Group Oriented Attribute-based Encryption Scheme from Lattices with Shamir's Secret Sharing scheme

Maharage Nisansala Sevwandi Perera¹,
Toru Nakamura², Takashi Matsunaka¹,
Hiroyuki Yokoyama¹, and Kouichi Sakurai³

1: Advanced Telecommunications Research Institute International (ATR), Kyoto, Japan
2: KDDI Research, Inc., Saitama, Japan
3: Kyushu University, Fukuoka, Japan

NSS 2023 – Canterbury, UK
15th Aug., 2023
Abstract

We construct Group Oriented (GO) Attribute-based Encryption (ABE) scheme (GO-ABE scheme) using the post-quantum cryptographic primitive lattices and employ Shamir's secret sharing scheme to satisfy GO-ABE requirements.
Abstract

We construct Group Oriented (GO) Attribute-based Encryption (ABE) scheme (GO-ABE scheme) using the post-quantum cryptographic primitive lattices and employ Shamir's secret sharing scheme to satisfy GO-ABE requirements.

Content

Group Oriented Attribute-based Encryption
- Attribute-based Encryption (ABE) : KP-ABE and CP-ABE
- GO-ABE Scheme
- Requirement of GO-ABE

Post-quantum construction of GO-ABE (our Goal)
- Post-quantum primitive – Lattices
- Need of Shamir’s Secret Sharing Scheme
- Post-quantum (step by step) construction
- Summary with Limitations
GROUP ORIENTED ATTRIBUTE BASED ENCRYPTION SCHEME
Attribute-based Encryption (ABE)

Public-key Encryption (PKE)

- Encryption
  - Public key
  - Encrypt
  - Secret key
  - Decrypt

- Decryption
  - text

Attribute-based Encryption (ABE)

- Encryption
  - Public key
  - Encrypt

- Decryption
  - Secret key
  - Decrypt

Attributes &
Access Policy

\(\Lambda\)
**Key Policy ABE:** Ciphertext is associated to an attribute set; private key associated to a policy. Policy decides which data can be decrypted.

**KP-ABE:** Ciphertext is associated to a policy; private key associated to an attribute set. Policy decides who can decrypt the data.

**CP-ABE:** Ciphertext is associated to a policy; private key associated to an attribute set. Policy decides who can decrypt the data.

---

**Data 1:**
- News: Celebrity
- Channel: Movies
- Place: Asia

**Data 2:**
- News: Matches
- Channel: Sports
- Place: Japan

**Data 3:**
- News: Culture
- Channel: Cartoon
- Place: Europe

---

**User 1:**
- Name: Alex
- Position: Doctor
- Place: ABC Hospital

**User 2:**
- Name: Ann
- Position: Patient
- Place: Japan

**User 3:**
- Name: Charles
- Position: Clark
- Place: ABC Hospital
GO-ABE [Li et al. 2015]

- Group Oriented Attribute-based Encryption (GO-ABE) was introduced by Li et al. in NSS2015
- Group Oriented Attribute-based Encryption (GO-ABE) allows
  - **Users** from the **Same Group** to cooperate to decrypt a ciphertext
  - Without revealing their **secret keys**

“Users from the same group are able to cooperate with each other to decrypt a ciphertext encrypted under a set of attributes $\alpha$ such that a single user may not have enough attributes to match the attribute set $\alpha$”

[Li et al. 2015].
In a company structure, it is obvious requiring high level managers involvement from different departments to access company confidential data probably saved in the cloud.
But CP-ABE allows a single party who possesses all the required attributes to access data. It is not practical because no manager may hold all the positions from different departments.
Requirement of GO-ABE – Confidential Data Access

Allow managers from all required departments to collaborate for accessing data – which is the real requirement of company structure.
Requirement of GO-ABE – Access Patient Data [Li et al. 2015]

Public key
Encrypt
Patient Data

Access Policy

To access data decrypting person should be a Cardiologist and Gastroenterologist

Secret key
Decrypt

Attributes

Doctor 1: Cardiologist
Doctor 2: Gastroenterologist

Doctor 3: Dermatologists
Doctor 4: Endocrinologists

Doctor 1 (Cardiologist) and Doctor 2 (Gastroenterologist) collaborate
## GO-ABE [Li et al. 2015]

