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

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 sk1, sk2, . . . ,skn

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

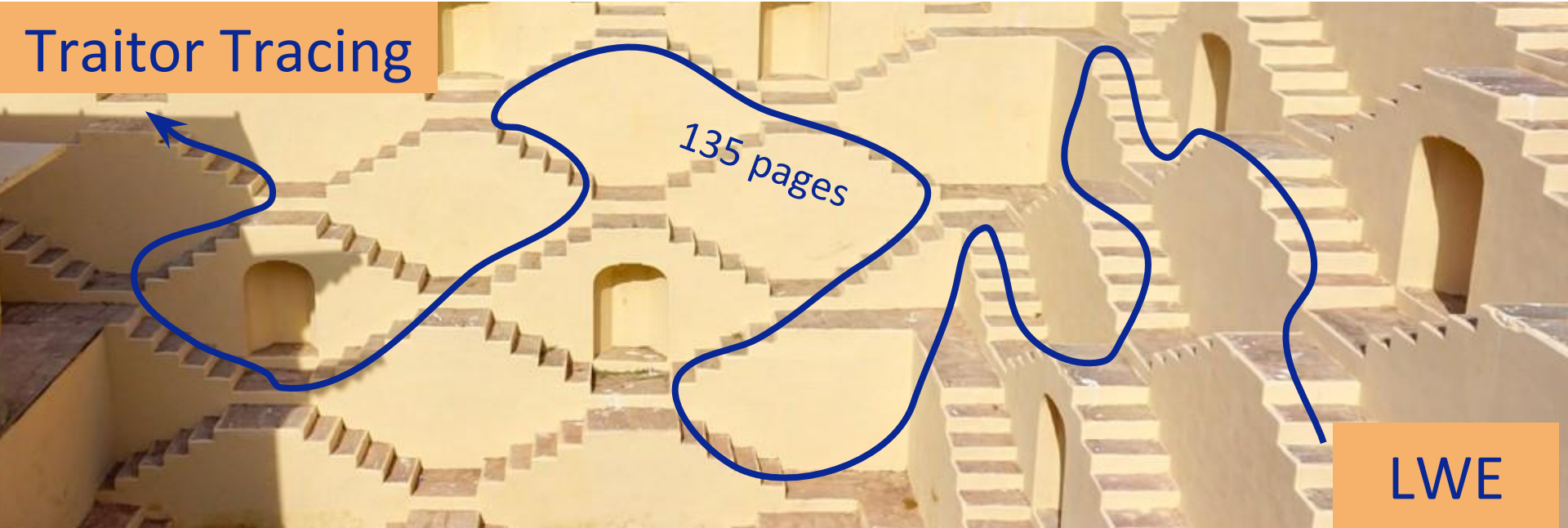
[GKW18, 1:4] Suppose some subset of users collude to create a decoding box D which is capable of decrypting ciphertexts with some non-negligible probability.

[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.

