Risky Traitor Tracing and New Differential Privacy Negative Results

Rishab Goyal Venkata Koppula
Andrew Russell Brent Waters

Hardness of Non-Interactive Differential Privacy from One-Way Functions

Lucas Kowalczyk Tal Malkin
Jonathan Ullman Daniel Wichs

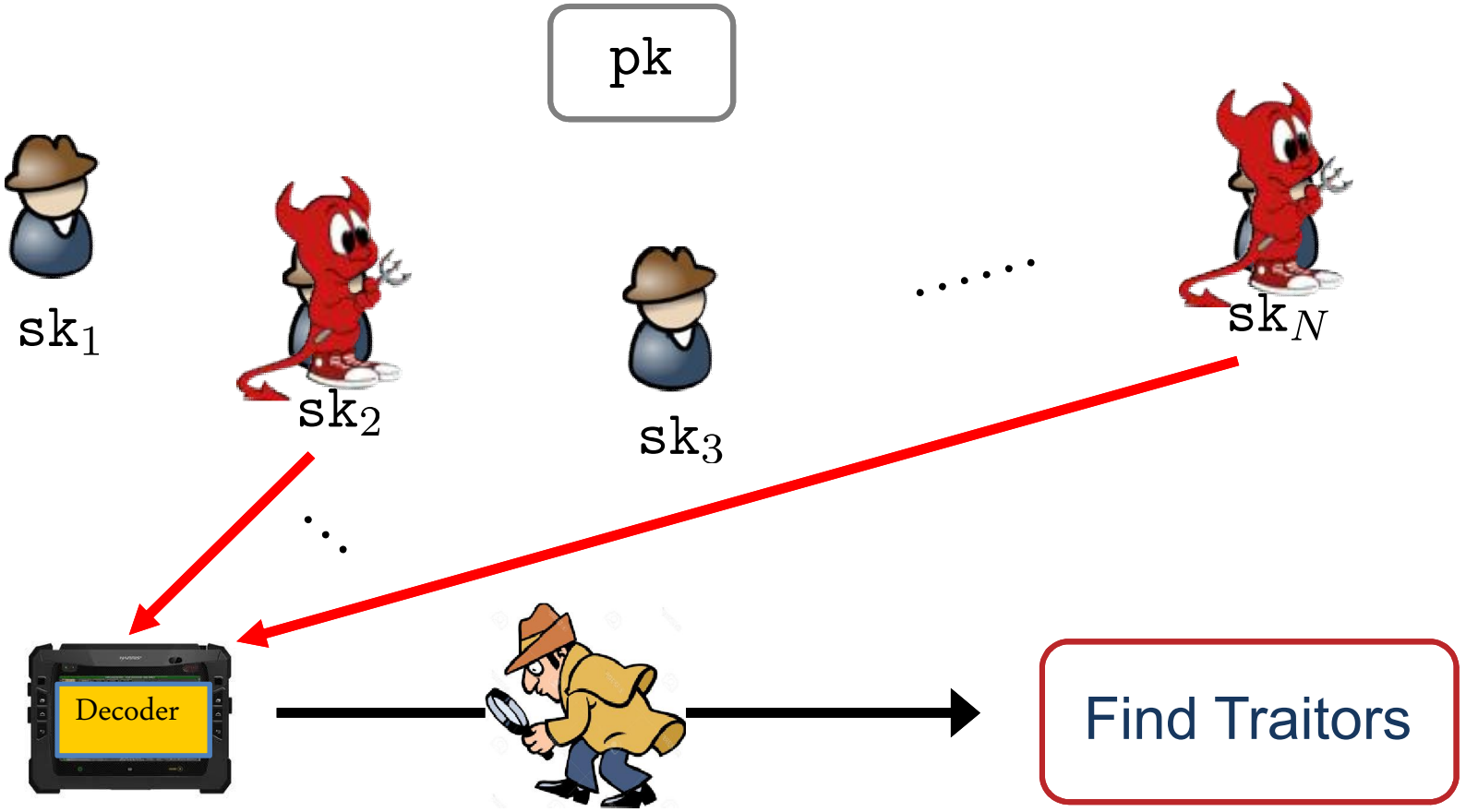
Risky Traitor Tracing and New Differential Privacy Negative Results