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Setup</strong></td>
<td>Security parameter $\lambda$</td>
<td>Public parameter $\text{PK}$&lt;br&gt;Master secret key $\text{MK}$</td>
</tr>
<tr>
<td><strong>Encryption</strong></td>
<td>Public parameter $\text{PK}$&lt;br&gt;Message $M$&lt;br&gt;Access Policy $\mathcal{W}$</td>
<td>Ciphertext $C$</td>
</tr>
<tr>
<td><strong>KeyGen</strong></td>
<td>Public parameter $\text{PK}$&lt;br&gt;Master secret key $\text{MK}$&lt;br&gt;Group id $g$&lt;br&gt;Attribute set $S$</td>
<td>Decryption Key $\text{SK}^g_S$</td>
</tr>
<tr>
<td><strong>Decryption</strong></td>
<td>Ciphertext $C$&lt;br&gt;Public parameter $\text{PK}$&lt;br&gt;Group id $g$</td>
<td>Message $M$</td>
</tr>
</tbody>
</table>

- Satisfies the selective set model security

Cooperating user attribute sets:

$$U = S_1 \cup S_2 \cup \cdots \cup S_N$$

Decrypt if $|\mathcal{W} \cap U| \geq t$, $t$ is the threshold value
Selective Set-model Security

The adversary's goal is to determine which of the two messages is encrypted using the predefined attribute set \( \mathcal{W}^* \).

\[ \mathcal{A} \] is an adversary against selective-set model anonymity.
\[ \mathcal{C} \] is a Challenger.
(1) \( \mathcal{A} \) sends the challenging access structure \( \mathcal{W}^* \).
(2) \( \mathcal{C} \) creates \( \text{PK} \) and \( \text{MK} \).
(3) \( \mathcal{C} \) gives \( \text{PK} \) to \( \mathcal{A} \).
(4) \( \mathcal{A} \) queries private keys for attribute set \( S \neq \mathcal{W}^* \) and
(5) \( \mathcal{C} \) replies with \( \text{SK}^g_S \) quering his own oracle.
(6) \( \mathcal{A} \) sends the message \( b^* \in \{0,1\} \).
(7) \( \mathcal{C} \) selects a random \( r \in \{0,1\} \).
(8) If \( r = 0; c_1^*, c_2^* \) are honest values. Else selects randomly.
(9) \( \mathcal{C} \) outputs \( C^* = (\mathcal{W}^*, c_1^*, c_2^*) \)
(10) \( \mathcal{A} \) sends \( r' \).
If \( r' = r \) then \( \mathcal{A} \) wins.
GO-ABE [Li et al. 2015]

- Group Oriented Attribute-based Encryption (GO-ABE) allows
  - **Users** from the **Same Group** to cooperate to decrypt a ciphertext
  - **Without revealing** their **secret keys**

Users from the same group are able to cooperate with each other to decrypt a ciphertext encrypted under a set of attributes $\alpha$ such that a single user may not have enough attributes to match the attribute set $\alpha$ [Li et al. 2015].

Li et al.’s proposal is constructed using bilinear mappings. Not quantum safe
Our Goal

- Provide a quantum safe construction for the GO-ABE scheme

- What are the supporting primitives / building blocks in our proposal
  - Lattice-based cryptography
  - Shamir’s secret sharing scheme
GO-ABE SCHEME FROM LATTICES
Lattice-based Cryptography

- Is quantum safe because computational problems like Approximate Shortest Independent Vector Problem ($SIVP_\lambda$) not broken (yet).
- We use Learning with error ($LWE$) and Small Integer Solution ($SIS$).
- LWE asked to distinguish LWE samples from truly random samples
- SIS asked to find small non-zero vector $x$, such that $A \cdot x = 0 \mod q$ and $\|x\|_\infty \leq \beta$

For given $(A, z)$, find $(x, e)$

For given $(A)$, find non-zero vector $x$
Shamir’s Secret Sharing (SSS) scheme

- A secret $s$ is split into $\ell$ shares; at least $k$ shares should be combined to reconstruct the secret $s$.
Why we use Shamir’s Secret Sharing (SSS) scheme

GO-ABE Requirement:

- Users should be from the same group
- Users should keep their attribute secret keys secure
Why we use Shamir’s Secret Sharing (SSS) scheme

GO-ABE Requirement:

Users should be from the same group
Users should keep their attribute secret keys secure