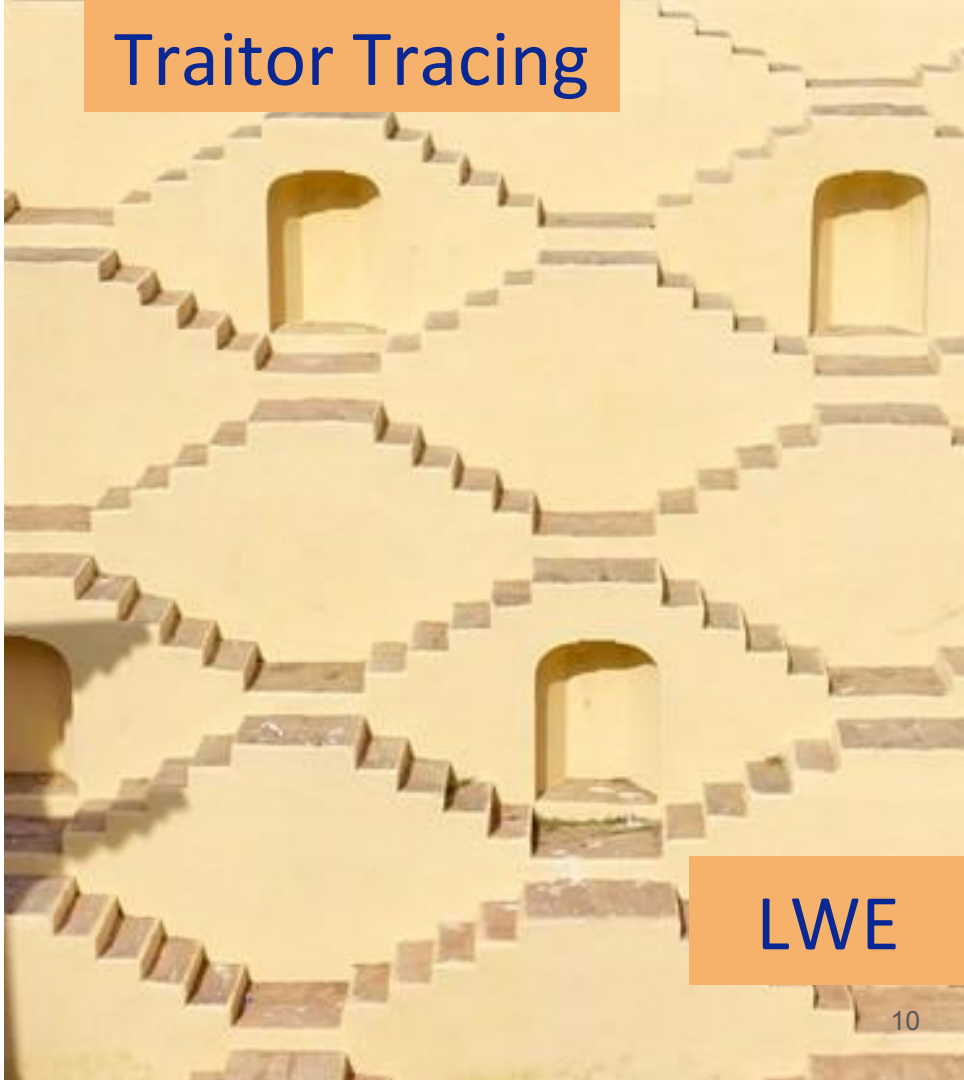
[Goyal, Koppula, Waters 18]

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

> Traitor tracing from … … … LWE



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



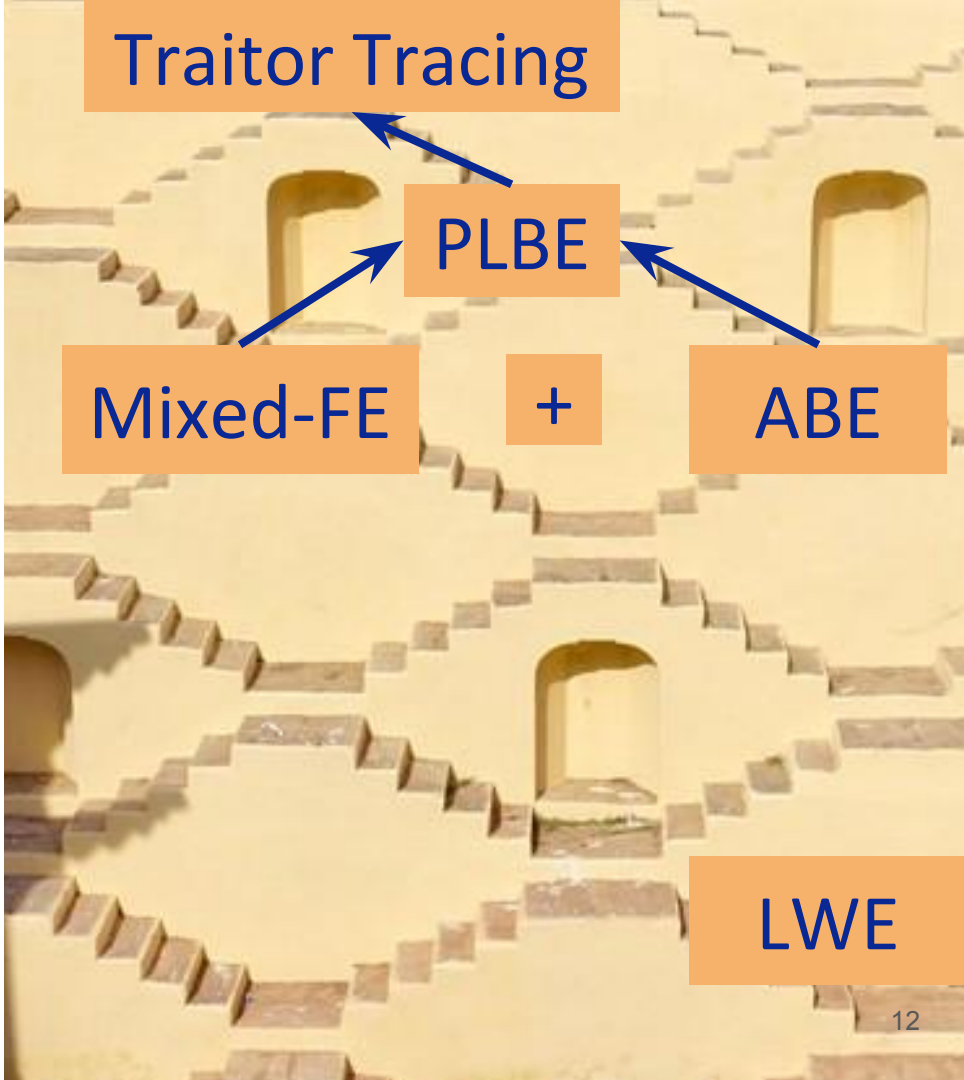
[Goyal, Koppula, Waters 18]
(135 pages)

