Multi-Authority ABE from Lattices without Random Oracles

Brent Waters, Hoeteck Wee, and David Wu
Attribute-Based Encryption (ABE)

master secret key

central authority

Secret keys associated with set of attributes

“U. Chicago” “faculty”

“U. Chicago” “student”

“UT” “faculty”

[SW05, GPSW06]
Attribute-Based Encryption (ABE)

- **Policy:** U. Chicago and faculty

- **Central Authority:**
  - Master secret key
  - Secret keys associated with set of attributes

- **Message Encryption:**
  - "U. Chicago" faculty
  - "U. Chicago" student
  - "UT" faculty

[SW05, GPSW06]
Attribute-Based Encryption (ABE)

**Message**

**Policy:** U. Chicago and faculty

- Secret keys associated with set of attributes

- "U. Chicago" and faculty
- "U. Chicago" and student
- "UT" and faculty

Can decrypt

[SW05, GPSW06]
Attribute-Based Encryption (ABE)

message

policy: U. Chicago and faculty

“U. Chicago” faculty
Can decrypt

“U. Chicago” student
Cannot decrypt

“UT” faculty
Cannot decrypt

Secret keys associated with set of attributes
Attribute-Based Encryption (ABE)

- **Policy:** U. Chicago and faculty

Users cannot collude to decrypt
Attribute-Based Encryption (ABE)

Message

Policy: U. Chicago and faculty

Central authority

Master secret key

Single authority controls all attributes

“U. Chicago” faculty

“U. Chicago” student

“UT” faculty
In practice, different authorities control different attributes.
In practice, different authorities control different attributes.

Multi-authority ABE: anyone can become an authority.
Multi-Authority ABE

In practice, different authorities control different attributes.

Multi-authority ABE: anyone can become an authority.
Multi-Authority ABE

In practice, different authorities control different attributes.

Each authority publishes a public key along with the set of attributes it controls.

message

policy: visitor (U Chicago) and student (UT)

policy is a function on attributes from one or more authorities.

Multi-authority ABE: anyone can become an authority.
Multi-Authority ABE

[LW11, RW15, DKW21b]: Multi-authority ABE for NC$^1$ from bilinear maps
Multi-Authority ABE

[Cha07, CC09, LW11]: Multi-authority ABE for $\mathsf{NC}^1$ from bilinear maps

[DKW21a]: Multi-authority ABE for conjunctions from LWE
Multi-Authority ABE

[Cha07, CC09, LW11]: Multi-authority ABE for \(\text{NC}^1\) from bilinear maps

[DKW21a]: Multi-authority ABE for conjunctions from LWE

All of these constructions are in the random oracle model
Multi-Authority ABE

[Cha07, CC09, LW11]: Multi-authority ABE for $\text{NC}^1$ from bilinear maps

[DKW21a]: Multi-authority ABE for conjunctions from LWE

All of these constructions are in the random oracle model

Can we construct multi-authority ABE without random oracles?
Multi-Authority ABE

[Cha07, CC09, LW11]: Multi-authority ABE for $\text{NC}^1$ from bilinear maps

[DKW21a]: Multi-authority ABE for conjunctions from LWE

All of these constructions are in the \textit{random oracle model}

\textit{Can we construct multi-authority ABE without random oracles?}

\textit{(and without strong tools like extractable witness encryption or indistinguishability obfuscation)}
This Work

[LW11, RW15, DKW21b]: Multi-authority ABE for $\text{NC}^1$ from bilinear maps

[DKW21a]: Multi-authority ABE for conjunctions from LWE

All of these constructions are in the random oracle model

This work: instantiate the random oracle in [DKW21a] with a concrete hash function and argue security using the evasive LWE assumption
This Work

[Cha07, CC09, LW11]: Multi-authority ABE for NC^1 from bilinear maps

[DKW21a]: Multi-authority ABE for conjunctions from LWE

All of these constructions are in the random oracle model

Hash function is not “random-looking:”

\[ H(x_1 x_2 \cdots x_n) := \left( \prod_{i \in [n]} D x_i \right) e_1 \]

where \( D_0, D_1 \) are public low-norm matrices and \( e_1 \) is first basis vector

