Efficient Non-Malleable Codes and Key-derivations against Poly-size Tampering Circuits

PRATYAY MUKHERJEE

(Aarhus University)

Joint work with

Sebastian Faust, Daniele Venturi and Daniel Wichs

EUROCRYPT 2014, COPENHAGEN

May 12, 2014

AARHUS UNIVERSITY

This talk

Two Parts

N psf

Part-1

Efficient Non-malleable **Codes**
against Poly-size circuits

Part-2

Efficient Non-malleable **Key-derivation**
against Poly-size circuits

Mftt

Part-1

Efficient Non-malleable Codes against Poly-size circuits

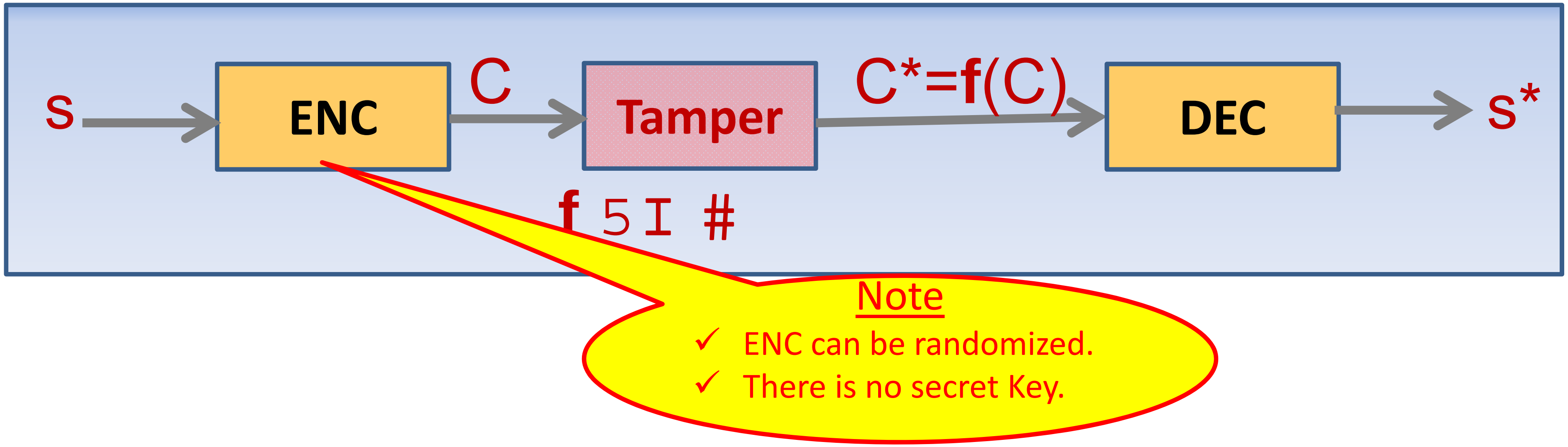
Non-malleable Codes (Informally)

A modified codeword contains either original or unrelated message.

E.g. Can not flip one bit of encoded message by modifying the codeword.

The "Tampering Experiment"

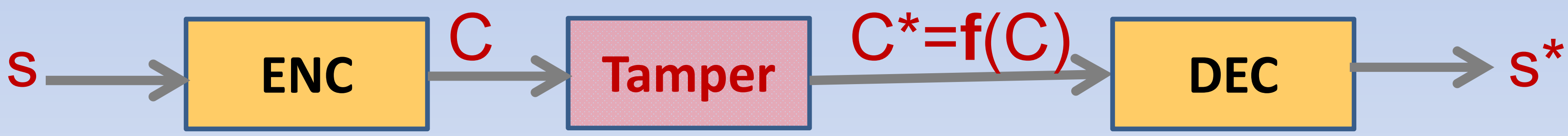
- Consider the following experiment for some encoding scheme (**ENC,DEC**)



Goal:

Design encoding scheme **(ENC,DEC)** which is **Non-malleable** for an **"interesting"** class I

$\text{Tamper}^f(s)$



f 5 I #

If C* = C return **same** Else return s*

Definition [DPW 10]:

A code (**ENC, DEC**) is **non-malleable** w.r.t. I #if

; **f** ∈ I and ; s_0, s_1 we have:

$$\text{Tamper}^f(s_0) \approx \text{Tamper}^f(s_1)$$

Application : Tamper-Resilient Cryptography

- Non-malleable codes are used to protect against key-tampering attacks.
- How ?
  - Encode the key using NMC.
  - The tampering adversary can not modify the encoded key to some related key.