Rishab Goyal Venkata Koppula

Andrew Russell Brent Waters

THE UNIVERSITY OF TEXAS AT AUSTIN

Traitor Tracing [Chor-Fiat-Naor 94]



Key Challenges:
(1) Obfuscated Decoder
(2) Collusions

Tracing Algorithms

$\mathtt{Setup}(1^\lambda, 1^N) \to (\mathtt{msk}, \mathtt{pk}, (\mathrm{sk}_1, \ldots, \mathrm{sk}_N))$

$\mathtt{Enc}(\mathtt{pk}, m \in \mathcal{M}) \to \mathtt{ct}$

$\mathtt{Dec}(\mathtt{sk}, \mathtt{ct}) \to m \in \mathcal{M}$

$\mathtt{Trace}^D(\mathtt{msk}, 1^y, m_0, m_1) \to T \subseteq [1, N]$

$$\texttt{ExptTT}_{\mathcal{A},\epsilon}(\lambda)$$

$$1^N$$

$$\texttt{Setup}(1^\lambda, 1^N) \to (\texttt{msk}, \texttt{pk}, (\texttt{sk}_1, \ldots, \texttt{sk}_N))$$

$$\texttt{pk}$$

$$S \subseteq [1, N]$$

$$\texttt{sk}_i : i \in S$$

$$D, m_0, m_1$$

$$T \leftarrow \texttt{Trace}^D(\texttt{msk}, 1^{1/\epsilon(\lambda)}, m_0, m_1)$$

(1) No false trace: $\Pr[\exists i : i \in T, i \notin S] = \text{negl}(\lambda)$

(2) Catch: $\Pr[T \neq \emptyset] \geq \Pr[D \ \epsilon\text{-distinguishes} \ (m_0, m_1)]$

It's About the Ciphertext Size!

PKE: $N \cdot \mathrm{poly}(\lambda)$

Bilinear Maps [Boneh-Sahai-Waters06,Boneh-Waters06…]: $\sqrt{N} \cdot \mathrm{poly}(\lambda)$

Functional Encryption/iO[GGHRSW13,Boneh-Zhandry14]: $\mathrm{poly}(\lambda, \lg N) = \mathrm{poly}(\lambda)$

Can we get better efficiency from standard assumptions if we relax tracing requirement?

(* Subsequent to this work, G-Koppula-Waters gave an LWE-based standard TT scheme matching FE/iO-based efficiency.)

Relaxing Tracing

Standard Traitor Tracing

(1) No false trace: $\Pr[\exists i : i \in T, i \notin S] = \mathrm{negl}(\lambda)$

(2) Catch: $\Pr[T \neq \emptyset] \geq \Pr[D \ \epsilon\text{-distinguishes} \ (m_0, m_1)]$

$f(\cdot,\cdot)$-*Risky* Traitor Tracing

(1) No false trace: $\Pr[\exists i : i \in T, i \notin S] = \mathrm{negl}(\lambda)$

(2) Catch: $\Pr[T \neq \emptyset] \geq f(\lambda, n) \cdot \Pr[D \ \epsilon\text{-distinguishes} \ (m_0, m_1)]$

(Main) Results [G-Koppula-Russell-Waters]

Theorem. Under simple assumptions over bilinear groups, we build a secure *f-risky Traitor Tracing* scheme with $|ct|$ ciphertext size.

$$\forall\ k > 0, \quad f(\lambda, N) = \frac{k}{N} \quad \Longrightarrow \quad |\mathtt{ct}| = O(k \cdot \lambda)$$

Theorem. Secure $\frac{1}{N}$ - *risky Traitor Tracing* scheme implies differential privacy negative results.

