Unbounded Inner Product Functional Encryption from Bilinear Maps
ASIACRYPT 2018

Junichi Tomida (NTT), Katsuyuki Takashima (Mitsubishi Electric)
Functional Encryption [O’Neill10, BSW11]

Alice

message \( x \)

Encoded Data

\( C_x \)

Bob

\( f(x) \)

Carol

\( g(x) \)

\( f \)

\( g \)

\( sk_f \)

\( sk_g \)

Alice’s public key \( pk \)

Bob’s secret key \( sk_f \)

Carol’s secret key \( sk_g \)

msk
Inner Product Functional Encryption (IPFE)[ABDP15]
Application of IPFE (DNA Sequence Similarity)

Encoding
- $A = (1,0,0,0)$
- $G = (0,1,0,0)$
- $C = (0,0,1,0)$
- $T = (0,0,0,1)$

Vector for $(A, T, T)$
- $x = (1,0,0,0,0,0,0,1,0,0,0,1)$

Vector for $(A, C, T)$
- $y = (1,0,0,0,0,0,1,0,0,0,0,1)$

$\langle x, y \rangle = 2$
Motivation of This Work

• All known IPFE schemes require to determine the vector length at the deployment phase.

• What kind of data will be handled? DNA sequences for the first chromosome? For humans? Or for all Species?

• Difficult to determine the length at the deployment phase.
  • Too large: efficiency loss
  • Too small: redeployment
Main Results of This Work

✓ **Unbounded IPFE for both private-key and public-key setting.**
  - Can handle any lengths of vectors for encryption and key generation.
  - Secure under the standard model and a standard assumption.
  - The former has function privacy. $\mathcal{E}_{sk_y}$ hides the information of $y$.

• **How to define the decryption?**
  - There are ciphertexts and secret keys of various lengths.
Main Results of This Work

✓ Allowing **partial decryption.**

• A ciphertext for indexed vector \( \mathbf{x} \in \mathbb{Z}^{S_c} \) can be decrypted with a secret key for indexed vector \( \mathbf{y} \in \mathbb{Z}^{S_k} \) iff \( S_c \supseteq S_k \).

• It seems inconvenient if decryption is possible only when the lengths of a ciphertext and a secret key are the same.

\[ C_x \text{ for indices (1,2,3,4)} \]

\[ \text{sk}_y \text{ for indices (2,3,4)} \]

\[ \text{sk}_z \text{ for indices (3,4,5)} \]
Why Partial Decryption?: $S_c \supseteq S_k$

Encrypted DNA sequence

1. ••ATTCTTTATCTGGGATC••ACTTCTGGATGCTTTTT••

2. ••ATTATCTTTATCTGAGATAT••

3. A••CTTGGATCCGTCAAAAT••

Alice

Bob

Alice

Bob

sk for part A

sk for part B
Roadmap

Function hiding technique [TAO16]
+PRF

Public-key partially decryptable IPFE

Entropy amplification [OT12]

Private-key unbounded IPFE

Public-key unbounded IPFE
Roadmap

Function hiding technique [TAO16] +PRF

Public-key partially decryptable IPFE

Entropy amplification [OT12]

Private-key unbounded IPFE

Public-key unbounded IPFE
Partially Decryptable IPFE

- Setup($1^\lambda, 1^m$) → pk, msk
- Enc(pk, x) → ct
- KeyGen(msk, y) → sk
- Dec(ct, sk) → d or ⊥

Normal IPFE

\[ x, y \in \mathbb{Z}^m \]
\[ d = \langle x, y \rangle \text{ over } \mathbb{Z} \]

Partially decryptable IPFE

\[ x \in \mathbb{Z}^{S_c}, y \in \mathbb{Z}^{S_k} \text{ for any } S_c, S_k \subseteq [m] = \{1, \ldots, m\} \]
\[ d = \sum_{i \in S_k} x_i y_i \text{ over } \mathbb{Z} \text{ iff } S_c \supseteq S_k. \]
Extension of Function Class

Normal IPFE

\[ f_y : \mathbb{Z}^m \rightarrow \mathbb{Z} \text{ s.t. } f_y(x) = \langle x, y \rangle \]

Partially decryptable IPFE

\[ f_{y,s_k} : \bigcup_{S \subseteq [m]} \mathbb{Z}^S \rightarrow \mathbb{Z} \text{ s.t. } f_{y,s_k}(x,S_c) = \begin{cases} \sum_{i \in S_k} x_i y_i & \text{if } S_c \supseteq S_k \\ \bot & \text{otherwise} \end{cases} \]
Security of Partially Decryptable IPFE

\[ b \leftarrow \{0,1\} \]

\[ sk = \text{KeyGen}(\text{msk}, y_i) \]

\[ (x_0, x_1, S_c) \]

\[ C_b = \text{Enc}(pk, x_b) \]

\[ b' \]