This work: instantiate the random oracle in [DKW21a] with a concrete hash function and argue security using the evasive LWE assumption
This Work

[Cha07, CC09, LW11]: Multi-authority ABE for NC^1 from bilinear maps

[DKW21a]: Multi-authority ABE for conjunctions from LWE

All of these constructions are in the random oracle model

Hash function is not “random-looking:”

\[ H(x_1 x_2 \cdots x_n) := \left( \prod_{i \in [n]} D_{x_i} \right) e_1 \]

where \( D_0, D_1 \) are public low-norm matrices and \( e_1 \) is first basis vector

This work: instantiate the random oracle in [DKW21a] with a concrete hash function and argue security using the evasive LWE assumption

Evasive LWE is not a standard assumption, but provides useful evidence for soundness of the approach
This Work

[LW11, RW15, DKW21b]: Multi-authority ABE for $\text{NC}^1$ from bilinear maps

[DKW21a]: Multi-authority ABE for conjunctions from LWE

All of these constructions are in the random oracle model

Hash function is not “random-looking:”

$$H(x_1x_2 \cdots x_n) := \left(\prod_{i \in [n]} D_{x_i}\right)e_1$$

where $D_0, D_1$ are public low-norm matrices and $e_1$ is first basis vector

This work: instantiate the random oracle in [DKW21a] with a concrete hash function and argue security using the evasive LWE assumption

Open question: prove security from standard LWE
Why Random Oracles?

[LW11, RW15, DKW21b]: Multi-authority ABE for $\text{NC}^1$ from bilinear maps

[DKW21a]: Multi-authority ABE for conjunctions from LWE

All of these constructions are in the random oracle model

Can we construct multi-authority ABE without random oracles?

(and without strong tools like extractable witness encryption or indistinguishability obfuscation)

message

policy: visitor (U Chicago) and student (UT)

• Different users should not be able to combine their keys to decrypt
Why Random Oracles?

[LW11, RW15, DKW21b]: Multi-authority ABE for NC¹ from bilinear maps

[DKW21a]: Multi-authority ABE for conjunctions from LWE

All of these constructions are in the **random oracle model**

*Can we construct multi-authority ABE without random oracles?*

*(and without strong tools like extractable witness encryption or indistinguishability obfuscation)*

**Single-authority setting**: generate all of the attribute keys for a user using common randomness to prevent mixing and matching across users

- Different users should not be able to combine their keys to decrypt
Why Random Oracles?

[LW11, RW15, DKW21b]: Multi-authority ABE for NC¹ from bilinear maps

[DKW21a]: Multi-authority ABE for conjunctions from LWE

All of these constructions are in the random oracle model

Can we construct multi-authority ABE without random oracles?
(and without strong tools like extractable witness encryption or indistinguishability obfuscation)

message

policy: visitor (U Chicago) and student (UT)

• Different users should not be able to combine their keys to decrypt
• Keys for a single user are generated using correlated randomness (derived by hashing unique user identifier: \( r \leftarrow H(\text{gid}) \))
Why Random Oracles?

[LW11, RW15, DKW21b]: Multi-authority ABE for NC¹ from bilinear maps

[DKW21a]: Multi-authority ABE for conjunctions from LWE

All of these constructions are in the random oracle model

Can we construct multi-authority ABE without random oracles?
(and without strong tools like extractable witness encryption or indistinguishability obfuscation)

message

policy: visitor (U Chicago) and student (UT)

Security proof needs to model \( H \) as a random oracle

• Different users should not be able to combine their keys to decrypt
• Keys for a single user are generated using correlated randomness (derived by hashing unique user identifier: \( r \leftarrow H(\text{gid}) \))
Starting point: ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute
Starting point: ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute

Public key for each authority/attribute consist of (random) matrices $A_i, B_i$ and vector $p_i$ (over $\mathbb{Z}_q$)
Starting point: ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute

Public key for each authority/attribute consist of (random) matrices $A_i, B_i$ and vector $p_i$ (over $\mathbb{Z}_q$)

Secret key for each authority/attribute is trapdoor $td_i$ for $A_i$
Starting point: ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute

Public key for each authority/attribute consist of (random) matrices $A_i, B_i$ and vector $p_i$ (over $\mathbb{Z}_q$)

Secret key for each authority/attribute is trapdoor $td_i$ for $A_i$

Trapdoor for $A_i$ can be used to sample short solution $x$ where $A_i x = y$

We denote this by writing $x \leftarrow A_i^{-1}(y)$
Starting point: ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute

- Authority 1: $A_1, B_1, p_1$, $t_1$
- Authority 2: $A_2, B_2, p_2$, $t_2$
- Authority 3: $A_3, B_3, p_3$, $t_3$

$r \leftarrow H(\text{gid})$
Construction Overview

**Starting point:** ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute

\[ A_1, B_1, p_1 \]
\[ td_1 \]

\[ A_2, B_2, p_2 \]
\[ td_2 \]

\[ A_3, B_3, p_3 \]
\[ td_3 \]

\[ r \leftarrow H(gid) \]
\[ k_1 \leftarrow A_1^{-1}(p_1 + B_1 r) \]

Invariant: \( A_i k_i = p_i + B_i r \)
Construction Overview

Starting point: ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute

\[ \text{Invariant: } A_i k_i = p_i + B_i r \]
**Starting point:** ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute

Encrypt to these attributes

Authority 1

\[ A_1, B_1, p_1 \]
\[ t_d_1 \]

Authority 2

\[ A_2, B_2, p_2 \]
\[ t_d_2 \]

Authority 3

\[ A_3, B_3, p_3 \]
\[ t_d_3 \]
Starting point: ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute

Encrypt to these attributes

\[ A_1, B_1, p_1 \]
\[ td_1 \]

\[ A_2, B_2, p_2 \]
\[ td_2 \]

\[ s^T A_1 \]

\[ s^T A_2 \]

squiggly underline denotes noise

\[ s^T A = s^T A + \text{error} \]
Starting point: ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute

Encrypt to these attributes

\[
A_1, B_1, p_1 \\
\text{td}_1
\]

\[
A_2, B_2, p_2 \\
\text{td}_2
\]

\[
A_3, B_3, p_3 \\
\text{td}_3
\]

\[
s^T A = s^T A + \text{error}
\]
Starting point: ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute

Encrypt to these attributes

\[ A_1, B_1, p_1 \]
\[ td_1 \]

\[ A_2, B_2, p_2 \]
\[ td_2 \]

\[ s_1^T A_1 \]
\[ s_2^T A_2 \]

\[ s_1^T B_1 + s_2^T B_2 \]
\[ s_1^T p_1 + s_2^T p_2 + \mu \cdot \lfloor q/2 \rfloor \]

squiggly underline denotes noise

\[ s^T A = s^T A + \text{error} \]
Construction Overview

**Starting point:** ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute

![Diagram](image)

Encrypt to these attributes

\[
\begin{align*}
A_1, B_1, p_1 & \quad s_1^T A_1 \\
A_2, B_2, p_2 & \quad s_2^T A_2 \\
td_1 & \\
td_2 &
\end{align*}
\]

Decryption:

\[
\begin{align*}
\mathbf{r} & \leftarrow H(\text{gid}) \\
\mathbf{k}_1 & \leftarrow A_1^{-1}(p_1 + B_1 \mathbf{r}) \\
\mathbf{k}_2 & \leftarrow A_2^{-1}(p_2 + B_2 \mathbf{r}) \\
gid &
\end{align*}
\]

Squiggly underline denotes noise

\[
s^T A = s^T A + \text{error}
\]

\[
\begin{align*}
\mathbf{s}_1^T B_1 + \mathbf{s}_2^T B_2 & \\
\mathbf{s}_1^T p_1 + \mathbf{s}_2^T p_2 + \mu \cdot \lceil q/2 \rceil
\end{align*}
\]
Construction Overview

