i = 1, \ldots, n$ $\operatorname{dist}(\langle u, w_i\rangle, \mathbb{Z}) \leq \frac{1}{n^7}$. From (3.4) it follows that $\sum_{i=1}^n a_i^2 \leq T^2$. Thus, by the Cauchy-Schwartz inequality we have that $\sum_{i=1}^n |a_i| \operatorname{dist}(\langle u, w_i\rangle, \mathbb{Z}) \leq \sqrt{\sum_{i=1}^n a_i^2} \times \sqrt{\sum_{i=1}^n \operatorname{dist}(\langle u, w_i\rangle, \mathbb{Z})^2} \leq T\sqrt{n \times n^{-14}} = \frac{T}{n^6\sqrt{n}}$.
Let us now upper bound the second term of (3.5). Similarly, for all $i = 1, \ldots, m$ $\operatorname{dist}(\langle u, v_i\rangle, \mathbb{Z}) \leq \frac{1}{n^7}$. From (3.4) we have that $\sum_{i=1}^m b_i^2 \leq \frac{T^2}{n^5}$. Applying the Cauchy-Schwartz inequality we get that $\sum_{i=1}^m |b_i| \operatorname{dist}(\langle u, v_i\rangle, \mathbb{Z}) \leq \sqrt{\sum_{i=1}^m b_i^2} \times \sqrt{\sum_{i=1}^m \operatorname{dist}(\langle u, v_i\rangle, \mathbb{Z})^2} \leq \frac{T}{n^2\sqrt{n}}\sqrt{n^3 \times n^{-14}} = \frac{T}{n^8} \leq \frac{T}{n^6\sqrt{n}}$.

Combining all together we obtain that $\operatorname{dist}(\langle c, u\rangle, \mathbb{Z}) \leq \frac{3T}{n^6\sqrt{n}}$

We are ready to present the protocol which form verifiable encryption schema for the equivalence relation when the claimed plaintext is '0'.

Protocol$_0$: proving that a ciphertext decrypts to '0'.
Let P_0 and V_0 denote the prover and the verifier. Let the common input to P_0 and V_0 be a pair (PK, c) where $PK = \{w_1, \ldots, w_n, v_1, \ldots, v_m, k\}$ is a public key of AD and c is a vector from $P(w_1, \ldots, w_n)$. The prover's auxiliary input is $b_1, \ldots, b_m \in \{0, 1\}$ such that $c = \sum_{i=1}^m b_i v_i \mod P(w_1, \ldots, w_n)$.

- **Prover** P_0 Calculates integers $a_1, \ldots, a_n$ such that $c = \sum_{i=1}^m b_i v_i + \sum_{i=1}^n a_i w_i$. Invokes the [21] prover (with auxiliary input $B_{PK}(a_1, \ldots, a_n, b_1, \ldots, b_m)^t$) to prove that input $\mathcal{F}(PK, c)$ is a YES instance of GapCVP$_\gamma$.
- **Verifier** V_0 Invoke the [21] verifier to verify that input $\mathcal{F}(PK, c)$ is a YES instance of GapCVP$_\gamma$.

Claim 34 *Protocol (P_0, V_0) satisfy the following completeness, soundness, and zero-knowledge properties:*

- **Completeness:** *If* $(SK, PK) \in \mathcal{K}'(1^n)$ *and* $c \in \mathcal{E}'_{PK}(0)$*, then* $Prob((P_0, V_0)(PK, c) = accepts) = 1$.
- **Soundness** *If* (PK, SK) *is an instance of* AD *and* $c \in P(w_1, \ldots, w_n)$ *such that* $D_{SK}(c) = `1'$*, then for all prover* P_0', $Prob((P_0', V_0)(PK, c) = rejects) > \frac{1}{2}$.
- **Zero-Knowledge :** *statistical zero-knowledge.*

Proof. The soundness condition relies on the part (2) of the theorem 32 and the soundness condition of the proof system from [21]. The completeness condition follows from the part (1) of the theorem 32 and completeness condition of the proof system from [21]. The lattice L_{PK} is an $(n+m)$-dimensional lattice, hence, the approximation factor $\gamma = \sqrt{\frac{n+m}{\log(n+m)}}$ is as required for statistical zero-knowledge property of the proof system from [21].

3.3 Connection between AD '1' ciphertexts and the GapCVP$_\gamma$ problem

In this subsection we construct a zero-knowledge protocol for proving that a ciphertext of AD decrypts to '1'. We use the nice observation that for a random ciphertext of AD of '1' the distribution of vector $(c - \frac{v_k}{2}) \mod P(w_1, \ldots, w_n)$ is the same as distribution of a random ciphertext of '0'. Thus, to prove that a ciphertext c decrypts to '1', we will prove that $(c - \frac{v_k}{2}) \mod P(w_1, \ldots, w_n)$ decrypts to '0', by running *protocol*$_0$ on inputs PK and $(c - \frac{v_k}{2}) \mod P(w_1, \ldots, w_n)$.

To prove soundness however, we must be careful, as we notice that for a c which decrypts to '0', $(c - \frac{v_k}{2}) \mod P(w_1, \ldots, w_n)$ is *not* distributed as a random ciphertext of '1', however as shown by the the following theorem it is quite close to it.

Theorem 35 *For any* (SK, PK) *instance of* AD*, for any vector* $c \in P(w_1, \ldots, w_n)$ *such that* $\mathcal{D}_{SK}(c) = `0'$*, for sufficiently large* n*, the* $\text{dist}(\langle y, u\rangle, \mathbb{Z}) > \frac{1}{4} - \frac{2}{n^2}$ *for* $y = (c - \frac{v_k}{2}) \mod P(w_1, \ldots, w_n)$

Proof. Let $c \in P(w_1, \ldots, w_n)$ decrypts to '0'.

