Traitor-Tracing from LWE Made Simple and Attribute-Based

Yilei Chen
Vinod Vaikuntanathan
Brent Waters
Hoeteck Wee
Daniel Wichs

TCC 2018
@Goa, India
Traitor-Tracing from **LWE** Made Simple and Attribute-Based

Yilei Chen  
Vinod Vaikuntanathan  
Brent Waters  
Hoeteck Wee  
Daniel Wichs

TCC 2018  
@Goa, India
Before ->

<- After
Private (or Constraint-hiding) Constrained PRF

[Boneh, Kim, Montgomery 17], [Canetti, Chen 17], [Brakerski, Tsabary, Vaikuntanathan, Wee 17]
**Lockable** obfuscation
[Goyal, Koppula, Waters 17]

a.k.a.
Compute and compare obfuscation
[ Wichs, Zirdelis 17 ]
Traitor tracing with polylog(n) blow-up [Goyal, Koppula, Waters 18]

[GKW18, 1:1] In a traitor tracing system [Chor, Fiat, Naor 94], an authority runs a setup algorithm that takes in $n$, the number of users in the system.

[GKW18, 1:2] The setup outputs a public key $pk$, master secret key $msk$, and $n$ secret keys $sk_1, sk_2, \ldots, sk_n$

[GKW18, 1:3] $CT = Enc(mpk, msg)$ that is decryptable by any of the $n$ secret keys. The msg is hidden for outsiders.

[GKW18, 1:4] Suppose some subset of users collude to create a decoding box $D$ which is capable of decrypting ciphertexts with some non-negligible probability.

[GKW18, 1:5] Trace: given the $msk$ and oracle access to $D$, outputs a set of users $T$ which contains at least 1 user from the colluding set $S$. 
[Goyal, Koppula, Waters 18]

Traitor Tracing
[Goyal, Koppula, Waters 18]
(135 pages)

> Traitor tracing from … … … … LWE
[Goyal, Koppula, Waters 18] (135 pages)
[Goyal, Koppula, Waters 18] (135 pages)

> Traitor tracing from “private linear broadcast encryption” (PLBE)
[Goyal, Koppula, Waters 18] (135 pages)

> Traitor tracing from “private linear broadcast encryption” (PLBE)

> PLBE
  = Attribute-based encryption (ABE)
  + Mixed Functional Encryption
[Goyal, Koppula, Waters 18] (135 pages)

> Traitor tracing from “private linear broadcast encryption” (PLBE)

> PLBE
= Attribute-based encryption (ABE)
+ Mixed Functional Encryption

> ABE from LWE is known since
[Gorbunov, Vaikuntanathan, Wee 13]
[Goyal, Koppula, Waters 18] (135 pages)

> Traitor tracing from “private linear broadcast encryption” (PLBE)

> PLBE
  = Attribute-based encryption (ABE)
  + Mixed Functional Encryption

> ABE from LWE is known since [Gorbunov, Vaikuntanathan, Wee 13]

> Mixed FE from LWE (70 pages)
[Goyal, Koppula, Waters 18]

This work: Mixed-FE from LWE made simpler.
[Goyal, Koppula, Waters 18]

This work:
Mixed-FE from LWE made simpler.

> Mixed FE from Lockable obfuscation and any collusion resistant SKFE (2 pages)
[Goyal, Koppula, Waters 18]

This work:
  Mixed-FE from LWE made simpler.

> Mixed FE from Lockable obfuscation and any collusion resistant SKFE (2 pages)

> Mixed FE from Key-homomorphic private constrained PRFs (7 pages)
The rest of the talk:

1. Mixed-FE definition, why is it useful in the Traitor tracing?

2. The two simpler constructions of Mixed FE from LWE.
Mixed-FE Definition [GKW18]
Mixed-FE = Secret-Key FE that
(1) swap the F and M;
(2) + 1-side public key mode
Mixed-FE Definition [GKW18]

Setup -> pp, msk;                      Enc(pp, msk, F) -> ct(F)
SkGen(pp, msk, m) -> sk(m);            Dec(pp, sk(m), ct(F)) -> F(m)