**Starting point:** ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute

For the authority 1, encrypt to these attributes:

\[ A_1, B_1, p_1 \]
\[ td_1 \]

For the authority 2, encrypt to these attributes:

\[ A_2, B_2, p_2 \]
\[ td_2 \]

Decryption:

\[ s_1^T A_1 k_1 \approx s_1^T B_1 r + s_1^T p_1 \]
\[ s_2^T A_2 k_2 \approx s_2^T B_2 r + s_2^T p_2 \]

$s^T A = s^T A + \text{error}$

squiggly underline denotes noise

$s^T B_1 + s_2^T B_2$

$s^T p_1 + s_2^T p_2 + \mu \cdot [q/2]$

\[ r \leftarrow H(\text{gid}) \]
\[ k_1 \leftarrow A_1^{-1}(p_1 + B_1 r) \]
\[ k_2 \leftarrow A_2^{-1}(p_2 + B_2 r) \]
Construction Overview

**Starting point:** ABE for conjunctions from LWE [DKW21a]

For simplicity, assume each authority has one attribute

Encrypt to these attributes

- Authority 1: $A_1, B_1, p_1$
  - $td_1$

- Authority 2: $A_2, B_2, p_2$
  - $td_2$

Encryption:

- Encrypt to $gid$
  - $s_1^T A_1$
  - $s_2^T A_2$

Decryption:

- $s_1^T B_1 + s_2^T B_2$
- $s_1^T p_1 + s_2^T p_2 + \mu \cdot [q/2]$

Subtract to obtain:

- $\mu \cdot [q/2] + \text{noise}$

Squiggly underline denotes noise

$s^T A = s^T A + \text{error}$
Security Analysis

public keys

Authority 1
$A_1, B_1, p_1$

Authority 2
$A_2, B_2, p_2$

challenge ciphertext

$s_1^T A_1$
$s_1^T B_1 + s_2^T B_2$

$s_2^T A_2$
$s_1^T p_1 + s_2^T p_2 + \mu \cdot [q/2]$
### Security Analysis

#### Strategy:
Argue ciphertext is pseudorandom (by LWE) if none of the keys satisfy the policy

<table>
<thead>
<tr>
<th>Public keys</th>
<th>Challenge ciphertext</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_1, B_1, p_1$</td>
<td>$s_1^T A_1$</td>
</tr>
<tr>
<td>$A_2, B_2, p_2$</td>
<td>$s_2^T A_2$</td>
</tr>
<tr>
<td>$s_1^T B_1 + s_2^T B_2$</td>
<td>$s_1^T p_1 + s_2^T p_2 + \mu \cdot [q/2]$</td>
</tr>
</tbody>
</table>

**Strategy:** Argue ciphertext is pseudorandom (by LWE) if **none** of the keys satisfy the policy.
Security Analysis

**public keys**

<table>
<thead>
<tr>
<th>Authority 1</th>
<th>Authority 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_1, B_1, p_1$</td>
<td>$A_2, B_2, p_2$</td>
</tr>
</tbody>
</table>

**challenge ciphertext**

- $s_1^T A_1$  
- $s_2^T A_2$  
- $s_1^T B_1 + s_2^T B_2$  
- $s_1^T p_1 + s_2^T p_2 + \mu \cdot [q/2]$  

**secret key**

- $r_1 \leftarrow H(gid_1)$  
- $k_1 \leftarrow A_1^{-1}(p_1 + B_1 r_1)$  
- $r_2 \leftarrow H(gid_2)$  
- $k_2 \leftarrow A_2^{-1}(p_2 + B_2 r_2)$

**Strategy**: Argue ciphertext is pseudorandom (by LWE) if none of the keys satisfy the policy

**Challenge**: Need to simulate keys $k_1$ and $k_2$ without trapdoors for $A_1$ or $A_2$
Security Analysis

public keys

\[ A_1, B_1, p_1 \]

Authority 1

\[ A_2, B_2, p_2 \]