Remaining Talk

Part I: Risky Traitor Tracing

Part II: Differential Privacy Negative Results (Luke)

Standalone Risky TT Applications

- Persistent decoder setting
  - Periodic key refreshes, decoder must work across cycles
  - Catching probability can be amplified
- Resource constrained settings
  - Get best possible tracing w/ 10 KB ciphertext overhead
- Risk averse attackers
  - Deterrence against attackers if traitors heavily penalized
  - Heavy penalty vs. Low catching probability

Framework for Risky TT – Mixed Bit Matching Encryption

$\mathtt{Setup}(1^\lambda, 1^\ell) \to (\mathtt{pk}, \mathtt{msk})$

$\mathtt{Enc\text{-}PK}(\mathtt{pk}, m) \to \mathtt{ct}$

$\mathtt{Enc\text{-}SK}(\mathtt{msk}, m, \mathbf{y} \in \{0,1\}^\ell) \to \mathtt{ct}$

$\mathtt{KeyGen}(\mathtt{msk}, \mathbf{x} \in \{0,1\}^\ell) \to \mathtt{sk}$

$\mathtt{Dec}(\mathtt{sk}, \mathtt{ct}) \to m$

$$g(\mathbf{x}, \mathbf{y}) = \bigwedge_{i \in [\ell]} x_i \vee y_i$$

Correctness Enc-PK: Dec outputs m

Correctness Enc-SK: Dec outputs m iff $g(\boldsymbol{x}, \boldsymbol{y}) = 1$

Mixed Bit Matching Security

Security: 3 properties

PK/SK CT Hiding → Distinguish

$\texttt{Enc-PK}(\texttt{pk}, m)$ vs $\texttt{Enc-SK}(\texttt{msk}, m, 1^{\ell})$

CT Hiding → Distinguish

$\texttt{Enc-SK}(\texttt{msk}, m_0, \mathbf{y}_0)$ vs $\texttt{Enc-SK}(\texttt{msk}, m_1, \mathbf{y}_1)$

Key Hiding → Distinguish

$\texttt{KeyGen}(\texttt{msk}, \mathbf{x}_0)$ vs $\texttt{KeyGen}(\texttt{msk}, \mathbf{x}_1)$

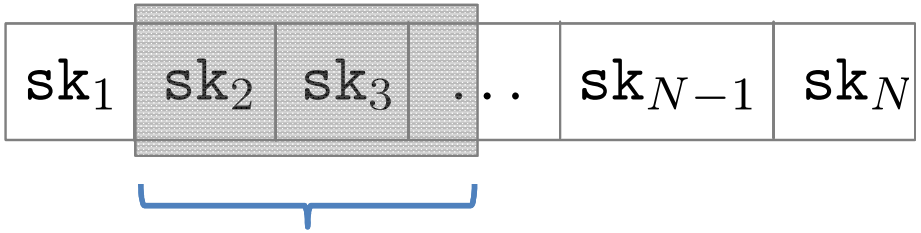
Transformation to Risky TT

$\underline{\mathtt{Setup}(1^\lambda, 1^N) \to (\mathtt{pk}, \mathtt{msk}, \{\mathtt{sk}_i\}_{i \le N})}$