> 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

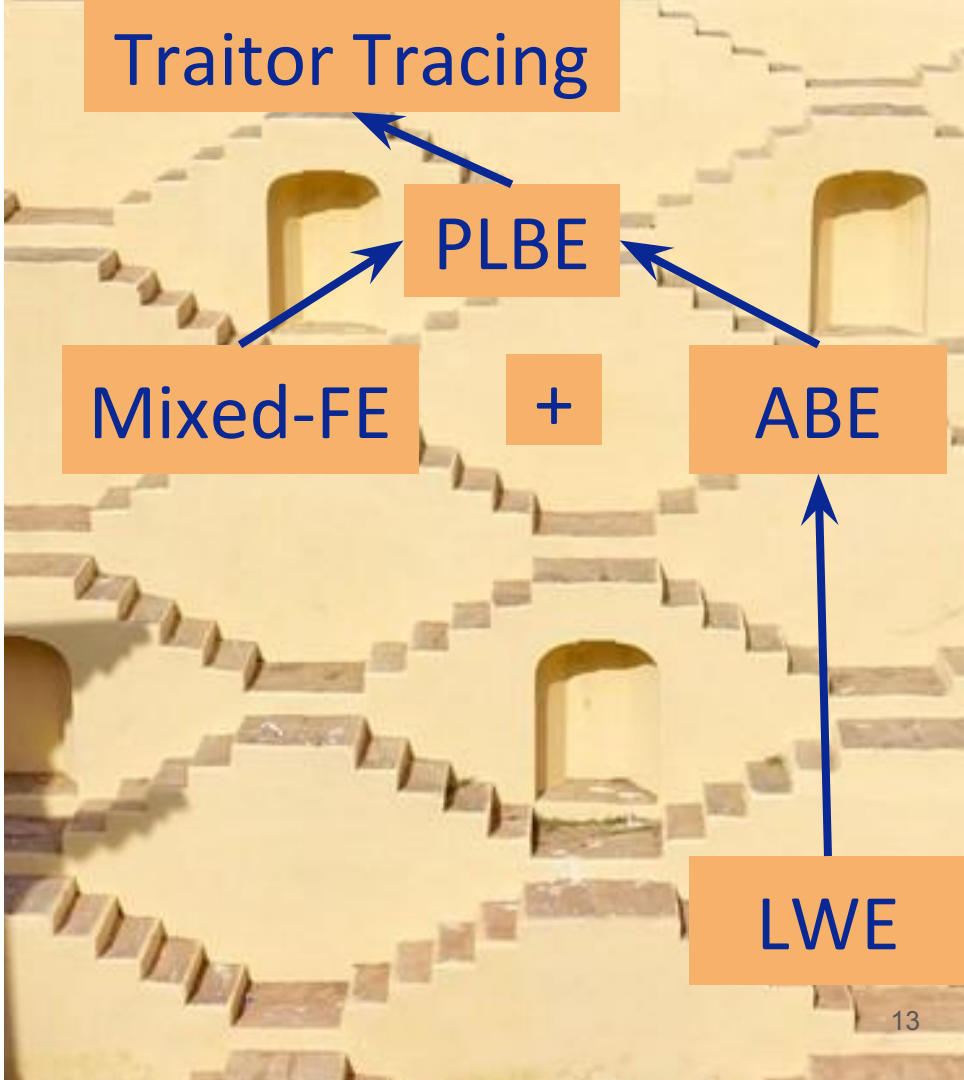


[Goyal, Koppula, Waters 18]
(135 pages)