Authority 2

challenge ciphertext

\[
\begin{align*}
  s_1^T A_1 &\equiv - s_1^T B_1 + s_2^T B_2 \\
  s_2^T A_2 &\equiv s_1^T p_1 + s_2^T p_2 + \mu \cdot [q/2]
\end{align*}
\]

secret key

\[
\begin{align*}
  r_1 &\leftarrow H(\text{gid}_1) \\
  k_1 &\leftarrow A_1^{-1}(p_1 + B_1 r_1) \\
  r_2 &\leftarrow H(\text{gid}_2) \\
  k_2 &\leftarrow A_2^{-1}(p_2 + B_2 r_2)
\end{align*}
\]

Strategy: Argue ciphertext is pseudorandom (by LWE) if none of the keys satisfy the policy

\[
\begin{align*}
  s_1^T B_1 &+ s_2^T B_2 \\
  s_1^T p_1 &+ s_2^T p_2 + \mu \cdot [q/2] \\
  s_2^T A_1 &+ s_1^T A_2
\end{align*}
\]

Challenge: Need to simulate keys \( k_1 \) and \( k_2 \) without trapdoors for \( A_1 \) or \( A_2 \)

Previously [DKW21a]: model \( H \) as a random oracle and rely on “lattice trapdoor sampling” lemma

- **This work:** We describe a modular approach that allows us to use LWE with a polynomial modulus-to-noise ratio (as opposed to a sub-exponential modulus-to-noise ratio)

[see paper for details]
Security Analysis

public keys

Authority 1
$A_1, B_1, p_1$

Authority 2
$A_2, B_2, p_2$

challenge ciphertext

$s_1^TA_1 \gets s_1^TB_1 + s_2^TB_2$
$s_2^TA_2 \gets s_1^Tp_1 + s_2^Tp_2 + \mu \cdot [q/2]$

secret key

$r_1 \gets H(gid_1)$
$k_1 \gets A_1^{-1}(p_1 + B_1r_1)$

$r_2 \gets H(gid_2)$
$k_2 \gets A_2^{-1}(p_2 + B_2r_2)$

Strategy: Argue ciphertext is pseudorandom (by LWE) if none of the keys satisfy the policy

Challenge: Need to simulate keys $k_1$ and $k_2$
without trapdoors for $A_1$ or $A_2$

Evasive LWE [Wee22, Tsa22]:

if $([A \mid P], s^T[A \mid P]) \approx ([A \mid P], u)$
Security Analysis

Evasive LWE [Wee22, Tsa22]:

\[
\text{if } ([A \mid P], s^T[A \mid P]) \approx ([A \mid P], u) \\
\text{then } ([A \mid P], s^T A \cdot A^{-1}(P)) \approx ([A \mid P], u, A^{-1}(P))
\]

Challenge: Need to simulate keys \(k_1\) and \(k_2\) without trapdoors for \(A_1\) or \(A_2\)

Strategy: Argue ciphertext is pseudorandom (by LWE) if none of the keys satisfy the policy

\[
s_1^T A_1 \approx s_1^T B_1 + s_2^T B_2 \\
s_2^T A_2 \approx s_1^T p_1 + s_2^T p_2 + \mu \cdot [q/2]
\]

<table>
<thead>
<tr>
<th>Authority 1</th>
<th>Authority 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A_1, B_1, p_1)</td>
<td>(A_2, B_2, p_2)</td>
</tr>
</tbody>
</table>

\[
r_1 \leftarrow H(\text{gid}_1) \\
k_1 \leftarrow A_1^{-1} (p_1 + B_1 r_1) \\
r_2 \leftarrow H(\text{gid}_2) \\
k_2 \leftarrow A_2^{-1} (p_2 + B_2 r_2)
\]
Security Analysis