$\mathtt{mBME.Setup}(1^\lambda, 1^{k+1}) \to (\mathtt{mbme.pk}, \mathtt{mbme.msk})$

$$\mathtt{pk} = \mathtt{mbme.pk}$$

$$\mathtt{msk} = \mathtt{mbme.msk}$$

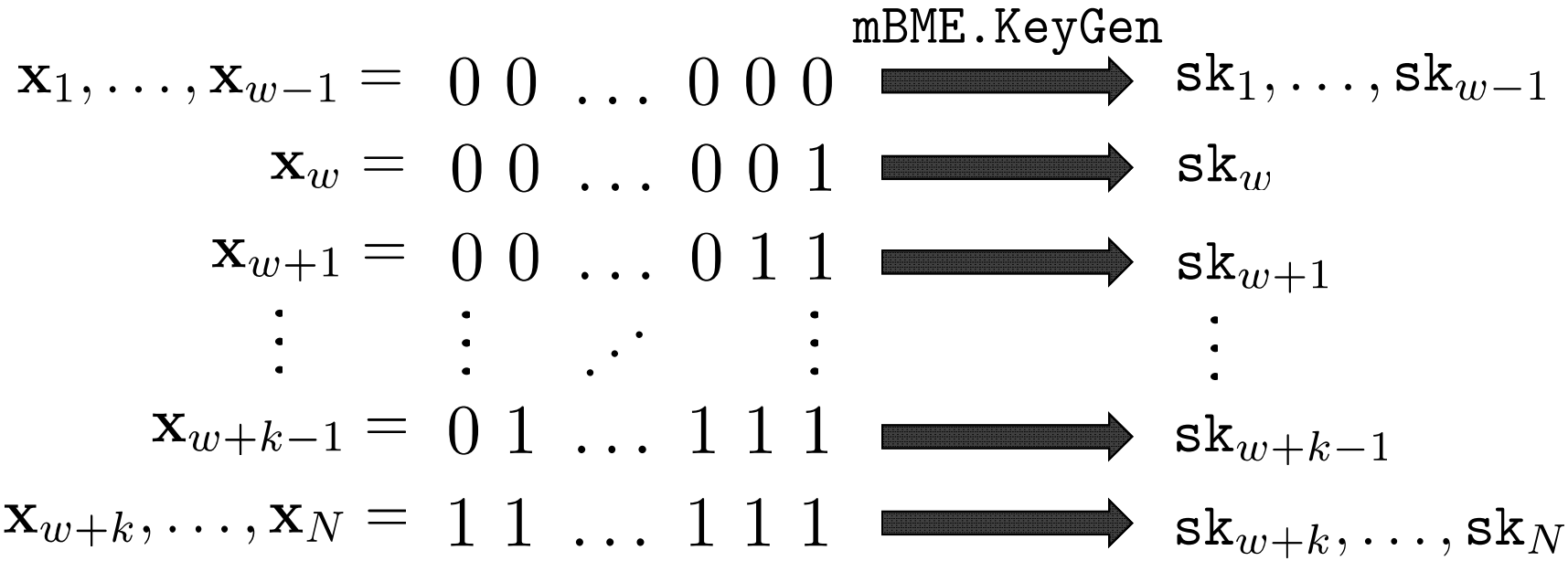


Transformation to Risky TT

$\underline{\texttt{Setup}(1^\lambda, 1^N) \to (\texttt{pk}, \texttt{msk}, \{\texttt{sk}_i\}_{i \le N})}$

$\texttt{mBME.Setup}(1^\lambda, 1^{k+1}) \to (\texttt{mbme.pk}, \texttt{mbme.msk})$