SSS allows $J$ shares of $\ell$ shares to construct the origin.
In our construction,
Public key $u = (u_1, u_2, \ldots, u_n)$
Share $u$ among $\ell$ shares, such that $j$-th share vector $\hat{u}_j = (\hat{u}_{j,1}, \ldots, \hat{u}_{j,n})$
The fractional Lagrangian coefficient $L_j$ is calculated such that, $u = \sum_{j \in J} L_j$, where $J \subset [\ell]$

❖ Our proposal does not use SSS to reconstruct a secret; use for proving the users are from the same group.
❖ Shares are used to generate secret keys of individual users.
Our Proposal: GO-ABE scheme construction from Lattices

Trusted Setup Party

Encrypting Party

Patient

Decrypting Party

Cardiologist

Gastroenterologist

System

Decide: group details and attribute set.
The \( l \) for SSS

Decide Access Policy including
the threshold value \( k \)

Key Generator
Let each group has an id $g$ and has unique group public key ($\text{GPK} = (G, G_0, G_1, g)$) and a secret key ($\text{GSK} = T$) selected from $(G, T_G) \leftarrow \text{TrapGen}(n, m, q)$ and $G_0, G_1 \in \mathbb{Z}_q^{m \times n}$ and $g \in \mathbb{Z}_q^n$ randomly.
Our Proposal: GO-ABE scheme construction from Lattices

\[
\text{Setup}(\lambda) \rightarrow (\text{PK} = \{\mathcal{A}_i\}_{i \in [\ell]}, \text{u}), \text{MK} = \{\mathcal{T}_i\}_{i \in [\ell]} \]

1. For all attribute \( i \in \mathcal{A} \):
   \[
   \mathcal{A}^\ell_{i=1}, \mathcal{T}_{i=1}^\ell \leftarrow \text{TrapGen}(n, m, q)
   \]

2. Select vector \( \text{u} \in \mathbb{Z}_q^n \)

\[\text{PK} = \{\mathcal{A}_i\}_{i \in [\ell]}, \text{u} \]
\[\text{MK} = \{\mathcal{T}_i\}_{i \in [\ell]} \]
Our Proposal: GO-ABE scheme construction from Lattices

Encrypt(PK, M, W) → (C = c₁, c₂)

1. Let D ≡ (ℓ!)²
2. Select s ∈ ℤⁿ, for i ∈ [w]: eᵢ ∈ ℤᵐ, and e ∈ ℤ₂
3. c₁ = Aᵢᵀs + Deᵢ for i ∈ [w],
c₂ = uᵀs + De + M[q/2]

Setup(λ) → (PK = (Aᵢ∈[ℓ], u), MK = (Tᵢ∈[ℓ])

1. For all attribute i ∈ A:
   Aᵢ=₁, Tᵢ=₁ ← TrapGen(n, m, q)
2. Select vector u ∈ ℤⁿ

PK = (Aᵢ∈[ℓ], u)
MK = (Tᵢ∈[ℓ])

PK = (Aᵢ∈[ℓ], u)

Our Proposal: GO-ABE scheme construction from Lattices

Encrypt($\text{PK}, M, \mathcal{W}$) → ($C = c_1, c_2$)

1. Let $D \equiv (\ell !)^2$
2. Select $s \in \mathbb{Z}_q^n$, for $i \in [w]: e_i \in \mathbb{Z}_q^m$, and $e \in \mathbb{Z}_q$
3. $c_1 = A_i^T s + De_i$ for $i \in [w]$,
   
   $c_2 = u^T s + D e + M[q/2]$

PK = ($\{A_i\}_{i \in [\ell]}, u$)

Setup($\lambda$) → ($\text{PK} = \{A_i\}_{i \in [\ell]}, \text{MK} = \{T_i\}_{i \in [\ell]}$)

1. For all attribute $i \in A$:
   $A_i^\ell \in [n, \ell], T_i^\ell \leftarrow \text{TrapGen}(n, m, q)$
2. Select vector $u \in \mathbb{Z}_q^n$

PK = ($\{A_i\}_{i \in [\ell]}, u$)
MK = ($\{T_i\}_{i \in [\ell]}$

Key Generator

KeyGen($\text{PK}, \text{MK}, g, S$) → ($\text{SK}^g_S = ((x^d_1, ..., x^d_S), d)$)