There is a representation $(c - \frac{v_k}{2}) \mod P(w_1, \ldots, w_n) = c - \frac{v_k}{2} + \sum_{i=1}^n a_i w_i$.

$$\text{dist}\left(\left\langle c - \frac{v_k}{2} + \sum_{i=1}^n a_i w_i, u\right\rangle, \mathbb{Z}\right) \geq \text{dist}\left(\left\langle \frac{v_k}{2}, u\right\rangle, \mathbb{Z}\right) - \text{dist}\left(\langle c, u\rangle, \mathbb{Z}\right) - \text{dist}\left(\left\langle \sum_{i=1}^n a_i w_i, u\right\rangle, \mathbb{Z}\right) \tag{3.6}$$

Let us bound the terms of (3.6).

$$\text{dist}\left(\left\langle \sum_{i=1}^n a_i w_i, u\right\rangle, \mathbb{Z}\right) \leq \sum_{i=1}^n |a_i| \, \text{dist}\left(\langle w_i, u\rangle, \mathbb{Z}\right) \leq \frac{1}{n^7} \sum_{i=1}^n |a_i|. \tag{3.7}$$

Note, that $a_i = \lfloor \theta_i \rceil$ for θ_i defined as $c - \frac{v_k}{2} = \sum_{i=1}^{n} \theta_i w_i$. Since the width of the parallelepiped $P(w_1, \ldots, w_n)$ is greater than $\frac{\rho_n}{n^2}$, (3.7) can be bounded by

$$\frac{1}{n^7}\sum_{i=1}^{n}|a_i| \le \frac{1}{n^7}\sum_{i=1}^{n}|\theta_i| \le \frac{1}{n^5\rho_n}\sum_{i=1}^{n}\left|\left\langle c - \frac{v_k}{2}, w_i^{\perp}\right\rangle\right| \le \frac{1}{n^4\rho_n}\left\|c - \frac{v_k}{2}\right\| \le \frac{1}{n^2}.$$

$\mathrm{dist}\left(\langle c, u\rangle, \mathbb{Z}\right) \le \frac{1}{4}$ and $\mathrm{dist}\left(\left\langle \frac{v_k}{2}, u\right\rangle, \mathbb{Z}\right) \ge \frac{1}{2} - \frac{1}{n^7}$. Collecting all together we get that (3.6) is greater than $\frac{1}{2} - \frac{1}{n^7} - \frac{1}{4} - \frac{1}{n^2}$ which is greater than $\frac{1}{4} - \frac{2}{n^2}$ for sufficiently large n.

The following theorem forms the theoretical basis for the protocol for proving that a ciphertext decrypts to '1'

Theorem 36 *For sufficiently large n,*

- *If $(SK, PK) \in \mathcal{K}'(1^n)$ and $c \in \mathcal{E}'_{PK}(1)$, then $\mathcal{F}(PK, y)$ is a YES instance of the GapCVP$_\gamma$ for $y = (c - \frac{v_k}{2}) \bmod P(w_1, \ldots, w_n)$.*
- *If (PK, SK) is an instance of AD cryptosystem and $c \in P(w_1, \ldots, w_n)$ such that $\mathcal{D}_{SK}(c) =$ '0', then $\mathcal{F}(PK, y)$ is a NO instance of the GapCVP$_\gamma$ for $y = (c - \frac{v_k}{2}) \bmod P(w_1, \ldots, w_n)$.*

(1) The statement directly follows from the definition of an $(\varepsilon, \varepsilon_1) -$ *good* ciphertext of '1'.

(2) Let $c \in P(w_1, \ldots, w_n)$ be any vector which decrypts to '0'. Define $y = (c - \frac{v_k}{2}) \bmod P(w_1, \ldots, w_n)$. From (2.3) it follows that

$$\frac{3t\gamma}{n^6\sqrt{n}} = 3\frac{\sqrt{1+\frac{1}{2\varepsilon\varepsilon_1}}\sqrt{n+n^3}}{n^2\sqrt{n}\sqrt{\log(n+n^3)}} < \frac{3\sqrt{1+\frac{1}{2\varepsilon\varepsilon_1}}\sqrt{2}}{n\sqrt{\log(n+n^3)}} \le \frac{1}{4} - \frac{2}{n^2} <$$

[By the theorem 35]$<$ $\mathrm{dist}\left(\langle y, u\rangle, \mathbb{Z}\right)$.

Thus, by the theorem 33 $\mathrm{dist}\left(\begin{pmatrix} n^6\sqrt{n}y \\ 0 \end{pmatrix}, L_{PK}\right) \le t\gamma$ can not hold, and $\left(L_{PK}, t, \begin{pmatrix} n^6\sqrt{n}y \\ 0 \end{pmatrix}\right)$ is a NO instance of the GapCVP$_\gamma$.

We are ready to present the protocol for proving that a ciphertext decrypts to '1'.

Protocol$_1$: proving that a ciphertext decrypts to '1'.
Let P_1 and V_1 denote the prover and the verifier. Let the common input to P_1 and V_1 be a pair (PK, c) where $PK = \{w_1, \ldots, w_n, v_1, \ldots, v_m, k\}$ is a public key of AD and c is a vector from $P(w_1, \ldots, w_n)$. Let P_1 auxiliary input be $b_1, \ldots, b_m \in \{0, 1\}$ such that $c = (\frac{v_k}{2} + \sum_{i=1}^{m} b_i v_i) \bmod P(w_1, \ldots, w_n)$.

– **Prover** P_1**:** Calculate $y = (c - \frac{v_k}{2}) \mod P(w_1, \ldots, w_n)$. Calculate integers $a_1, \ldots, a_n$ such that $y = \sum_{i=1}^{m} b_i v_i + \sum_{i=1}^{n} a_i w_i$. Invoke the [21] prover (with auxiliary input $B_{PK}(a_1, \ldots, a_n, b_1, \ldots, b_n)$) to prove that input $\mathcal{F}(PK, y)$ is a YES instance of GapCVP_γ.

– **Verifier** V_1**:** Calculate $y = (c - \frac{v_k}{2}) \mod P(w_1, \ldots, w_n)$. Invoke the [21] verifier to verify that $\mathcal{F}(PK, y)$ is a YES instance of GapCVP_γ .