Choose random target window start location $w \in [N - k]$



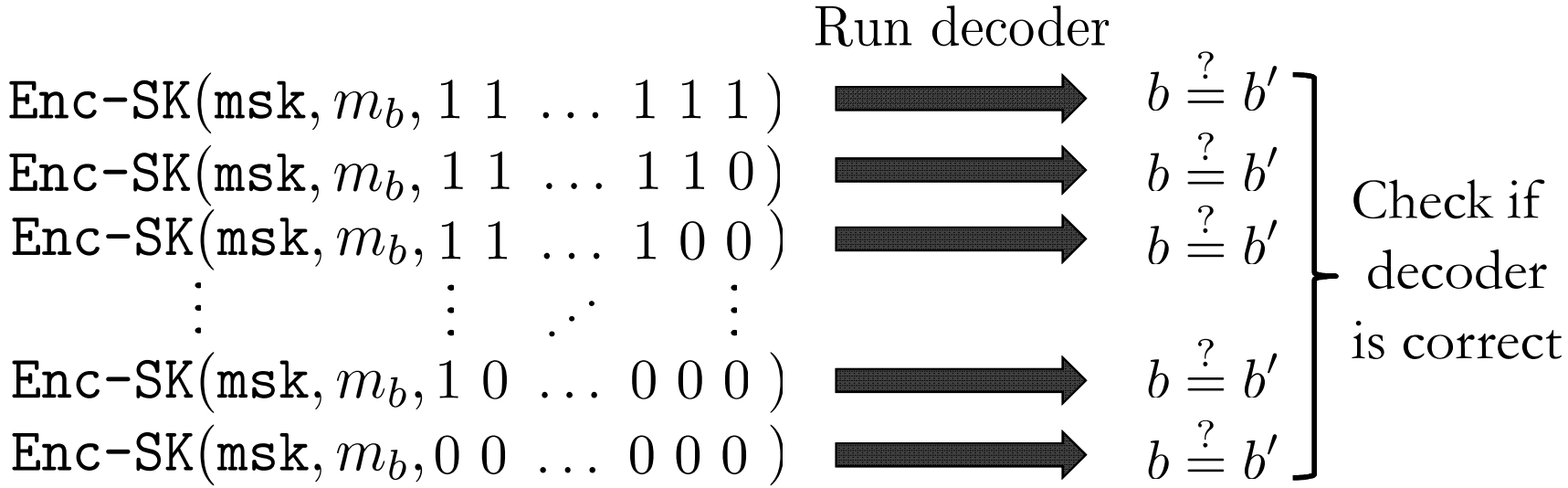
Transformation to Risky TT

$\underline{\texttt{Enc}(\texttt{pk}, m) = \texttt{mBME.Enc-PK}(\texttt{pk}, m)}$

$\underline{\texttt{Dec}(\texttt{sk}, \texttt{ct}) = \texttt{mBME.Dec}(\texttt{sk}, \texttt{ct})}$

Transformation to Risky TT

$\underline{\texttt{Trace}^{D}(\texttt{msk}, 1^{y}, m_0, m_1) \to T}$



Do this poly times

Transformation to Risky TT

$\underline{\texttt{Trace}^{D}(\texttt{msk}, 1^{y}, m_0, m_1) \rightarrow T}$

Successful decryption probability

$$\texttt{Enc-SK}(\texttt{msk}, m_b, 1\ 1\ \ldots\ 1\ 1\ 1) \Rightarrow p_1$$
$$\texttt{Enc-SK}(\texttt{msk}, m_b, 1\ 1\ \ldots\ 1\ 1\ 0) \Rightarrow p_2$$
$$\texttt{Enc-SK}(\texttt{msk}, m_b, 1\ 1\ \ldots\ 1\ 0\ 0) \Rightarrow p_3$$
$$\vdots \qquad \vdots \qquad \ddots \qquad \vdots$$
$$\texttt{Enc-SK}(\texttt{msk}, m_b, 1\ 0\ \ldots\ 0\ 0\ 0) \Rightarrow p_{k+1}$$
$$\texttt{Enc-SK}(\texttt{msk}, m_b, 0\ 0\ \ldots\ 0\ 0\ 0) \Rightarrow p_{k+2}$$

If p_i and p_{i+1} are noticeably far, then add index $(i + w - 1)$ to the set T

Missing Pieces and Other Results

- Security proof of the transformation
  - Significantly departs from existing proof techniques for TT
- Building Mixed Bit Matching Encryption from Bilinear Maps
- Generic risky amplification
  - Improving success probability of tracing

Remaining Talk

Part I: Building Risky Traitor Tracing

Part II: Differential Privacy Negative Results (Luke)


Q&A