Evasive LWE [Wee22, Tsa22]:

\[
\begin{align*}
\text{if } \left([A \mid P], s^T[A \mid P]\right) & \approx \left([A \mid P], u\right) \\
\text{then } \left([A \mid P], s^TA, A^{-1}(P)\right) & \approx \left([A \mid P], u, A^{-1}(P)\right)
\end{align*}
\]

\[
\begin{align*}
&\text{Challenge: Need to simulate keys } k_1 \text{ and } k_2 \text{ without trapdoors for } A_1 \text{ or } A_2 \\
&\text{Strategy: Argue ciphertext is pseudorandom} \\
&\text{(by LWE) if none of the keys satisfy the policy}
\end{align*}
\]

\[
\begin{align*}
\text{public keys} & \quad A_1, B_1, p_1 \\
& \quad A_2, B_2, p_2 \\
\text{challenge ciphertext} & \quad s_1^TA_1 \quad s_1^TB_1 + s_2^T B_2 \\
& \quad s_2^TA_2 \quad s_1^Tp_1 + s_2^Tp_2 + \mu \cdot [q/2] \\
\text{secret key} & \quad r_1 \leftarrow H(gid_1) \\
& \quad k_1 \leftarrow A_1^{-1}(p_1 + B_1 r_1) \\
& \quad r_2 \leftarrow H(gid_2) \\
& \quad k_2 \leftarrow A_2^{-1}(p_2 + B_2 r_2)
\end{align*}
\]
Security Analysis

Evasive LWE [Wee22, Tsa22]:

If \( ([A \mid P], s^T [A \mid P]) \approx ([A \mid P], u) \)

then \( ([A \mid P], s^T A, A^{-1}(P)) \approx ([A \mid P], u, A^{-1}(P)) \)

public keys

\( A_1, B_1, p_1 \)

\( A_2, B_2, p_2 \)

Authority 1

Authority 2

challenge ciphertext

\[
\begin{align*}
& s_1^T A_1 \\
& s_1^T B_1 + s_2^T B_2 \\
& s_2^T A_2 \\
& s_1^T p_1 + s_2^T p_2 + \mu \cdot [q/2]
\end{align*}
\]

Strategy: Argue ciphertext is pseudorandom (by LWE) if none of the keys satisfy the policy.

Challenge: Need to simulate keys \( k_1 \) and \( k_2 \) without trapdoors for \( A_1 \) or \( A_2 \).

secret key

\( r_1 \leftarrow H(gid_1) \)

\( k_1 \leftarrow A_1^{-1}(p_1 + B_1 r_1) \)

\( r_2 \leftarrow H(gid_2) \)

\( k_2 \leftarrow A_2^{-1}(p_2 + B_2 r_2) \)

Show: \( s_1^T (p_1 + B_1 r_1) \) is pseudorandom when \( r_1 \leftarrow H(gid_1) \).
Security Analysis

Evasive LWE [Wee22, Tsa22]:

if \([A \mid P], s^T A \] \approx ([A \mid P], u)

then \([A \mid P], s^T A, A^{-1}(P) \] \approx ([A \mid P], u, A^{-1}(P))

\[
\begin{align*}
 s_1^T A_1 & \quad s_1^T B_1 + s_2^T B_2 \\
 s_2^T A_2 & \quad s_1^T p_1 + s_2^T p_2 + \mu \cdot [q/2]
\end{align*}
\]

Strategy: Argue ciphertext is pseudorandom (by LWE) if none of the keys satisfy the policy

Challenge: Need to simulate keys \(k_1\) and \(k_2\) without trapdoors for \(A_1\) or \(A_2\)

Show: \(s_1^T(p_1 + B_1 r_1)\) is pseudorandom when \(r_1 \leftarrow H(\text{gid}_1)\)

How to design the hash function \(H\)?
Show: $s_1^T(p_1 + B_1r_1)$ is pseudorandom when $r_1 \leftarrow H(gid_1)$

(and given some additional components that depend on $s_1^T p_1$ and $s_1^T B_1$)

Main idea: for an input $x \in \{0,1\}^\ell$, define $H(x) = \left(\prod_{i \in [\ell]} D_{x_i}\right) e_1$

where $D_0, D_1$ are public short matrices and $e_1$ is the first basis vector

subset product of short matrices
Security Analysis

Show: \( s_1^T(p_1 + B_1 r_1) \) is pseudorandom when \( r_1 \leftarrow H(\text{gid}_1) \)
(and given some additional components that depend on \( s_1^T p_1 \) and \( s_1^T B_1 \))