> 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]



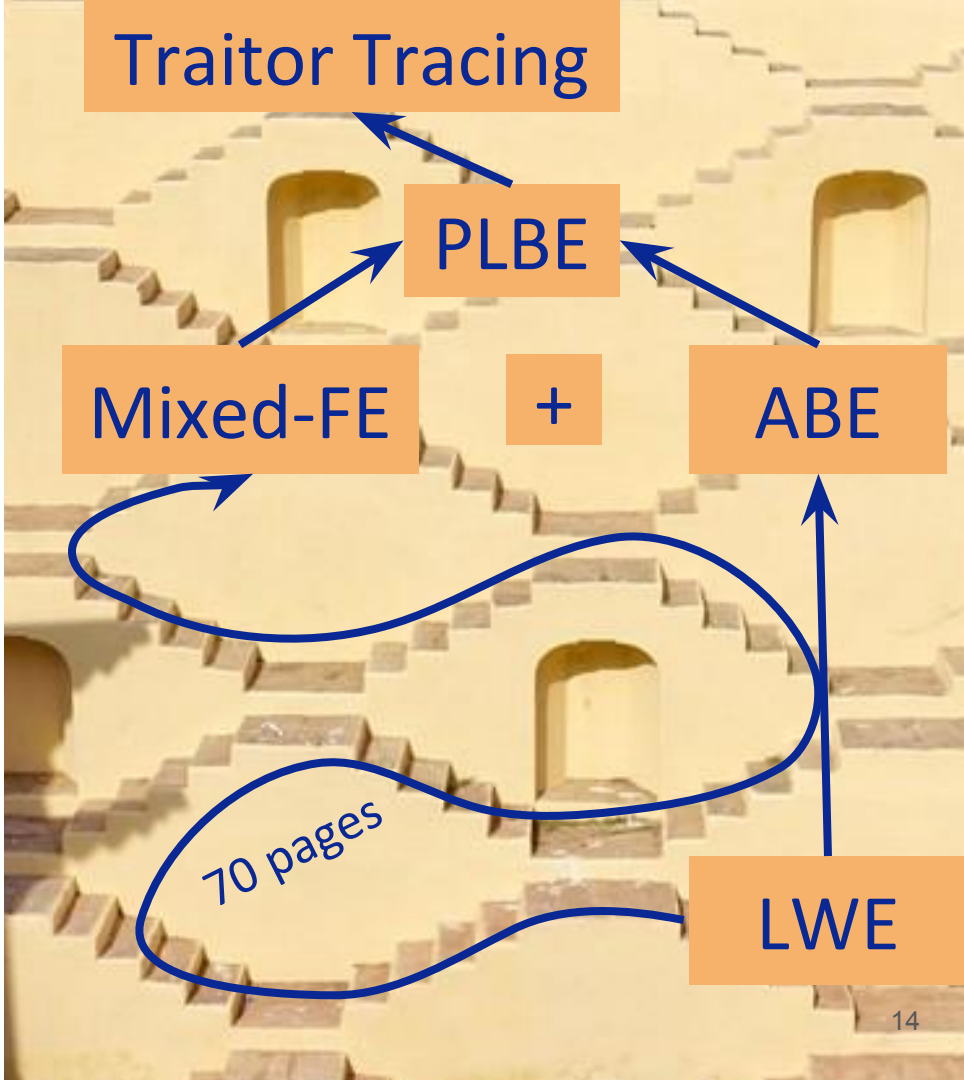
[Goyal, Koppula, Waters 18]
(135 pages)

> 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)