It is evident that the soundness, completeness, and Zero-knowledge properties of $Protocol_1$ are similar to the soundness and Zero-Knowledge properties of $Protocol_0$.

4 Proof of AD Plaintext Knowledge

4.1 Definition of proofs of knowledge

We use the definition of a proof of knowledge from [18]

Definition 41 *Let $Q(\cdot)$ be a polynomial, x the common input for the prover P and verifier V, and r a uniformly selected random tape of prver P. Run the protocol between P and V, $Q(|x|)$ times, each time runniing prover P on the same random tape r and the verifier V on a newly selected uniformly chosen random tape. Let (P, V, x, Q) denote the sequence of the verifier's views obtaind from the above execution. We call the distribution over such sequences* a valid (P, V, x, Q) - distribution.

Definition 42 *Let $\eta \in \{0, 1\}$, an interactive protocol (P, V) with prover P and a verifier V is a* proof of knowledge system with knowledge error η for a relation R *if the following holds:*

Completeness: For every common input x for which there exists y such that $(x, y) \in R$ the verifier V always accepts interacting with the prover P.
Validity with error η: There exists a polynomial time interacting oracle Turing machine `Sample` *and a polynomial time algorithm* `Extract`*, a constant $c > 0$ and a polynomial $Q(\cdot)$ such that for every $x \in \{0, 1\}^*$ such that $R(x) \neq \emptyset$ and for every prover P' the following holds:*

- $\mathtt{Sample}^{P'}(x)$ *outputs a valid (P', V, x, Q)- distribution of verifier's view.*
- $\mathtt{Extract}(\mathtt{Sample}^{P'}(x)) \in R(x) \cup \{\text{"fail"}\}$
- $Pr[\mathtt{Extract}(\mathtt{Sample}^{P'}(x)) \in R(x)] \geq (p - \eta)^c$*, where $p > \eta$ is a probability that V accepts while interacting with P' on common input x.*

We call the pair $(\mathtt{Sample}, \mathtt{Extract})$ *a* knowledge extractor.

4.2 The Plaintext Knowledge Relation for AD cryptosystem

Throughout the rest of section 4 we assume that n denotes the security parameter, and m, L_{PK}, and γ, are as defined in section 3 whereas $t = 4\sqrt{1 + \frac{1}{2\varepsilon\varepsilon_1}}n^4$. Define relation R_{AD} corresponding to knowing a plaintext of an AD ciphertext as follows.

Definition 43 *Let $PK = \{w_1, \ldots, w_n, v_1, \ldots, v_m, k\}$ be a public key of AD, c and c' vectors from $P(w_1, \ldots, w_n)$, b' and $b'' \in \{0,1\}$, $r' \in \{0,1\}^m$, and p be a point from L_{PK}. We say that input (PK, c) and witness (c', b', r', b'', p) are in R_{AD} if:*

- $c' = \mathcal{E}_{PK}(b'; r')$
- $\mathrm{dist}\left(\begin{pmatrix} n^6\sqrt{n}((c' + c - b''\frac{v_k}{2}) \bmod P(w_1, \ldots, w_n)) \\ 0 \end{pmatrix}, p\right) \leq \gamma t$ *(i.e. $(c{+}c') \bmod P(w_1, \ldots, w_n)$) decrypts to b'')*

Intuitively, proving knowledge of a witness for (PK, c), implies knowledge of plaintext of c under PK. This is formally captured by the following theorem.

Theorem 44 *Let (PK, SK) be an instance of the AD cryptosystem. If $((PK, c), w) \in R_{AD}$ for $w = (c', b', r', b'', p)$, then $b' \oplus b'' = D_{SK}(c)$.*

Proof. Let $PK = \{w_1, \ldots, w_n, v_1, \ldots, v_m, k\}$.

Consider the case when $b'' = 0$. In this case

$$\mathrm{dist}\left(\begin{pmatrix} n^6\sqrt{n}((c' + c) \bmod P(w_1, \ldots, w_n)) \\ 0 \end{pmatrix}, p\right) \leq \gamma t$$

,

By theorem 33, $\mathrm{dist}\left(\langle (c + c') \bmod P(w_1, \ldots, w_n), SK\rangle, \mathbb{Z}\right) \leq \frac{3T}{n^6\sqrt{n}} = 12\frac{\sqrt{1+\frac{1}{2\varepsilon\varepsilon_1}}\sqrt{n+n^3}}{n^2\sqrt{n}\sqrt{\log(n+n^3)}} \leq \frac{\sqrt{n+n^3}}{8n\sqrt{2}\sqrt{n}} \leq \frac{1}{8}$.

Suppose $b' = 0$. Since c' is a legal ciphertext, $\mathrm{dist}\left(\langle c', SK\rangle, \mathbb{Z}\right) \leq \frac{1}{n}$ which implies that $\mathrm{dist}\left(\langle c, SK\rangle, \mathbb{Z}\right) < \frac{1}{4}$ and $\mathcal{D}_{SK}(c) =$ '0'.

Suppose $b' = 1$. Since c' is a legal ciphertext, $\mathrm{dist}\left(\langle c', SK\rangle, \mathbb{Z}\right) \geq \frac{1}{2} - \frac{1}{n}$ which implies that $\mathrm{dist}\left(\langle c, SK\rangle, \mathbb{Z}\right) > \frac{1}{4}$ and $\mathcal{D}_{SK}(c) =$ '1'.

A similar case analysis follows when $b'' = 1$.

Note, that one can easily check whether a pair (PK, c) and a particular witness are in the relation R_{AD}. Since AD is semantically secure, for a public key PK of AD generated in random according to the key generating algorithm and a random ciphertext c of a uniformly chosen bit encrypted under the public key PK it is impossible to construct a witness for (PK, c) with non-negligible probability.