“Secret-Key FE that (1) swap the F and M; (2) + 1-side public key mode”
Mixed-FE Definition [GKW18]

Setup -> pp, msk; Enc(pp, msk, F) -> ct(F)
SkGen(pp, msk, m) -> sk(m); Dec(pp, sk(m), ct(F)) -> F(m)

T - Bounded collusion SKFE: (T=2 suffices for Traitor-tracing)

*The adversary can make T ct queries and poly sk queries, cannot distinguish between the real and simulated ciphertexts.*

“Secret-Key FE that (1) swap the F and M; (2) + 1-side public key mode”
Setup -> pp, msk; Enc(pp, msk, F) -> ct(F)
SkGen(pp, msk, m) -> sk(m); Dec(pp, sk(m), ct(F)) -> F(m)

T - Bounded collusion SKFE: (T=2 suffices for Traitor-tracing)

The adversary can make T ct queries and poly sk queries, cannot distinguish between the real and simulated ciphertexts.

1-side public key mode: Enc(pp) -> ct(“always accept”) CTs of “always accept functions” can be sampled publically.

“Secret-Key FE that (1) swap the F and M; (2) + 1-side public key mode”
[GKW 18] Why mixed-FE helps in building traitor tracing with polylog(n) overhead?

Private linear broadcast encryption:
- Secret-key associated to an index
- Normal encryption: everyone can dec
- TraceEnc mode: encrypt with an attribute $x > t$

Security requires:
- Broadcast enc. type of compactness
- ABE type of attribute
  + Attribute-hiding

Solution: ABE( $m$, mixedFE.Enc($x$) )
“Any T-bounded SKFE + Lockable obfuscation”
=> Use lockable obfuscation for the public mode
“Any T-bounded SKFE + Lockable obfuscation” => Use LO for the public mode

The concrete construction: left as an exercise for the audiences :)}
“Any T-bounded SKFE + Lockable obfuscation” => Use LO for the public mode

The concrete construction: left as an exercise for the audiences :)

Reviewers’ comments:

“(The first part) is really trivial … but this is not a bad thing.”

“I find the mixed FE construction from symmetric-key FE and lockable obfuscation cute and simple… Knowing the result of [GKW18] and reading this submitted work is such a joy!”
“Private constrained PRF + key-homomorphism”
=> bounded collusion mixed-FE
“Private constrained PRF + key-homomorphism” => bounded collusion mixed-FE

Gen -> msk for the PRF; Eval(msk, x) = F[msk](x)
Constrain(msk, C) -> ck[C]; Constrain.Eval(ck[C], x) -> ck[C](x)
Functionality: On x s.t. C(x) = 0,
    F[msk](x) = ck[C](x)
Rand: Given ck[C], F[msk](x) looks random on x s.t. C(x) = 1
Constraint-hiding: ck[C] hides C

1-constraint security from LWE: [BKM 17], [CC 17], [BTVW 17]
[CC 17]: 2-key secure PCPRF implies obfuscation.

This work: using key-homomorphic 1-CK PCPRF to get
T-bounded mixed-FE.

Private Constrained PRF
[Boneh, Lewi, Wu 17]
A 1-CT mixed-FE from PCPRF is implicit in [BKM 17], [CC 17]:
A 1-CT mixed-FE from PCPRF is implicit in [BKM 17], [CC 17]:

\[ \text{Setup}(\lambda): \quad \text{PRF.ms}k \]
A 1-CT mixed-FE from PCPRF is implicit in [BKM 17], [CC 17]:

**Setup(\(\lambda\)):**  PRF.msk

**SkGen(pp, msk, x):**  \(sk[x] = x, \quad F[msk](x)\)
A 1-CT mixed-FE from PCPRF is implicit in [BKM 17], [CC 17]:

\textbf{Setup}(\lambda): \quad \text{PRF.msk}

\textbf{SkGen}(pp, msk, x): \quad sk[x] = x, \quad F[msk](x)

\textbf{Enc}(pp, msk, F): \quad CT[F] = \text{CK}[F]
T-bounded Mixed-FE from LWE - Part II (example: $T = 1$)

A 1-CT mixed-FE from PCPRF is implicit in [BKM 17], [CC 17]:

**Setup($\lambda$):**

```
PRF.msk
```

**SkGen($pp$, $msk$, $x$):**

```
sk[x] = x,  F[msk](x)
```

**Enc($pp$, $msk$, $F$):**

```
CT[F] = CK[F]
```

**Enc($pp$):**

```
Simulated CK
```
A 1-CT mixed-FE from PCPRF is implicit in [BKM 17], [CC 17]:

**Setup(λ):**  \[\text{PRF.msk}\]

**SkGen(pp, msk, x):**  \[\text{sk}[x] = x, \text{F}[msk](x)\]

**Enc(pp, msk, F):**  \[\text{CT}[F] = \text{CK}[F]\]

**Enc(pp):**  \[\text{Simulated CK}\]

**Dec( CT[F], sk[x]) :** Compare  \[\text{CK}[F](x)\] and  \[\text{F}[msk](x)\]
if close, output 0; if not, output 1.
Run the 1-CT mixed-FE in $2\lambda$ copies.
T-bounded Mixed-FE from LWE - Part II (example: T = 2)

Setup(\lambda):  
\begin{align*}
\text{PRF.msk}[1,0] & \quad \ldots, \quad \text{PRF.msk}[\lambda,0] \\
\text{PRF.msk}[1,1] & \quad \ldots, \quad \text{PRF.msk}[\lambda,1]
\end{align*}

SkGen(pp, msk, x):  
\begin{align*}
\text{sk}[x] = x, \\
\text{F}[ \text{msk}[1,0] ](x) & \quad \ldots, \quad \text{F}[ \text{msk}[\lambda,0] ](x) \\
\text{F}[ \text{msk}[1,1] ](x) & \quad \ldots, \quad \text{F}[ \text{msk}[\lambda,1] ](x)
\end{align*}
T-bounded Mixed-FE from LWE - Part II (example: $T = 2$)

**Setup($\lambda$):**

- $\text{PRF.msk}[1, 0]$
- $\text{PRF.msk}[1, 1]$
- $\text{PRF.msk}[\lambda, 0]$
- $\text{PRF.msk}[\lambda, 1]$

**SkGen(pp, msk, x):**

- $sk[x] = x$,
- $F[msk[1, 0]](x)$
- $F[msk[1, 1]](x)$
- $\dots$
- $F[msk[\lambda, 0]](x)$
- $F[msk[\lambda, 1]](x)$

**Enc(pp, msk, F):**

- $CT[F] = S \in \{0, 1\}^\lambda$
- $\text{CK}[F] = \text{Constrain}(F, msk[1, s[1]])$

Choose a random tag that selects $\lambda$ msks out of $2\lambda$. 
Choose a random tag that selects $\lambda$ msks out of $2\lambda$. 

\[ \text{CT}[F] = S \in \{0,1\}^\lambda, \quad \text{CK}[F] = \text{Constrain}(F, +\text{msk}[2,s[2]]) \]
T-bounded Mixed-FE from LWE - Part II (example: T = 2)

Setup(λ):
- PRF.msk[1,0]
- PRF.msk[λ,0]
- PRF.msk[1,1]
- PRF.msk[λ,1]

SkGen(pp, msk, x):
- sk[x] = x,
- F[ msk[1,0] ]( x )
- F[ msk[1,1] ]( x )
- F[ msk[λ,0] ]( x )
- F[ msk[λ,1] ]( x )

Enc(pp, msk, F):
- CT[F] = S ∈ \{0,1\}^\lambda,
- CK[ F ] = Constrain(F, +msk[λ,s[λ]] +msk[2,s[2]] +msk[λ,s[λ]] )

Choose a random tag that selects λ msks out of 2λ.
T-bounded Mixed-FE from LWE - Part II (example: $T = 2$)

**Setup($\lambda$):**

- $\text{PRF.msk}[1,0]$
- $\text{PRF.msk}[1,1]$
- $\ldots$, $\ldots$
- $\text{PRF.msk}[\lambda,0]$
- $\text{PRF.msk}[\lambda,1]$