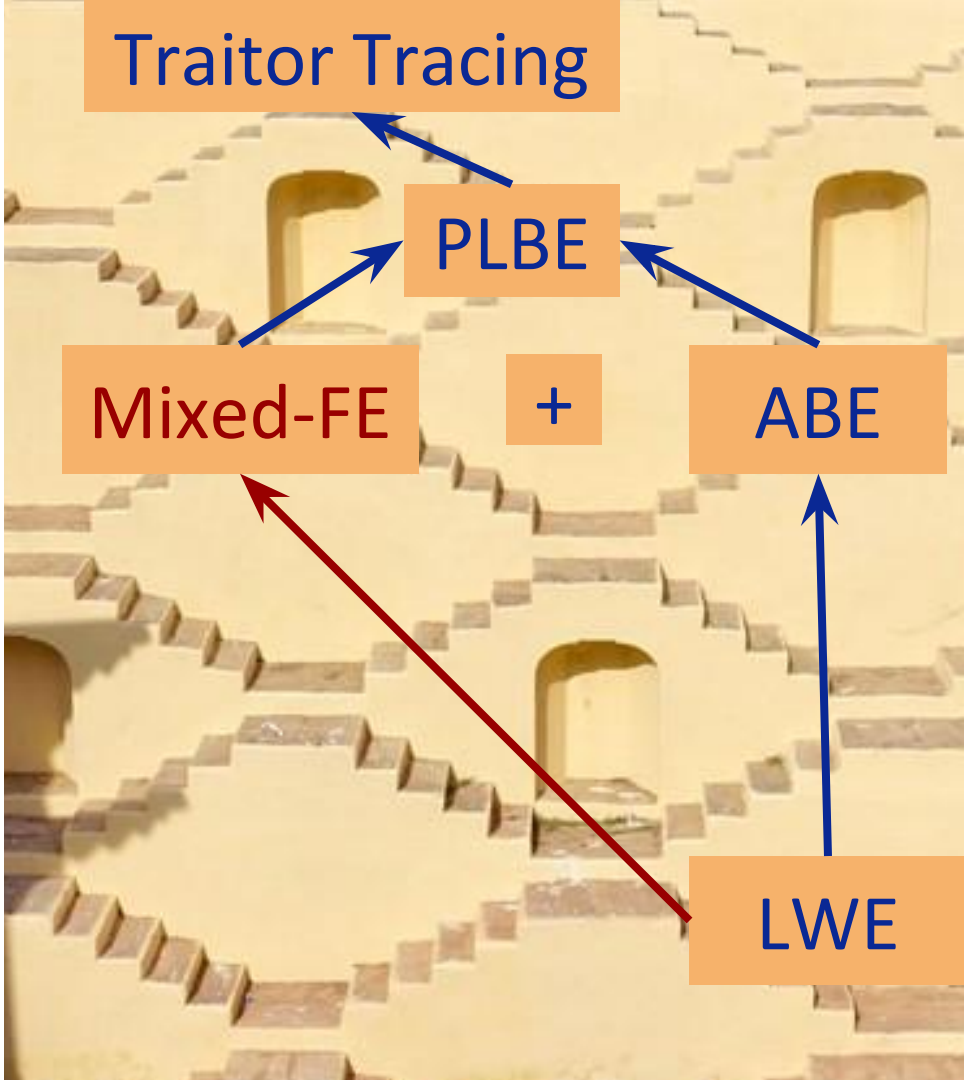


[Goyal, Koppula, Waters 18]

==========

This work:

Mixed-FE from LWE made simpler.



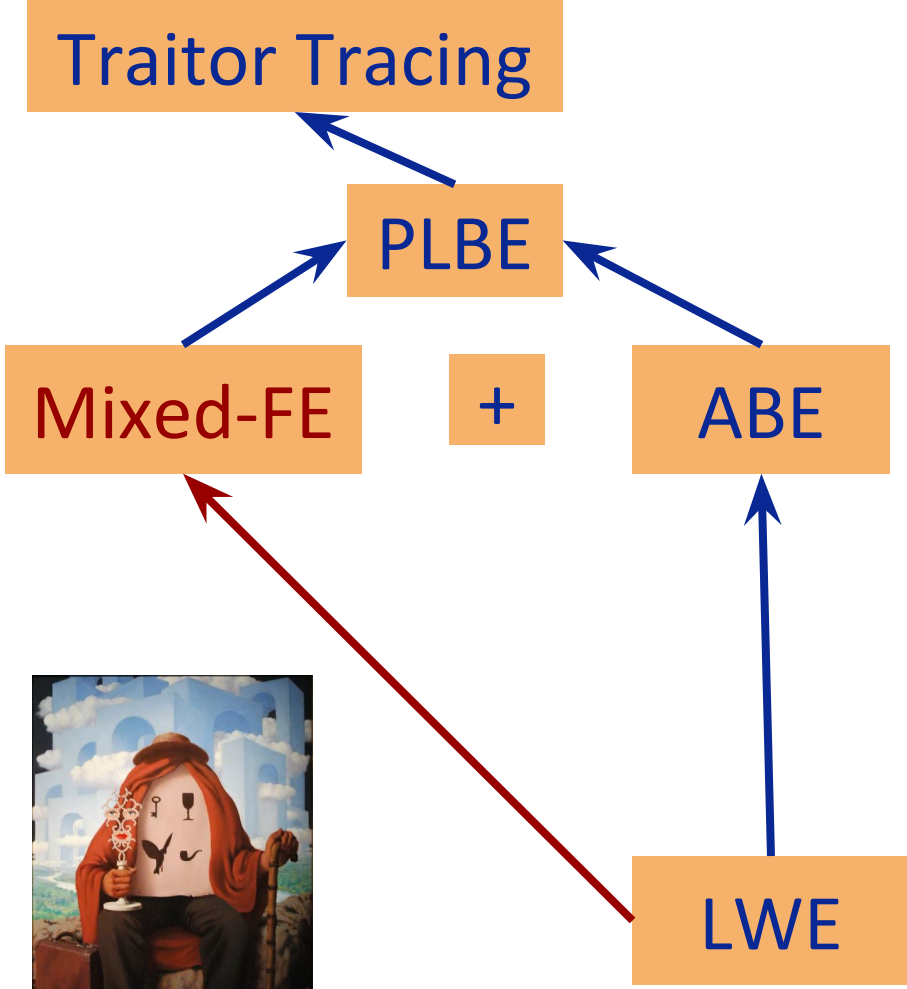
[Goyal, Koppula, Waters 18]