Main idea: for an input \( x \in \{0,1\}^\ell \), define 
\[
    H(x) = \left( \prod_{i \in [\ell]} D_{x_i} \right) e_1
\]
where \( D_0, D_1 \) are public short matrices and \( e_1 \) is the first basis vector

subset product of short matrices

[BLMR13]: \( F_{D_0,D_1}(s, x) := s^T \prod_{i \in [\ell]} D_{x_i} \) is a pseudorandom function
Security Analysis

Show: \( s_1^T(p_1 + B_1 r_1) \) is pseudorandom when \( r_1 \leftarrow H(\text{gid}_1) \)
(and given some additional components that depend on \( s_1^T p_1 \) and \( s_1^T B_1 \))

Main idea: for an input \( x \in \{0,1\}^\ell \), define \( H(x) = \left( \prod_{i \in [\ell]} D_{x_i} \right) e_1 \)
where \( D_0, D_1 \) are public short matrices and \( e_1 \) is the first basis vector

subset product of short matrices

[BLMR13]: \( F_{D_0,D_1} (s, x) := s^T \prod_{i \in [\ell]} D_{x_i} \) is a pseudorandom function

Evasive LWE precondition (essentially) follows via [BLMR13]

see paper for full details
Multi-authority ABE for conjunctions based on [DKW21a] is secure assuming either

- LWE (with polynomial modulus-to-noise ratio) if $H$ is modeled as a random oracle; or
Multi-authority ABE for conjunctions based on [DKW21a] is secure assuming either

1. LWE (with polynomial modulus-to-noise ratio) if $H$ is modeled as a random oracle; or
2. evasive LWE if $H$ is a subset-product of short matrices

### public keys

<table>
<thead>
<tr>
<th>Authority 1</th>
<th>Authority 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_1, B_1, p_1$</td>
<td>$A_2, B_2, p_2$</td>
</tr>
</tbody>
</table>

### ciphertext

<table>
<thead>
<tr>
<th>$s_1^T A_1$</th>
<th>$s_1^T B_1 + s_2^T B_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_2^T A_2$</td>
<td>$s_1^T p_1 + s_2^T p_2 + \mu \cdot [q/2]$</td>
</tr>
</tbody>
</table>

### secret key

- $r \leftarrow H(gid)$
- $k_1 \leftarrow A_1^{-1}(p_1 + B_1 r)$
- $k_2 \leftarrow A_2^{-1}(p_2 + B_2 r)$
Multi-authority ABE for conjunctions based on [DKW21a] is secure assuming either

- LWE (with polynomial modulus-to-noise ratio) if \( H \) is modeled as a random oracle; or
- evasive LWE if \( H \) is a subset-product of short matrices

Not a “random looking” function!
Multi-authority ABE for conjunctions based on [DKW21a] is secure assuming either
- LWE (with polynomial modulus-to-noise ratio) if $H$ is modeled as a random oracle; or
- evasive LWE if $H$ is a subset-product of short matrices

Open problems:
- Multi-authority ABE from plain LWE
Multi-authority ABE for conjunctions based on \cite{DKW21a} is secure assuming either
\begin{itemize}
\item LWE (with polynomial modulus-to-noise ratio) if $H$ is modeled as a random oracle; or
\item evasive LWE if $H$ is a subset-product of short matrices
\end{itemize}

**Open problems:**
\begin{itemize}
\item Multi-authority ABE from *plain* LWE
\item Lattice-based multi-authority ABE beyond conjunctions
\end{itemize}
Multi-authority ABE for conjunctions based on [DKW21a] is secure assuming either
- LWE (with polynomial modulus-to-noise ratio) if $H$ is modeled as a random oracle; or
- evasive LWE if $H$ is a subset-product of short matrices

Open problems:
- Multi-authority ABE from plain LWE
- Lattice-based multi-authority ABE beyond conjunctions

https://eprint.iacr.org/2022/1194

Thank you!