Limitation and Possibility

Limitation: For any (**ENC, DEC**) there exists $\mathbf{f}_{bad}$ which decodes C, flips 1-bit and re-encodes to C*.

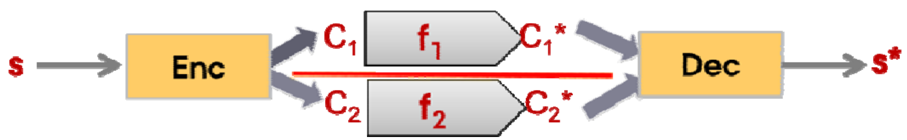
Corollary-1: It is impossible to construct encoding scheme which is non-malleable w.r.t. all functions I_{all} .

Corollary-2: It is impossible to construct efficient encoding scheme which is non-malleable w.r.t. all efficient functions I_{eff} .

Question: How to restrict I ?

Way-1: Restrict granularity

- Codeword consists of components which are independently tamperable.
- Example: Split-state tampering [DPW10, LL12, DKO13, ADL13, CG13, FMNV13, ADK14]:



Way-2: Restrict complexity

- The whole codeword is tamperable but only with functions that are not "too complicated".

Our Focus!

Our Result

recall

Corollary-2: It is impossible to construct efficient encoding scheme which is non-malleable w.r.t. all efficient functions I_{eff} .

Main Result: "The next best thing"

For any fixed polynomial P, there exists an efficient non-malleable code for any family of functions $|I\ \#| \leq 2^P$.

Corollary-3

For any fixed polynomial P, there exists an efficient non-malleable code for all circuits of size $\leq$ **P** .

Our Result

A similar result [CG 14]
But the encoding/decoding becomes
"inefficient" in order to get negligible error

Main Result: "The next best thing"

For any fixed polynomial P, there exists an efficient non-malleable code for any family of functions $|I\ \#| \leq 2^{P}$.

Corollary-3

For any fixed polynomial P, there exists an efficient non-malleable code for all circuits of size $\leq$ **P** .

Caveat: Our results hold in CRS model.

NMC in CRS model

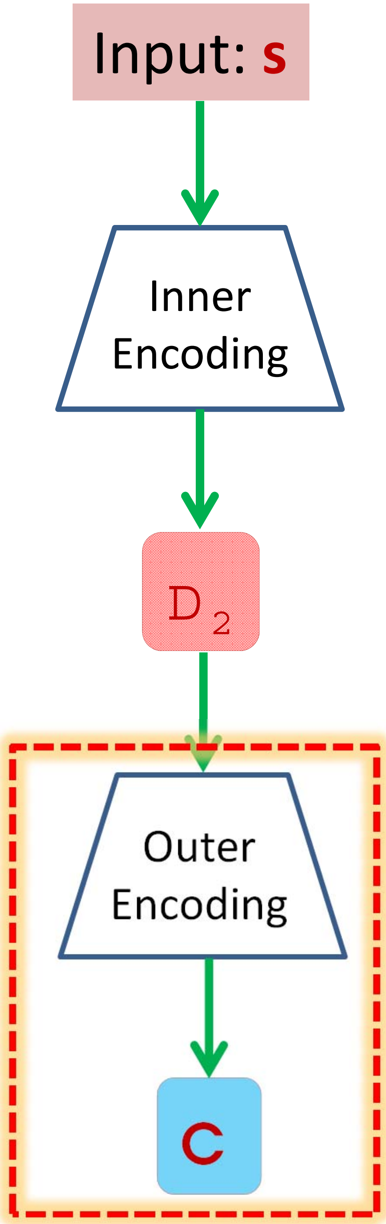
- Fix some polynomial **P**

- We construct a family of efficient codes parameterized by **CRS**: ($\mathbf{ENC}_{\mathbf{CRS}}$, $\mathbf{DEC}_{\mathbf{CRS}}$)

- We show that, w.h.p. over the random choice of **CRS** : ($\mathbf{ENC}_{\mathbf{CRS}}$, $\mathbf{DEC}_{\mathbf{CRS}}$) is an NMC w.r.t. all tampering circuits of size $\leq$ **P**

Although **P** is chosen apriori, the tampering circuit can be chosen from the family of all circuits of size $\leq$ **P** adaptively.