==========

This work:

Mixed-FE from LWE made simpler.

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



[Goyal, Koppula, Waters 18]

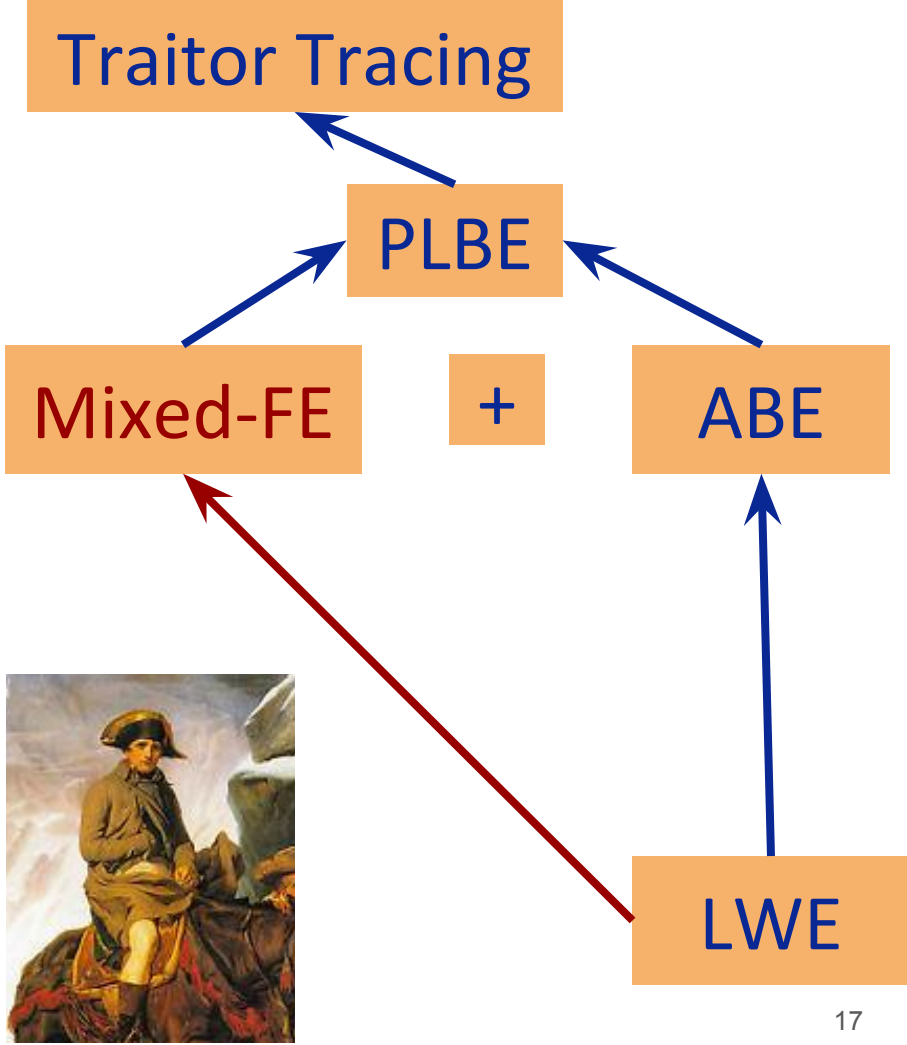
==========

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 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;

SkGen(pp, msk, m) -> sk(m);

Enc(pp, msk, F) -> ct(F)

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"

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.