1. For a group: $\text{G}, \text{T}_\text{G} \leftarrow \text{TrapGen}(n, m, q)$
   $G_0, G_1 \in \mathbb{Z}_q^{m \times n}, g \in \mathbb{Z}_q$
Set $\text{GPK} = (\text{G}, G_0, G_1, g), \text{GSK} = \text{T}_\text{G}$
2. User id $d \in \mathbb{N}$
3. Use SSS on $u$, such that $u = \sum_{j \in J} L_j \cdot \tilde{u}_j$
4. For $i \in S$:
   $v_i \leftarrow \text{SamplePre}(A_i, T_i, \tilde{u}_i \cdot g, \sigma); A_i \cdot v_i = \tilde{u}_i \cdot g$
5. Compute $G_d = [G|G_0 + dG_1]$ and
   $T_d \leftarrow \text{ExtBasis}(T_G, G_d)$
6. For $i \in S$: $x^d_i \leftarrow \text{SamplePre}(G_d, T_d, v_i, \sigma); G \cdot x^d_i = v_i$
Our Proposal: GO-ABE scheme construction from Lattices

Encrypt(\(PK\), \(M, W\)) \(\rightarrow (C = c_1, c_2)\)

1. Let \(D \equiv (\ell!)^2\)
2. Select \(s \in \mathbb{Z}_q^m\), for \(i \in [w]: e_i \in \mathbb{Z}_q^m\), and \(e \in \mathbb{Z}_q\)
3. \(c_1 = A_i^T s + De_i\) for \(i \in [w]\),
   \(c_2 = u^T s + De + M[q/2]\)

\(PK = (\{A_i\}_{i \in [\ell]}, u)\)

\(MK = (T)_{i \in [\ell]}\)

\(\text{Setup}(\lambda) \rightarrow (PK = (\{A_i\}_{i \in [\ell]}, u), MK = (T)_{i \in [\ell]}\)

1. For all attribute \(i \in A\):
   \(A_i^e, T_i^e \leftarrow \text{TrapGen}(n, m, q)\)
2. Select vector \(u \in \mathbb{Z}_q^n\)

\(PK = (\{A_i\}_{i \in [\ell]}, u)\)

\(MK = (T)_{i \in [\ell]}\)

\(\text{KeyGen}(PK, MK, g, S) \rightarrow (SK^g_S = ((x_1^d, ..., x_d^d), d))\)

1. For a group:
   \(G, T_G \leftarrow \text{TrapGen}(n, m, q)\)
   \(GSK = T_G\)

\(SK^g_S = \text{Cardi}\)

2. User id \(d \in \mathbb{N}\)
3. Use SSS on \(u\), such that \(u = \sum_{j \in J} \hat{u}_j\)
4. For \(i \in S:\)
   \(v_i \leftarrow \text{SamplePre}(A_i, T_i, \hat{u}_i - g, \sigma)\)
   \(A_i \cdot v_i = \hat{u}_i - g\)
5. Compute \(G_d = [G|G_0 + dG_1]\) and
   \(T_d \leftarrow \text{ExtBasis}(T_G, G_d)\)
6. For \(i \in S:\)
   \(x_i^d \leftarrow \text{SamplePre}(G_d, T_d, v_i, \sigma)\)
   \(G \cdot x_i^d = v_i\)

Calculate \(L_i; \sum_{i \in [k]} L_iA_i y_i = u \mod q\)

Compute \(r \leftarrow c_2 - \left((k \times g)^T + \sum_{i \in [k]} L_iy_i^T c_1\right)\)

If \(|r| < \frac{q}{4}\), output 0, else 1 as the message \(M\)
Our Proposal: GO-ABE scheme construction from Lattices

Encrypt(\(\mathbf{PK}, M, \mathcal{W}\)) \rightarrow (C = c_1, c_2)

1. Let \(D \triangleq (\ell!)^2\)
2. Select \(s \in \mathbb{Z}_q^n\), for \(i \in [w]: e_i \in \mathbb{Z}_q^m\), and \(e \in \mathbb{Z}_q\)
3. \(c_1 = A_i^T s + De_i\) for \(i \in [w]\),
   \(c_2 = u^T s + De + M[q/2]\)

\(\mathbf{PK} = (\{A_i\}_{i \in [\ell]}, \mathbf{u})\)