Condition

For \( x_0, x_1 \in \mathbb{Z}_S \), \( y_i \in \mathbb{Z}_{S_{ki}} \),

\[ \forall i, \sum_{j \in S_{ki}} x_0, j y_j = \sum_{j \in S_{ki}} x_1, j y_j \text{ when } S_c \supseteq S_{ki}. \]

Colluded illegitimate keys (i.e. \( S_c \nsubseteq S_{ki} \)) are useless for decryption.
Construction of Partially Decryptable IPFE

• Normal IPFE scheme (not secure).
  • Setup → pk = $[\mathbf{B}]_1$, msk = $\mathbf{B}^* = (\mathbf{B}^\top)^{-1}$, where $\mathbf{B} \leftarrow \mathbb{Z}_p^{m \times m}$
  • Enc(pk, x ) → $[\mathbf{xB}]_1$
  • KeyGen(msk, y ) → $[\mathbf{yB}^*]_2$
  • Dec(ct, sk) → $e([\mathbf{xB}]_1, [\mathbf{yB}^*]_2) = [\langle \mathbf{x}, \mathbf{y} \rangle]_T$

• $[\mathbf{M}]_i = g_i^\mathbf{M} = \left( g_i^{m_j,\ell} \right)_{j \in [m], \ell \in [n]}$
Construction of Partially Decryptable IPFE

Setup($1^m$) \begin{align*}
(pk_1, msk_1) & \quad (pk_2, msk_2) & \quad \cdots & \quad (pk_m, msk_m) \end{align*}

Normal IPFE scheme for 2 dimensions

Enc($x, S_c$) \begin{align*}
(x_1, z) & \quad (x_2, z) & \quad \cdots & \quad (x_m, z) \end{align*}

For $i \in [S_c]$

KeyGen($y, S_k$) \begin{align*}
(y_1, r_i) & \quad \cdots & \quad (y_m, r_m) \end{align*}

For $i \in [S_k]$, $\sum_{i \in S_k} r_i = 0$

Dec \[d = \sum_{i \in S_k} \text{Dec}(ct_i, sk_i) = \sum_{i \in S_k} x_i y_i + zr_i = \sum_{i \in S_k} x_i y_i\]

This part seems pseudorandom by SXDH when $S_c \not\subseteq S_k$, preventing collusion attack. But what the adversary obtains are ct and sk...
## Comparison with Bounded Schemes

### (Function-Hiding) Private-Key IPFE

| Reference | |msk| |ct| |sk| Pairing | Assumption |
|-----------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| DDM16 | $(8m^2 + 12m + 28) \mathbb{Z}_p$ | $(4m + 8) \mathbb{G}_1$ | $(4m + 8) \mathbb{G}_2$ | Yes | SXDH |
| TAO16 | $(4m^2 + 18m + 20) \mathbb{Z}_p$ | $(2m + 5) \mathbb{G}_1$ | $(2m + 5) \mathbb{G}_2$ | Yes | XDLIN |
| KKS17 | $(6m + 8) \mathbb{Z}_p$ | $(2m + 8) \mathbb{G}_1$ | $(2m + 8) \mathbb{G}_2$ | Yes | SXDH |
| Ours | | | | | | | | |

### Public-Key IPFE

| Reference | |pk| |msk| |ct| |sk| Pairing | Assumption |
|-----------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| ALS16 | $(m + 1) \mathbb{G}$ | $(m + 1) \mathbb{Z}_p$ | $(2m + 1) \mathbb{G}$ | $2 \mathbb{Z}_p$ | No | DDH |
| Ours | $28 \mathbb{G}_1$ | $28 \mathbb{Z}_p$ | $7m \mathbb{G}_1$ | $7m \mathbb{G}_2$ | Yes | SXDH |

$m$: vector length
Remark

• In our paper, we **directly** construct unbounded IPFE schemes essentially utilizing properties of bilinear groups.
• The notion, “Partial decryptable IPFE”, does not appear in our paper.
Conclusion

• Construction of unbounded IPFE schemes for both public and private-key setting.
  • Secure under the SXDH assumption in the standard model.
  • Almost the same efficiency as previous private-key IPFE schemes.

• It provides flexible decryption (partial decryption).
  • Decryptable iff $S_c \supseteq S_k$. 
Thank you for your attention