1-side public key mode: Enc(pp) -> ct("always accept")

CTs of "always accept functions" can be sampled publically.

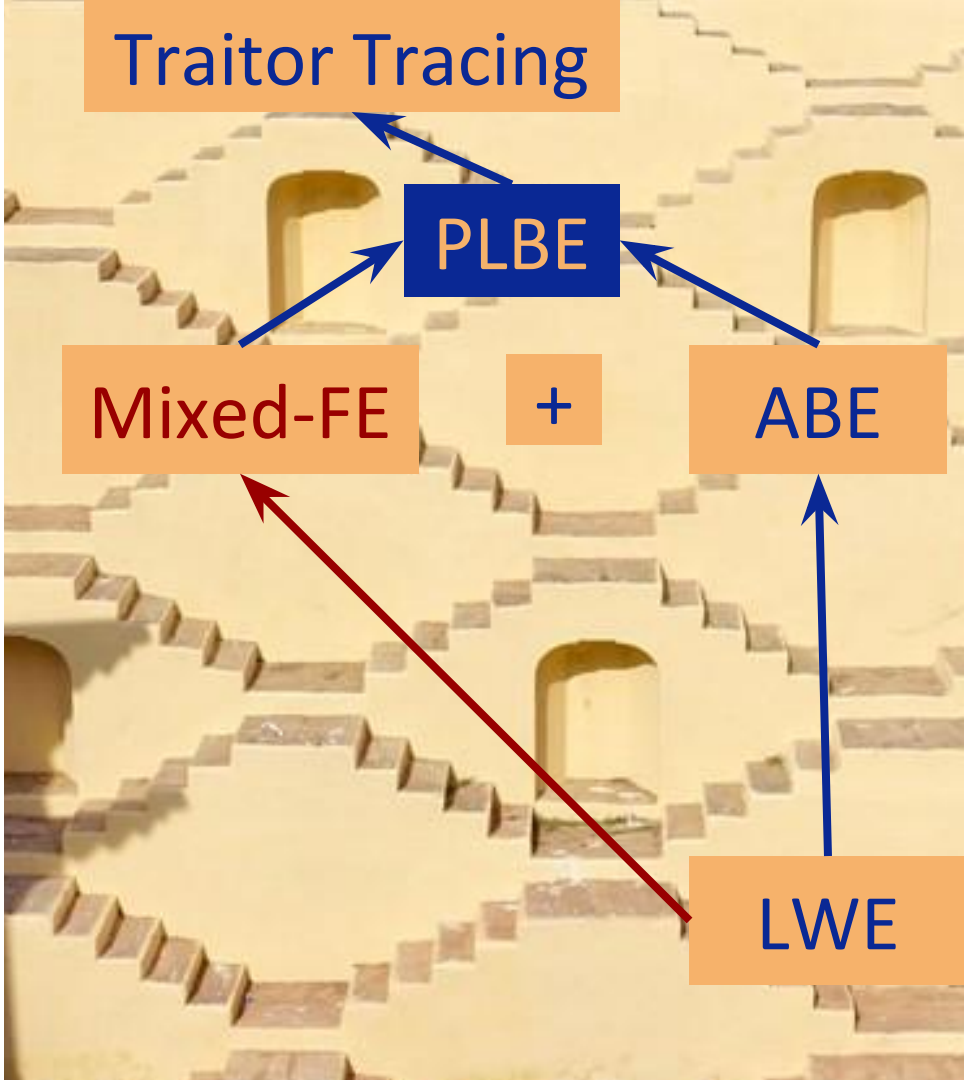
"Secret-Key FE that (1) swap the F and M; (2) + 1-side public key mode"

[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))



T-bounded Mixed-FE from LWE - Part I

“Any T-bounded SKFE + Lockable obfuscation”
=> Use lockable obfuscation for the public mode

T-bounded Mixed-FE from LWE - Part I

"Any T-bounded SKFE + Lockable obfuscation" => Use LO for the public mode

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

T-bounded Mixed-FE from LWE - Part I

"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! "

T-bounded Mixed-FE from LWE - Part II

“Private constrained PRF + key-homomorphism”
=> bounded collusion mixed-FE

T-bounded Mixed-FE from LWE - Part II

"Private constrained PRF + key-homomorphism" => bounded collusion mixed-FE

Private Constrained PRF
[Boneh, Lewi, Wu 17]

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.