4.3 Protocol for proof of plaintext knowledge for AD

Let us first provide a sketch of the protocol. For public key $PK = \{w_1, \ldots, w_n, v_1, \ldots, v_m, k\}$ and ciphertext c, we distill the following nice homomorphic properties of AD:

- If c is an encryption of the bit b, then $c + \frac{v_k}{2} \mod P(w_1 \ldots w_n)$ is decrypted to $\bar{b}$
- If c, c' are encryptions of b, b' (respectively) then $c + c' \mod P(w_1 \ldots w_n)$ is decrypted to $b \oplus b'$.

Using these properties, it is simple to design a proof of knowledge of bit b encrypted by ciphertext c: the prover sends a random encryption c' of a random bit b', and the verifier asks the prover to show either that it knows the decryption of c' or that it knows a decryption of $c + c'$. The former can be done simply revealing the randomness used to encrypt c' and the latter can be done by proving in zero-knowledge that $c + c'$ decrypts to $b \oplus b'$. This is achieved by utilizing a variant of the protocols of section 3.2 to show that $(c+c')$ decrypts to zero (in case of $b \oplus b' = 0$) or that $(c+c') + \frac{v_k}{2}$ decrypts to zero (when $b \oplus b' =$ '1').

Protocol PPK

Let P_{PPK} and V_{PPK} denote the prover and the verifier respectively. The common input to P_{PPK} and V_{PPK} is (PK, c) where $PK = \{w_1, \ldots, w_n, v_1, \ldots, v_m, k\}$ is an AD public-key and c is a vector from $P(w_1, \ldots, w_n)$. The prover's auxiliary input is plaintext b and randomness r such that $c = \mathcal{E}'_{PK}(b; r)$.

- **Step (P1):** P_{PPK} selects $b' \in_R \{0,1\}$, computes $c' \in_R \mathcal{E}'_{PK}(b')$ and sends c' to V_{PPK}.
- **Step (V1):** V_{PPK} sends a random challenge bit $\delta \in_R \{0,1\}$ to P_{PPK}.
- **Step (P2):**
  - If $\delta = 0$, P_{PPK} sends pair (b', r') where $c' = \mathcal{E}'_{PK}(b'; r')$ to V_{PPK}.
  - If $\delta = 1$, P_{PPK} computes $b'' = b \oplus b'$; sends b'' to verifier; lets $\overline{c} = (c + c') \mod P(w_1, \ldots, w_n))$ and runs the prover of $Protocol'_{b''}$ on input $(PK, \overline{c})$
- **Step (V2):**
  - If $\delta = 0$, then (c', r') has been received in step (P2). V_{PPK} rejects if $c' \neq E(b'; r')$, else it accepts.
  - If $\delta = 1$, let b'' be bit received in step P2. V_{PPK} set $\overline{c} = (c + c') \mod P(w_1, \ldots, w_n))$; run the verifier of Protocol'$_{b''}$ on input $(PK, \overline{c})$.

The flow of message communication is presented in picture 1.

Protocol PPK (in steps P2,V2) makes calls to two zero-knowledge protocols $Protocol'_0$ and $Protocol'_1$ which enable the prover to prove that a given sum of two ciphertexts of AD decrypt to '0' (or '1' respectively). These protocols are identical in structure to the protocols of section 3.2 and 3.3, except for a slight difference in the YES instances of GapCVP$_\gamma$ constructed.

Define [4] mapping $\mathcal{G}(PK, c) = (L_{PK}, t, x_c)$ where $t = 4\sqrt{1+\frac{1}{2\varepsilon\varepsilon_1}}n^4$ and x_c, L_{PK} are as in section 3.1

Protocol$'_0$ on input $(PK, \overline{c})$ is the statistical ZK protocol of [21] proving that input $\mathcal{G}(PK, \overline{c})$ is a YES instance of GapCVP$_\gamma$.

Protocol$'_1$ on input $(PK, \overline{c})$ is the statistical ZK protocol of [21] proving that input $\mathcal{G}(PK, (\overline{c} - \frac{v_k}{2}) \mod P(w_1, \ldots, w_n))$ is a YES instance of GapCVP$_\gamma$.

The following properties of these protocols are needed for larger protocol PPK. Note the similarity with theorem 32 and 36.

Claim 45 *For sufficiently large n,*

1. *If $(SK, PK) \in \mathcal{K}'(1^n)$, $c = (c_1+c_2) \mod P(w_1, \ldots, w_n)$ such that $D_{SK}(c) =$ '$0'$ and $c_1, c_2 \in \mathcal{E}'_{PK}(\cdot)$, $\mathcal{G}(PK, c)$ is a YES instance of GapCVP$_\gamma$.*
2. *Let (SK, PK) be an instance of AD and $c \in P(w_1, \ldots, w_n)$. If $\operatorname{dist}(\langle c, SK\rangle, \mathbb{Z}) > \frac{1}{8}$, then $\mathcal{G}(PK, c)$ is a NO instance of GapCVP$_\gamma$.*

Proof. We defer the proof to the end of the section.

Claim 46 *For sufficiently large n, the following holds:*

1. *For any $(SK, PK) \in \mathcal{K}'(1^n)$, for any $c = (c_1 + c_2) \mod P(w_1, \ldots, w_n)$ such that $\mathcal{D}_{SK}(c) =$ '$1'$ and where $c_1, c_2 \in \mathcal{E}'_{PK}(\cdot)$, $\mathcal{G}(PK, y)$ is a YES instance of the GapCVP$_\gamma$ where $y = (c - \frac{v_k}{2}) \mod P(w_1, \ldots, w_n)$.*
2. *For any instance (SK, PK) of AD, and for any $c = P(w_1, \ldots, w_n)$ such that $\operatorname{dist}(\langle c, SK\rangle, \mathbb{Z}) < \frac{3}{8}$, $\mathcal{G}(PK, y)$ is a NO instance of GapCVP$_\gamma$ for $y = (c - \frac{v_k}{2}) \mod P(w_1, \ldots, w_n)$.*

The proof is similar to the proof of theorem 45 and is omitted.