Decrypt(\(\mathbf{PK}, C, g\)) \rightarrow M

Calculate \(L_i; \sum_{i \in [k]} L_i A_i y_i = u \mod q\)
Compute \(r \leftarrow c_2 - ((k \times g)^T + \sum_{i \in [k]} L_i y_i^T c_1)\)
If \(|r| < \frac{q}{4}\), output 0, else 1 as the message \(M\)

KeyGen(\(\mathbf{PK}, \mathbf{MK}, g, S\)) \rightarrow (\(\mathbf{SK}_S^g = ((x_1^d, \ldots, x_s^d), d)\))

1. For a group:
   \(G, T_G \leftarrow \text{TrapGen}(n, m, q)\)
   Set \(\mathbf{GPK} = (G, G_0, G_1, g), \mathbf{GSK} = T_G\)
2. User id \(d \in \mathbb{N}\)
3. Use SSS on \(u\), such that \(u = \sum_{j \in J} L_j \cdot \hat{u}_j\)
4. For \(i \in S:\)
   \(v_i \leftarrow \text{SamplePre}(A_i, T_i, \hat{u}_i - g, \sigma); A_i \cdot v_i = \hat{u}_i - g\)
5. Compute \(G_d = [G|G_0 + dG_1]\) and \(T_d \leftarrow \text{ExtBasis}(T_G, G_d)\)
6. For \(i \in S: x_i^d \leftarrow \text{SamplePre}(G_d, T_d, v_i, \sigma); G \cdot x_i^d = v_i\)

Setup(\(\lambda\)) \rightarrow (\(\mathbf{PK} = (\{A_i\}_{i \in [\ell]}, \mathbf{u}), \mathbf{MK} = (T)_{i \in [\ell]}\))

1. For all attribute \(i \in \mathbb{A}:
   A_i^\ell, T_i^\ell \leftarrow \text{TrapGen}(n, m, q)\)
2. Select vector \(u \in \mathbb{Z}_q^n\)

Trusted Setup Party

\(\mathbf{PK} = (\{A_i\}_{i \in [\ell]}, \mathbf{u})\)

System

\(\mathbf{MK} = (T)_{i \in [\ell]}\)
Security Proof

- Based on the hardness of Decision-LWE problem we proved that Lattice-based construction of GO-ABE scheme provides ciphertext privacy in the Selective-Set model.
- Selective-Set model: The adversary declares the attribute set \( \mathcal{W} \) that he wishes to be challenged upon.

**Theorem 1.** If there is an adversary \( \mathcal{A} \) with advantage \( > 0 \) against the selective-set model for the GO-ABE scheme, then there exists a PPT algorithm \( \mathcal{B} \) that can solve the decision-LWE problem.
Selective Set-model Security

Proof. The simulator $\mathcal{B}$ uses the adversary $\mathcal{A}$ to distinguish LWE oracle $\mathcal{O}$. First $\mathcal{B}$ queries the LWE oracle $\mathcal{O}$ for $(\ell m + 1)$ times and obtain LWE samples $(a_k, b_k) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$, where $k \in \{0,1,2, \ldots, m\}$. Then $\mathcal{B}$ proceeds as below.

<table>
<thead>
<tr>
<th>$\mathcal{B}$</th>
<th>$\mathcal{A}$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Setup</strong></td>
<td></td>
</tr>
<tr>
<td>$(1) \mathcal{W}^*$</td>
<td>$(2) \text{PK, MK}$</td>
</tr>
<tr>
<td>$(3) \text{PK}$</td>
<td></td>
</tr>
<tr>
<td><strong>Query Phase 1</strong></td>
<td></td>
</tr>
<tr>
<td>$(4) S$</td>
<td>$(5) \text{SK}_s^g$</td>
</tr>
<tr>
<td><strong>Challenge</strong></td>
<td></td>
</tr>
<tr>
<td>$(6) b^* \in {0,1}$</td>
<td>$(7) r \in {0,1}$</td>
</tr>
<tr>
<td>$(8) C^*$</td>
<td>$(9) \text{C}^*$</td>
</tr>
<tr>
<td><strong>Guess</strong></td>
<td></td>
</tr>
<tr>
<td>$(10) r'$</td>
<td></td>
</tr>
</tbody>
</table>