The Construction Overview



Intuitions (outer encoding)

Ingredient: a **t**-wise independent hash function **h**

Fixed by **CRS**

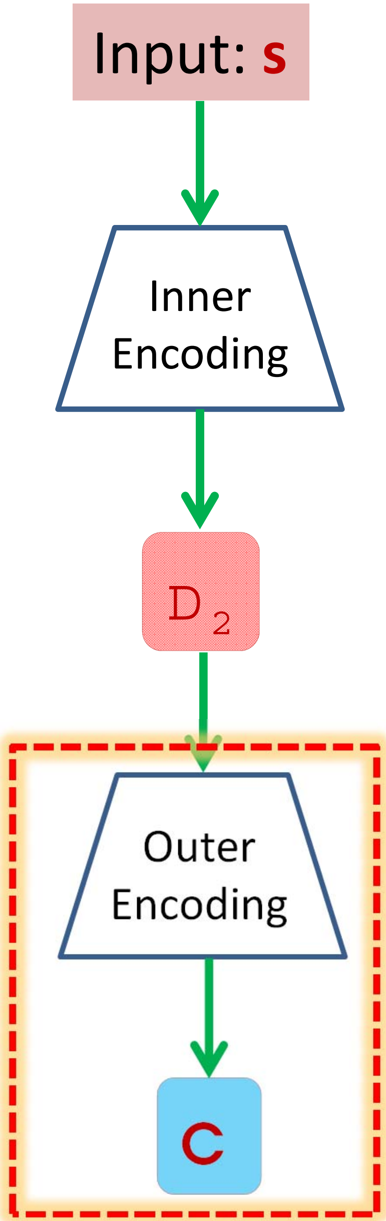
$D = D_2 \,||\, h(D_2)$

D is Valid $\Leftrightarrow$ D is of the form $s \,||\, h(s)$

- We choose **CRS** such that |Circuit computing **h**| > **P** ⇒ No circuit of size ≤ **P** can compute **h** on "too many" points. (Proof: Probabilistic Method)

- For every tampering function **f** there is a "small set" S_f such that if a tampered codeword is valid, then it is in S_f w.h.p.