We are now ready to prove that protocol PPK forms a proof of knowledge system with error $\frac{3}{4}$ for binary relation R_{AD} which is zero-knowledge.

Theorem 47 (Completeness and Soundness of PPK) *Interactive protocol (P_{PPK}, V_{PPK}) is a proof of knowledge system with knowledge error $\frac{3}{4}$ for R_{AD}.*

Proof. First lets argue completeness. Namely, if PK is an ε-good ciphertext and c is an $\varepsilon, \varepsilon_1$-good ciphertext under PK then the V_{PPK} always accepts interacting with the P_{PPK}.

The completeness property becomes evident, due to the simple fact about ciphertexts of AD: for two legal ciphertexts c_1 and c_2 of AD with plaintexts b_1 and b_2 the vector $(c_1 + c_2) \mod P(w_1, \ldots, w_n)$ decrypts to $b_1 \oplus b_2$.

Second, lets argue validity with knowledge error $\frac{3}{4}$. We will present a knowledge extractor consisting of two algorithms `Sample` and `Extract` which satisfy the conditions of the definition of a proof of knowledge.

Let $PK = \{w_1, \ldots, w_n, v_1, \ldots, v_m, k\}$ be a public key of AD and $c \in P(w_1, \ldots, w_n)$. Let P' be an arbitrary prover making the V_{PPK} accept with probability $\frac{3}{4} + \sigma$, for $\sigma > 0$ on common input (PK, c).

[4] The only difference between $\mathcal{G}$ and $\mathcal{F}$ of section 3 is in the value of t used

PPK-prover		PPK-verifier
	Common input: (PK, c) s.t. $PK = \{w_1, \ldots, w_n, v_1, \ldots, v_m\}$ - public key of AD $c \in P(w_1, \ldots, w_n)$	
Auxiliary input: $b, b' \in \{0,1\}$, $r, r' \in \{0,1\}^m$ s.t. $c = E_{PK}(b; r)$ and $c' = E_{PK}(b'; r')$		
	$\xrightarrow{\quad c' = E_{PK}(b'; r') \quad}$	
	$\xleftarrow{\quad \delta \in_R \{0,1\} \quad}$	
$\delta = 0:$	$\xrightarrow{\quad b', r' \quad}$	$c' \stackrel{?}{=} E_{PK}(b'; r')$
$\delta = 1:$	$\xrightarrow{\quad b'' = b \oplus b' \quad}$ Prove in ZK that $c + c' \mod P(w_1, \ldots, w_n)$ decrypts to b''	

The algorithm `Sample`*:* The algorithm `Sample` is an interactive Turing machine with oracle access to P'. The input of `Sample` is (PK, c). The algorithm outputs three strings distributed as verifier's views at the end of the protocol between P' and V_{PPK} run on common input (PK, c) (i.e `Sample` outputs a valid $(P', V_{PPK}, (PK, c), 3)$-distribution of verifier's view). `Sample` chooses a random string r which will serve as a random tape for P'. `Smaple` outputs three verifiers views V_1, V_2, V_3 independently according to the following procedure: Set the random tape of P' to r. Generate a random bit δ which will be used for verifier's challenge. If $\delta = 1$ the prover and the verifier should be involved in one of the subprotocols ($Protocol'_0$ or $Protocol'_1$). Each subprotocol is a three-move interactive proof system with one-bit verifier's challenge. Generate a random bit δ_1 for the second verifier's challenge. Simulate the protocol between P' an the V_{PPK} on common input (PK, c) interacting with P' as a verifier and sending challenge bits δ and δ_1 (if needed). Output the verifiers view which consists of common input (PK, c), simulated transcript of the protocol and random bits δ and δ_1 (if needed).

The algorithm `Extract`*:* Input of the algorithm `Extract` consists of three verifier's views V_1, V_2, V_3 generated by `Sample`. Let transcripts of the protocol involved in the views be denoted as T_1, T_2 and T_3. If one of the transcript is not accepting, `Extract` outputs "fail" and halts. Since the probability that

P' makes V_{PPK} accept is $\frac{3}{4}+\sigma$, `Extarct` continues with probability at least σ. The algorithm checks the following conditions:

Verifier's view V_1 involves $\delta = 0$.
Verifier's view V_2 involves $\delta = 1$ and $\delta_1 = 0$.
Verifier's view V_3 involves $\delta = 1$ and $\delta_1 = 1$.

If at least one of the conditions does not hold then `Extract` outputs "fail" and halts. If the algorithm continues, what happens with probability $\frac{1}{32}$, T_1, T_2 and T_3 has the following form:

$T_1 = (c', 0, b', r')$
$T_2 = (c', 1, b'', T_1')$
$T_3 = (c', 1, b'', T_2')$

Where T_1' and T_2' are transcripts of $Protocol_0'$ and $Protocol_1'$ respectively. Note, that the subprotocols are based on the proof system of Micciancio and Vadhan and actually are aimed to prove that

$$\left(L_{PK}, t, \begin{pmatrix} n^6\sqrt{n}((c'+c-b''\frac{v_k}{2}) \bmod P(w_1,\ldots,w_n)) \\ 0 \end{pmatrix}\right)$$

is not a NO instance of the GapCVP$_\gamma$ problem for L_{PK}, γ and t as defined in this section. Assume T_1' and T_2' are accepting transcripts with the same prover's random tape and different verifier's challenges. Then, when $b'' = 0$ it is possible to obtain from T_1' and T_2' a point p in L_{PK} such that

$$\mathrm{dist}\left(\begin{pmatrix} n^6\sqrt{n}((c'+c) \bmod P(w_1,\ldots,w_n)) \\ 0 \end{pmatrix}, p\right) \le \gamma t.$$

When $b'' = 1$ it is possible to obtain a point p such that

$$\mathrm{dist}\left(\begin{pmatrix} n^6\sqrt{n}((c'+c-\frac{v_k}{2}) \bmod P(w_1,\ldots,w_n)) \\ 0 \end{pmatrix}, p\right) \le \gamma t.$$

Since T_1 is an accepting transcript, ciphertext c' and b',r' satisfy $c' = \mathcal{E}'_{PK}(b'; r')$. `Extract` outputs the witness $(c', b', b_1', \ldots, b_m', b'', p)$. The algorithm succeeds with probability at least $\frac{1}{32}\sigma$.