Initialize: $\mathcal{A}$ announces to $\mathcal{W}^*$ to $\mathcal{B}$

Setup: $\mathcal{B}$ selects LWE challenges $\{(a_0, b_0), (a_1^1, b_1^1), (a_2^2, b_2^2), \ldots, (a_i^m, b_i^m)\}_{i \in [\ell]}$ for public matrices $\overline{A}_i$ and $a_0$ as $\mathbf{u}$

Phase 1: $\mathcal{B}$ answers each private key query by selecting parameters from LWE

Challenge: When $\mathcal{A}$ sends $b^* \in \{0,1\}$, $\mathcal{B}$ uses $\mathcal{W}^*$ and sets $c_1 = (Db_1^1, Db_2^2, \ldots, Db_i^m)$ for $i \in [\ell]$

$c_2 = Da_0 + M_b [q/2]$ if he wishes to generate $\text{C}^*$. That is $r = 0$. Otherwise he randomly selects values.

Guess: $\mathcal{A}$ outputs $b'$ If
Summary

- We present the Lattice based construction of GO-ABE scheme
- We employed Shamir’s Secret Sharing Scheme to satisfy GO-ABE requirements

**Limitations:**

1. Efficiency is less in decryption because need to collect users’ shares; however, this is reasonable fulfilling practical applications like access company confidential data

2. Only AND-gets on multivalued attributes are considered; not complex access policies
3. There is no tracing mechanism to track cooperated users
4. The cooperating situation is not controlled
5. Issues may occur due to the use of SSS:
   - Ex: If any structural change happens like introducing new attributes, need recreate all the keys

Thank you for Listening

perera.nisansala@atr.jp
Attribute-based Encryption (ABE)

Public-key Encryption (PKE)
公開鍵暗号

Encryption
- Public key
  - Encrypt
  - Ciphertext

Decryption
- Secret key
  - Decrypt
  - Text

Key-Policy Attribute-based Encryption (KP-ABE)

Encryption
- Public key
  - Encrypt
  - Attributes
  - Department = Financial Role= Manager...

Decryption
- Secret key
  - Decrypt
  - Access Policy
  - Text

Ciphertext-Policy Attribute-based Encryption (CP-ABE)

Encryption
- Public key
  - Encrypt
  - Access Policy

Decryption
- Secret key
  - Decrypt
  - Attributes
  - Department = Financial Role= Manager...

ATR Proprietary Information
CP-ABE Application – Patient Health Record System??

Public key → Encrypt → Patient Data → Access Policy → Decrypt → Secret key

To access data decrypting person should be a Cardiologist and Hospital HH

Attributes:
- Doctor 1: Cardiologist
- Doctor 2: Gastroenterologist
- Doctor 3: Dermatologists
- Doctor 4: Endocrinologists
Necessity of GO-ABE [Li et al. 2015]

Necessity of GO-ABE [Li et al. 2015]

To access data decrypting person should be a **Cardiologist and Gastroenterologist**

- **Public key**
  - Encrypt
  - Patient Data
  - Access Policy
  - Attributes
  - Doctor 1: Cardiologist
  - Doctor 2: Gastroenterologist
  - Doctor 3: Dermatologists
  - Doctor 4: Endocrinologists

- **Secret key**
  - Decrypt

Doctor 1 (**Cardiologist**) and Doctor 2 (**Gastroenterologist**) collaborate
Even though both numerator and denominator in $L_i$ can be bounded as a fraction of integers, when presenting Author Proof 6 M. N. S. Perera et al. as an element in $\mathbb{Z}_q$ the value $L_i$ is arbitrarily large. The idea of clearing the denominators prevents the large-value problem of $L_i$. Let $D := (\frac{!}{2})^2$ be a sufficiently large constant, such that $DL_i \in \mathbb{Z}$ for all $i$. Multiplying noise vectors of the encryption function with $D$ we get, $C_{id} = \text{IBE.Enc}(id, b \in \{0, 1\}) = (AT_{1, id} s + D e_1, ..., AT_{, id} s + D e, u^T s + D e + bq/2)$. Thus, it is sufficient to bind the below for the correctness of the IBE scheme by $q/4$. $De_i - k \in S DL_ix_T^i e_i$ Since $DL_i$ is an integer bounded by $D^2$, it is enough to select noise vectors bounded by $q/4D$ with overwhelming probability.
Lattices

Set of points in a n-dimensional space, arranged on a periodical manner.