**SkGen(pp, msk, x):** $\text{sk}[x] = x$, $\text{F}[\text{msk}[1,0]](x)$, $\text{F}[\text{msk}[1,1]](x)$, $\ldots$, $\text{F}[\text{msk}[\lambda,0]](x)$, $\text{F}[\text{msk}[\lambda,1]](x)$

**Enc(pp, msk, F):** $\text{CT}[F] = S \in \{0,1\}^\lambda$, $\text{CK}[F] = \text{Constrain}(F, +\text{msk}[\lambda,s[\lambda]])$

**Enc(pp):** $S \in \{0,1\}^\lambda$

Simulated CK
T-bounded Mixed-FE from LWE - Part II (example: \( T = 2 \))

**Setup(\( \lambda \))**:
- \( \text{PRF.msk}[1,0] \)
- \( \text{PRF.msk}[\lambda,0] \)
- \( \text{PRF.msk}[1,1] \)
- \( \ldots, \ldots \)
- \( \text{PRF.msk}[\lambda,1] \)

**SkGen(pp, msk, x)**:
- \( \text{sk}[x] = x \)
- \( F[ \text{msk}[1,0] ](x) \)
- \( F[ \text{msk}[1,1] ](x) \)
- \( \ldots, \ldots \)
- \( F[ \text{msk}[\lambda,0] ](x) \)
- \( F[ \text{msk}[\lambda,1] ](x) \)

**Enc(pp, msk, F)**:
- \( \text{CT}[F] = S \in \{0,1\}^\lambda \)
- \( \text{CK}[F] = \text{Constrain}(F, +\text{msk}[\lambda,s[\lambda]]) \)
- \( \text{Simulated CK} \)

**Enc(pp)**:
- \( S \in \{0,1\}^\lambda \)

**Dec( CT[F], sk[x] )**:
- Compare \( \text{CK}[F](x) \) and \( F[\text{msk}[1,s[1]]](x) \)
- If close, output 0; if not, output 1.
T-bounded Mixed-FE from LWE - Part II (example: $T = 2$)

**Setup($\lambda$):**
- $\text{PRF.msk}[1,0]$
- $\text{PRF.msk}[\lambda,0]$
- $\text{PRF.msk}[1,1]$
- $\text{PRF.msk}[\lambda,1]$

**SkGen($pp$, $msk$, $x$):**
- $\text{sk}[x] = x$
- $F[msk[1,0]](x)$
- $F[msk[1,1]](x)$
- $F[msk[\lambda,0]](x)$
- $F[msk[\lambda,1]](x)$

**Enc($pp$, $msk$, $F$):**
- $CT[F] = S \in \{0,1\}^{\lambda}$, $\text{CK}[F] = \text{Constrain}(F, +\text{msk}[\lambda,\text{s}[\lambda]])$

**Enc($pp$):**
- $S \in \{0,1\}^{\lambda}$
- Simulated CK

**Dec( $CT[F]$, $sk[x]$ ):** Compare $\text{CK}[F](x)$ and $+F[msk[2,s[2]]](x)$.
- If close, output 0; if not, output 1.
T-bounded Mixed-FE from LWE - Part II (example: $T = 2$)

Setup($\lambda$): $\text{PRF.msk}[1,0]$, $\text{PRF.msk}[1,1]$, ..., $\text{PRF.msk}[\lambda,0]$, $\text{PRF.msk}[\lambda,1]$

SkGen(pp, msk, x): $\text{sk}[x] = x$, $\text{F}[\text{msk}[1,0]](x)$, $\text{F}[\text{msk}[1,1]](x)$, ..., $\text{F}[\text{msk}[\lambda,0]](x)$, $\text{F}[\text{msk}[\lambda,1]](x)$

Enc(pp, msk, F): $\text{CT}[F] = S \in \{0,1\}^\lambda$, $\text{CK}[F] = \text{Constrain}(F, +\text{msk}[\lambda,s[\lambda]])$

Enc(pp): $S \in \{0,1\}^\lambda$, Simulated CK

Dec(CT[F], sk[x]): Compare $\text{CK}[F](x)$ and $+...+\text{F}[\text{msk}[\lambda,s[\lambda]]](x)$, and if close, output 0; if not, output 1.
T-bounded Mixed-FE from LWE - Part II (example: T = 2)