We prove that the PPK protocol is computational zero-knowledge under the same intractability assumption of the AD cryptosystem.

Assumption ISVP:

(Infeasibility of Shortest Vector Problem): There is no polynomial time algorithm, which given an arbitrary basis for an n-dimensional lattice, having a "unique $poly(n)$-shortest" vector, finds the shortest non-zero vector in the lattice. By having a "unique $poly(n)$-shortest" vector we mean that any vector of length at most "poly(n)" times bigger than the shortest vector is parallel to the shortest vector.

Theorem 48 (Zero-Knowledge of PPK) *The protocol PPK is computational zero knowledge under the assumption ISVP.*

Proof. For every verifier V' we construct an expected polynomial time simulator S such that on input (PK, c) where PK an ε-good public key of AD and c is an $(\varepsilon, \varepsilon_1)$-good ciphertext encrypted under PK the output of the simulator is computationally indistinguishable from a transcript of the protocol between the P_{PPK} and the verifier V' on common input (PK, c).

The simulator S proceeds as follows:

Simulate prover's first step: Chose Δ uniformly from $\{0, 1\}$. If $\Delta = 0$ uniformly select a random bit b' and generate a random $(\varepsilon, \varepsilon_1)$-good ciphertext c' of b' under PK using uniformly generated random string $r' \in \{0, 1\}^m$. If $\Delta = 1$ uniformly select a bit b'' and generate a random $(\varepsilon, \varepsilon_1)$-good ciphertext $\overline{c}$ of b'', set $c' = (\overline{c} - c) \mod P(w_1, \ldots, w_n)$. Pass c' to the verifier V'.

Simulate verifiers's first step: Receive a challenge bit δ from V'.

Simulate prover's second step and output the transcript of the protocol:

If $\delta \neq \Delta$ go to the step "Simulate prover's first step".

Let us show that the simulator repeats the step "Simulate prover's first step" only an expected polynomial number of times. Let U be the uniform distribution in $P(w_1, \ldots, w_n)$. We assume that ISVP holds, hence according to the security property of AD if $\Delta = 0$ then c' is computationally indistinguishable from U; if $\Delta = 1$ then $c' = (\overline{c} - c) \mod P(w_1, \ldots, w_n)$ is also indistinguishable from U. c' generated for $\Delta = 0$ is computationally indistinguishable from c' generated for $\Delta = 1$. δ equal to Δ with probability less than $\frac{1}{2} + v(n)$ for some negligible function $v(n)$, otherwise verifier can distinguish between c' generated for $\Delta = 0$ and c' generated for $\Delta = 1$. Thus the expected number of repetitions of the step "Simulate prover's first step" is polynomial.

- If $\delta = 0$ send bits b' and r' to V' and receive a verifier's verdict v on acceptance or rejectance. Output the transcript (c', δ, b', r', v). Since c' is indeed an $(\varepsilon, \varepsilon_1)$-good ciphertext of b' with random bits r', the simulator perfectly simulates a real transcript between P_{PPK}-prover and the verifier V'.
- Consider the case when $\delta = 1$. Note, that $\overline{c} = (c + c') \mod P(w_1, \ldots, w_n)$, hence, according to zero-knowledge property of $Protocol_0'$ and $Protocol_1'$, there exist simulators S_1 and S_2 with the following properties: if $b'' = 0$ then an output of S_1 on input $(PK, \overline{c})$ is computationally indistinguishable from the real transcript of $Protocol_0'$ run between the P_{PPK} and the verifier V'. If $b'' = 1$ then output of S_2 on input $(PK, \overline{c})$ is indistinguishable from the real transcript of $Protocol_1'$. If $b'' = 0$ set $T = S_1(PK, \overline{c})$ otherwise set $T = S_2(PK, \overline{c})$. Output the transcript (c', δ, b'', T). Since the ISVP assumption holds, according to the security property of AD the distribution of $\overline{c}$ is computationally indistinguishable from U, hence the distribution of $c' = (\overline{c} - c) \mod P(w_1, \ldots, w_n)$ is also indistinguishable from U which is indistinguishable from the distribution of c' generated by the P_{PPK}. Therefore, the generated transcript is computationally indistinguishable from a real transcript of the protocol between the P_{PPK} and V'.

missing proof of claim 46.

(1) The vector c can decrypts to '0' in two cases: when both c_1 and c_2 are ciphertexts of '0' and when both c_1 and c_2 are ciphertexts of '1'.

- Let c_1 and c_2 be $(\varepsilon, \varepsilon_1)$-good ciphertexts of '0'. For c_1 and c_2 equation (2.2) holds. Thus for $c = (c_1 + c_2) \mod P(w_1, \ldots, w_2)$ by lemma 49 below

$$\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}c \\ 0 \end{pmatrix}, L_{PK}\right) \leq 2\sqrt{1 + \frac{1}{2\varepsilon\varepsilon_1}}n^4 + \sqrt{n}$$

  which is less then t for sufficiently large n. By the definition of a YES instance of the GapCVP$_\gamma$ the statement of part (1) holds.
- Let c_1 and c_2 be $(\varepsilon, \varepsilon_1)$-good ciphertexts of '1'. By the definition of an $(\varepsilon, \varepsilon_1)$-good ciphertext of '1' the vectors $\overline{c_1} = (c_1 - \frac{v_k}{2}) \mod P(w_1, \ldots, w_n)$ and $\overline{c_2} = (c_2 - \frac{v_k}{2}) \mod$ are $(\varepsilon, \varepsilon_1)$-good ciphertexts of '0', thus for $\overline{c} = (\overline{c_1} + \overline{c_2}) \mod P(w_1, \ldots, w_n)$ the following statement holds:

$$\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}\overline{c} \\ 0 \end{pmatrix}, L_{PK}\right) \leq 2\sqrt{1 + \frac{1}{2\varepsilon\varepsilon_1}}n^4 + \sqrt{n}$$

  The vector $c = (\overline{c} + v_k) \mod P(w_1, \ldots, w_n)$ thus by lemma 410 below for sufficiently large n the following statement holds:

$$\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}c \\ 0 \end{pmatrix}, L_{PK}\right) \leq 2\sqrt{1 + \frac{1}{2\varepsilon\varepsilon_1}}n^4 + \sqrt{n} + n^4 \tag{4.1}$$

  Expression (4.1) is less than t for sufficiently large n.