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]:

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(λ): PRF.msk

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(λ): PRF.msk

SkGen(pp, msk, x): sk[x] = x, F[msk](x)

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(λ): PRF.msk

SkGen(pp, msk, x): sk[x] = x, F[msk](x)

Enc(pp, msk, F): CT[F] = 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(λ): PRF.msk

SkGen(pp, msk, x): sk[x] = x, F[msk](x)

Enc(pp, msk, F): CT[F] = CK[F]

Enc(pp): Simulated CK

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(λ): PRF.msk

SkGen(pp, msk, x): sk[x] = x, F[msk](x)

Enc(pp, msk, F): CT[F] = CK[F]

Enc(pp): Simulated CK

Dec(CT[F], sk[x])): Compare CK[F](x) and F[msk](x)

if close, output 0; if not, output 1.

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]

Run the 1-CT mixed-FE in 2λ copies.

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

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

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

Setup(λ): PRF.msk[1,0] PRF.msk[1,1] …, … PRF.msk[λ,0] 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] = $s \in \{0,1\}^\lambda$, CK[F] = Constrain(F, msk[1,s[1]])

Choose a random tag that selects λ msks out of 2λ.

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

Setup(λ): PRF.msk[1,0] PRF.msk[1,1] …, … PRF.msk[λ,0] 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] = $S \in \{0,1\}^{\lambda}$, CK[F] = Constrain(F, +msk[2,s[2]])

Choose a random tag that selects λ msks out of 2λ.

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

Setup(λ): PRF.msk[1,0] PRF.msk[1,1] …, … PRF.msk[λ,0] 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] = $S \in \{0,1\}^{\lambda}$, CK[F] = Constrain(F, +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(λ): PRF.msk[1,0] PRF.msk[1,1] …, … PRF.msk[λ,0] 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] = $S \in \{0,1\}^\lambda$, CK[F] = Constrain(F, +msk[λ,s[λ]] +msk[2,s[2]])

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

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

Setup(λ): PRF.msk[1,0] PRF.msk[1,1] …, … PRF.msk[λ,0] 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] = $S \in \{0,1\}^\lambda$, CK[F] = Constrain(F, +msk[λ,s[λ]])

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

Dec(CT[F], sk[x])): Compare CK[F](x) and F[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(λ): PRF.msk[1,0] PRF.msk[1,1] …, … PRF.msk[λ,0] 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] = $S \in \{0,1\}^\lambda$, CK[F] = Constrain(F, +msk[λ,s[λ]] +msk[2,s[2]])

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

Dec(CT[F], sk[x])): Compare 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(λ): PRF.msk[1,0] PRF.msk[1,1] …, … PRF.msk[λ,0] 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] = $S \in \{0,1\}^\lambda$, CK[F] = Constrain(F, +msk[λ,s[λ]])

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

Dec(CT[F], sk[x])): Compare CK[F](x) and +…+F[msk[λ,s[λ]]](x)

if close, output 0; if not, output 1.

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

Setup(λ): PRF.msk[1,0] PRF.msk[1,1] …, … PRF.msk[λ,0] 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] = $S \in \{0,1\}^\lambda$, CK[F] = Constrain(F, +msk[λ,s[λ]])

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

Dec(CT[F], sk[x])): Compare CK[F](x) and +…+F[msk[λ,s[λ]]](x)

if close, output 0; if not, output 1

Correctness follows key-homomorphism.

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

Setup(λ): PRF.msk[1,0] PRF.msk[1,1] …, … PRF.msk[λ,0] 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] = $S \in \{0,1\}^\lambda$, CK[F] = Constrain(F, +msk[λ,s[λ]])

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

Dec(CT[F], sk[x])): Compare CK[F](x) and +…+F[msk[λ,s[λ]]](x)

Two tags have 1 bit difference with high probability.
Can use that diff bit to simulate everything back using key-homomorphism.

T-bounded Mixed-FE from LWE - Part II (poly T)

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 >½, 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

Mixed-FE Construction	Function class	Blow up	Security	Pages	LWE Tech*
[GKW 18]	BP	$O(T^2)$	selective	70	GGH15-zeta
LO + [GVW 12]	Poly circuit	$O(T^4)$	adaptive	2	GGH15-beta
LO + [Agrawal, Rosen 17]	Poly circuit	$O(T^2)$	selective	2	GGH15-beta
Key-homomorphic PCPRF	Poly circuit	$O(T)$	selective	7	GGH15/BGG+

Thanks for your time!

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