**Setup(λ):**

- \( \text{PRF.msk[1,0]} \)
- \( \text{PRF.msk[1,1]} \)
- \( \text{PRF.msk[λ,0]} \)
- \( \text{PRF.msk[λ,1]} \)

**SkGen(pp, msk, x):**

- \( \text{sk}[x] = x, \)
- \( \text{F[msk[1,0]](x)} \)
- \( \text{F[msk[1,1]](x)} \)
- \( \text{F[msk[λ,0]](x)} \)
- \( \text{F[msk[λ,1]](x)} \)

**Enc(pp, msk, F):**

- \( \text{CT[F]} = S \in \{0,1\}^λ \)
- \( \text{CK[F] = Constrain(F, +\text{msk[λ,s[λ]]}, +\text{msk[2,3]}, ...)} \)

**Enc(pp):**

- \( S \in \{0,1\}^λ \)
- \( \text{Simulated CK} \)

**Dec(CT[F], sk[x]):**

- \( \text{Compare CK[F](x)} \)
- \( \text{if close_output 0: if not_output 1} \)

**Correctness follows key-homomorphism.**
T-bounded Mixed-FE from LWE - Part II (example: $T = 2$)

**Setup($\lambda$):**
- $\text{PRF.msk}[1,0]$
- $\text{PRF.msk}[\lambda,0]$
- $\text{PRF.msk}[1,1]$
- $\text{PRF.msk}[\lambda,1]$

**SkGen($pp$, msk, x):**
- $sk[x] = x$
- $\text{F}[\text{msk}[1,0]](x)$
- $\text{F}[\text{msk}[1,1]](x)$
- $\text{F}[\text{msk}[\lambda,0]](x)$
- $\text{F}[\text{msk}[\lambda,1]](x)$

**Enc($pp$, msk, F):**
- $CT[F] = S \in \{0,1\}^\lambda$
- $\text{CK}[F] = \text{Constrain}(F, +\text{msk}[\lambda,s[\lambda]])$

**Enc($pp$):**
- $S \in \{0,1\}^\lambda$
- Simulated CK

**Dec($CT[F]$, sk[x]):**
- Compare $\text{CK}[F](x)$ and $\text{F}[\text{msk}[\lambda,s[\lambda]]](x)$
- $\text{F}[\text{msk}[\lambda,s[\lambda]]](x)$

Two tags have 1 bit difference with high probability. Can use that diff bit to simulate everything back using key-homomorphism.
Run the 1-CT mixed-FE in $2(T-1)\lambda$ copies, choose a random tag $s \in [2(T-1)]^\lambda$;

For the tag in each CT, the probability that the j-th entry doesn’t show up in the other tags is $>\frac{1}{2}$, so with probability $1 - 2^{-\lambda}$, there will be a free entry.

By a union bound, with probability $1 - T \cdot 2^{-\lambda}$, every tag has a distinct entry.
# Summary of T-bounded mixed-FE schemes from LWE

<table>
<thead>
<tr>
<th>Mixed-FE Construction</th>
<th>Function class</th>
<th>Blow up</th>
<th>Security</th>
<th>Pages</th>
<th>LWE Tech*</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ GKW 18 ]</td>
<td>BP</td>
<td>$O(T^2)$</td>
<td>selective</td>
<td>70</td>
<td>GGH15-zeta</td>
</tr>
<tr>
<td>LO + [ GVW 12 ]</td>
<td>Poly circuit</td>
<td>$O(T^4)$</td>
<td>adaptive</td>
<td>2</td>
<td>GGH15-beta</td>
</tr>
<tr>
<td>LO + [ Agrawal, Rosen 17 ]</td>
<td>Poly circuit</td>
<td>$O(T^2)$</td>
<td>selective</td>
<td>2</td>
<td>GGH15-beta</td>
</tr>
<tr>
<td>Key-homomorphic PCPRF</td>
<td>Poly circuit</td>
<td>$O(T)$</td>
<td>selective</td>
<td>7</td>
<td>GGH15/BGG+</td>
</tr>
</tbody>
</table>
Thanks for your time!

https://eprint.iacr.org/2018/897