(2) Let $c \in P(w_1, \ldots, w_n)$ be any vector which decrypts to '1'. Let $T = t\gamma$. From (2.4) it follows that $\frac{3T}{n^6\sqrt{n}} = 12\frac{\sqrt{1+\frac{1}{2\varepsilon\varepsilon_1}}\sqrt{n+n^3}}{n^2\sqrt{n}\sqrt{\log(n+n^3)}} \leq 12\sqrt{2}\frac{\sqrt{1+\frac{1}{2\varepsilon\varepsilon_1}}}{n\sqrt{\log(n+n^3)}} \leq \frac{1}{4} - \frac{2}{n^2} < \operatorname{dist}(\langle c, u\rangle, \mathbb{Z})$.

Hence, by theorem 33 $\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}c \\ 0 \end{pmatrix}, L_{PK}\right) \leq T$ can not hold. Thus $\left(L_{PK}, t, \begin{pmatrix} n^6\sqrt{n}c \\ 0 \end{pmatrix}\right)$ is a NO instance of the GapCVP$_\gamma$.

The following lemmas complete the proof.

Lemma 49 *Let $PK = \{w_1, \ldots, w_n, v_1, \ldots, v_m, k\}$ be a public key of AD, p_1 and p_2 be points from L_{PK}. If for $c_1, c_2 \in P(w_1, \ldots, w_n)$*
$\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}c_1 \\ 0 \end{pmatrix}, p_1\right) = D_1$ *and* $\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}c_2 \\ 0 \end{pmatrix}, p_2\right) = D_2$ *then*

$$\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}((c_1 + c_2) \mod P(w_1, \ldots, w_n)) \\ 0 \end{pmatrix}, L_{PK}\right) \leq D_1 + D_2 + \sqrt{n}.$$

Proof. We can represent $n^6\sqrt{n}((c_1 + c_2) \mod P(w_1, \ldots, w_n)) = n^6\sqrt{n}(c_1 + c_2 + \sum_{i=1}^n a_i w_i)$. Since both vectors c_1 and c_2 belong to $P(w_1, \ldots, w_n)$ we can bound

$|a_i| \le 1$ for all i. Consider a vector $p_3 = B_{PK}(a_1, \dots, a_n, 0, \dots, 0)^t$ where B_{PK} is the matrix defined in (3.2).

$$\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}\sum_{i=1}^n a_i w_i \\ 0 \end{pmatrix}, p_3\right) = \sqrt{\sum_{i=1}^n a_i^2} \le \sqrt{n}.$$

The lemma follows.

Lemma 410 *Let $PK = \{w_1, \dots, w_n, v_1, \dots, v_m, k\}$ be a public key of AD, and p be a point from L_{PK}. If for $c \in P(w_1, \dots, w_n)$ $\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}c \\ 0 \end{pmatrix}, p\right) = D$ then for sufficiently large n*

$$\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}((c+v_k) \mod P(w_1, \dots, w_n)) \\ 0 \end{pmatrix}, L_{PK}\right) \le D + n^4.$$

Proof. We can represent $n^6\sqrt{n}((c+v_k) \mod P(w_1, \dots, w_n)) = n^6\sqrt{n}(c + v_k + \sum_{i=1}^n a_i w_i)$. Consider a point p' from L_{PK} such that $p' = B_{PK}(a_1, \dots, a_n, 0, \dots, 0, 1, 0, \dots, 0)^t$ (with '1' at the $(n+k)$-th position). It is easy to see that

$$\operatorname{dist}\left(\begin{pmatrix} n^6\sqrt{n}(v_k + \sum_{i=1}^n a_i w_i) \\ 0 \end{pmatrix}, p'\right) \le \sqrt{n^5 + \sum_{i=1}^n a_i^2}. \tag{4.2}$$

Let us bound $\sum_{i=1}^n a_i^2$. Note, that $a_i = \lfloor \theta_i \rfloor$ for θ_i defined as $c + v_k = \sum_{i=1}^n \theta_i w_i$. Since the width of the parallelepiped $P(w_1, \dots, w_n)$ is greater than $\frac{\rho_n}{n^2}$, we can bound $\sum_{i=1}^n a_i^2$ as follows:

$$\sum_{i=1}^n a_i^2 \le \sum_{i=1}^n \theta_i^2 \le \frac{n^4}{\rho_n^2}\sum_{i=1}^n \left\langle c + v_k, w_i^\perp \right\rangle^2 \le \frac{n^5}{\rho_n^2}\|c + v_k\|^2 \le \frac{n^5}{\rho_n^2}(\|c\| + \|v_k\|)^2 \le 4n^7$$

Expression (4.2) is less than n^4 for sufficiently large n. The lemma follows.

5 Open Problems

There are a great deal of open problems. We highlight a few here.

Verifiable Decryption for AD cryptosystem. The AD cryptosystem is a probabilistic scheme for which in the process of decryption, the legal decryptor who knows the private key computes the plaintext without being able to recover the randomness used by the encryptor. This latter task, requires the ability to solve subset sum problem instances. A similar situation holds with respect to the El-Gamal and Cramer-Shoup cryptosystems [10, 6] in which a legal decryptor who knows the private key can decrypt, and yet cannot recover the randomness used by an encryptor, as that would require solving discrete log problem instances.

Such cryptosystems raise an interesting challenge: can a legal decryptor, who knows the private-key of the cryptosystem but does not know the randomness used in the computation of a given ciphertext, prove to a third party that a given ciphertext corresponds to a cleartext without revealing his private key?[5] A cryptosystem for which this can be done was named a *verifiable decryption* scheme by Camenisch and Shoup in [5]. The the challenge is to do this efficiently for the AD cryptosystem. In principle it is achievable based on the existence of one-way functions (which is implied in the context of encryption in any case) using general computational zero-knowledge proofs for NP statements [17].

Non Malleable Proofs of Plaintext Knowledge for the AD cryptosystem. Katz[20] shows efficient non-malleable PPKs for the Blum-Goldwasser RSA and Rabin based encryption, Paillier and El-Gamal, and gets as an application CCA2 secure efficient interactive encryption schems. A promising open problem (although far from obvious) is to design an efficient **non-malleable** PPK for the AD cryptosysetm, and thus obtain a CCA2 secure efficient interactive encryption variant of the AD cryptosystem. One obstacle in tackling this problem is that Katz's protocol utilizes one-time signatures (which although exist in principle under ISVP) for which there are no efficient constructions under ISVP.

Regev Crytosystem. In this paper we addressed the AD cryptoststem. Design a PPK for the Regev cryptosystem, and address the above open problems for the Regev cryptosystem.

Acknowledgment: This work was supported in part by NSF Cybertrust 043045, a Minerva project grant 8495 and grant from Potters Wheel Foundation.

References

1. M. Ajtai and C. Dwork. A Public-Key Cryptosystem with Worst-Case/Average-Case Equivalence. *ECCC, TR96-065, Dec. 1996.*
2. N. Asokan, V. Shoup, M. Waidner. Optimistic Fair Exchange of Digital Signatures (Extended Abstract). *EUROCRYPT 1998: 591-606*
3. M. Bellare and M. Yung, Certifying Permutations: Noninteractive Zero-Knowledge Based on Any Trapdoor Permutation *J. Cryptology 9(3): 149-166 (1996)*
4. J. Camenisch and I. Damgard Verifiable Encryption, Group Encryption, and Their Applications to Separable Group Signatures and Signature Sharing Schemes. *ASIACRYPT 2000: 331-345*
5. J. Camenisch and V. Shoup. Practical Verifiable Encryption and Decryption of Discrete Logarithms. *CRYPTO 2003: 126-144*
6. R. Cramer, V. Shoup. A Practical Public Key Cryptosystem Provably Secure against Adaptive Chosen Ciphertext Attack. (1998)
7. R. Cramer, V. Shoup. Universal Hash Proofs and a paradigm for adaptive chosen ciphertext secure public key encryption. *Eurocrypt 2002.*

[5] In other words, is there a verifiable encryption scheme for the equivalence relation by a prover who does not know the randomness used to encrypt.

8. D. Dolev, C. Dwork, M. Naor: Nonmalleable Cryptography. SIAM J. Comput. 30(2): 391-437 (2000)
9. W. Diffie, M. E. Hellman. New Directions in Cryptography . IEEE Transactions on Information Theory 1976.
10. T. ElGamal. A Public Key cryptosystem and a signature scheme based on discrete logarithm. Proceedings of Crypto 84.
11. U. Feige, D. Lapidot, A. Shamir: Multiple NonInteractive Zero Knowledge Proofs Under General Assumptions. *SIAM J. Comput. 29(1): 1-28 (1999)*
12. U. Feige, A. Fiat, A. Shamir: Zero Knowledge Proofs of Identity. Journal of Cryptology1(2):77-94 (1988).
13. Zvi Galil, Stuart Haber, Moti Yung. Symmetric Public-Key Encryption. CRYPTO 1985: 128-137
14. O. Goldreich, S. Goldwasser. On the Limits of Nonapproximability of Lattice Problems. *JCSS 60(3): 540-563 (2000)*
15. O. Goldreich, S. Goldwasser, and S. Halevi. Eliminating decryption errors in the Ajtai-Dwork cryptosystem. In: *Advances of Cryptology, Proc of Crypto'97 Lecture Notes in Computer Science, 1997.*
16. O. Goldreich. Foundations of Cryptography: Basic Tools. Cambridge University Press, Cambridge, UK, 2001
17. O. Goldreich, S. Micali, and A. Wigderon. Proofs that Yield Nothing but their Validity or NP in Zero Knowledge. JACM 91.
18. S. Halevi and S. Micali. More on Proofs of Knowledge. *LCS Document Number: MIT-LCS-TM-578*
19. Chris Hall, Ian Goldberg, Bruce Schneier, Reaction Attacks Against Several Public-Key Cryptosystems. Proceedings of Information and Communication Security, ICICS'99
20. Jonathan Katz. Efficient and Non-Malleable Proofs of Plaintext Knowledge and Applications. Eurocrypt 2003 .
21. D. Micciancio and S. Vadhan. Statistical zero-knowledge proofs with efficient provers: lattice problems and more. *Advances in Cryptology - Crypto 2003. Santa Barbara, CA, USA, August 2003. LNCS 2729, Springer.*
22. M. Naor , M. Yung, Public-key cryptosystems provably secure against chosen ciphertext attacks, *Proceedings of the twenty-second annual ACM symposium on Theory of computing, p.427-437, May 13-17, 1990, Baltimore, Maryland, United States*
23. P. Nguyen and J. Stern. Cryptanalysis of the Ajtai-Dwork cryptosystem. *In Advances in Cryptology: Proceedings of Crypto '98, volume 1462 of Lecture Notes in Computer Science, pages 223–242. Springer-Verlag, 1998.*
24. O. Regev, *New Lattice Based Cryptographic Constructions*, *STOC 2003.*
25. R. L. Rivest, A. Shamir, L. Adleman Public key cryptography , CACM 21, 120-126, 1978.
26. M. Stadler. Publicly Verifiable Secret Sharing. *EUROCRYPT 1996: 190-199*