The Construction Overview

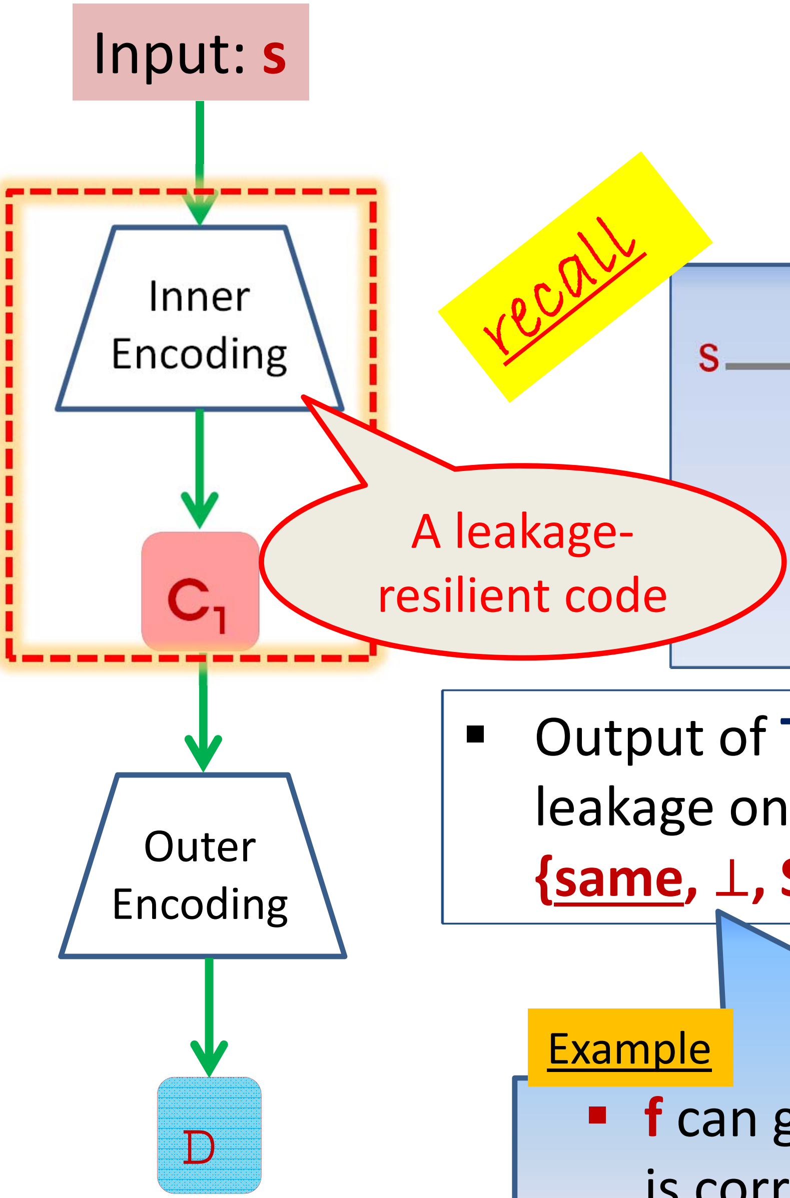


Intuitions (outer encoding)

- For every tampering function **f** there is a "small set" **S_f** such that if a tampered codeword is valid, then it is in **S_f** w.h.p.

We call this property Bounded Malleability which ensures that the tampered codeword does not contain "too much information" about the input codeword

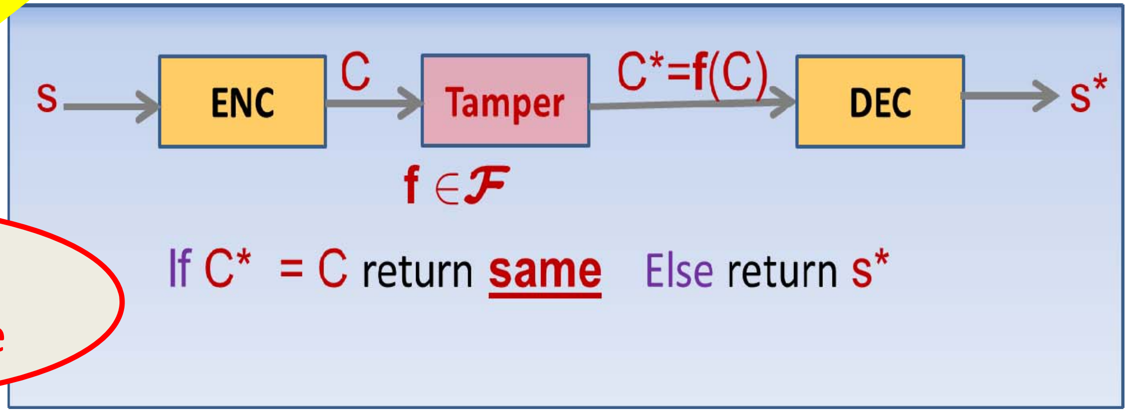
The Construction Overview



recall

Intuitions (Inner encoding)

Tamper^f(s)



- Output of **Tamper^f(s)** can be thought of as some leakage on C_1. w.h.p. the leakage range is "small": {same, ⊥, S_f}

Example

- f can guess some bit(s) of D_2 and if the guess is correct, leave D same otherwise overwrites to some invalid code.

Leakage-Resilient Code

Def [DDV 10]: A code (**LRENC, LRDEC**) is **leakage-resilient** w.r.t. J if

; **g** ∈ J and ; **s** : **g**(**LRENC**(**s**)) ≈ **g**(**U**)

Construction [DDV 10]: Let **h'** be a **t**-wise hash function. Then to encode **s** choose a random **r** and output **c** = **r** || **h'** (**r**) ⊕ **s**

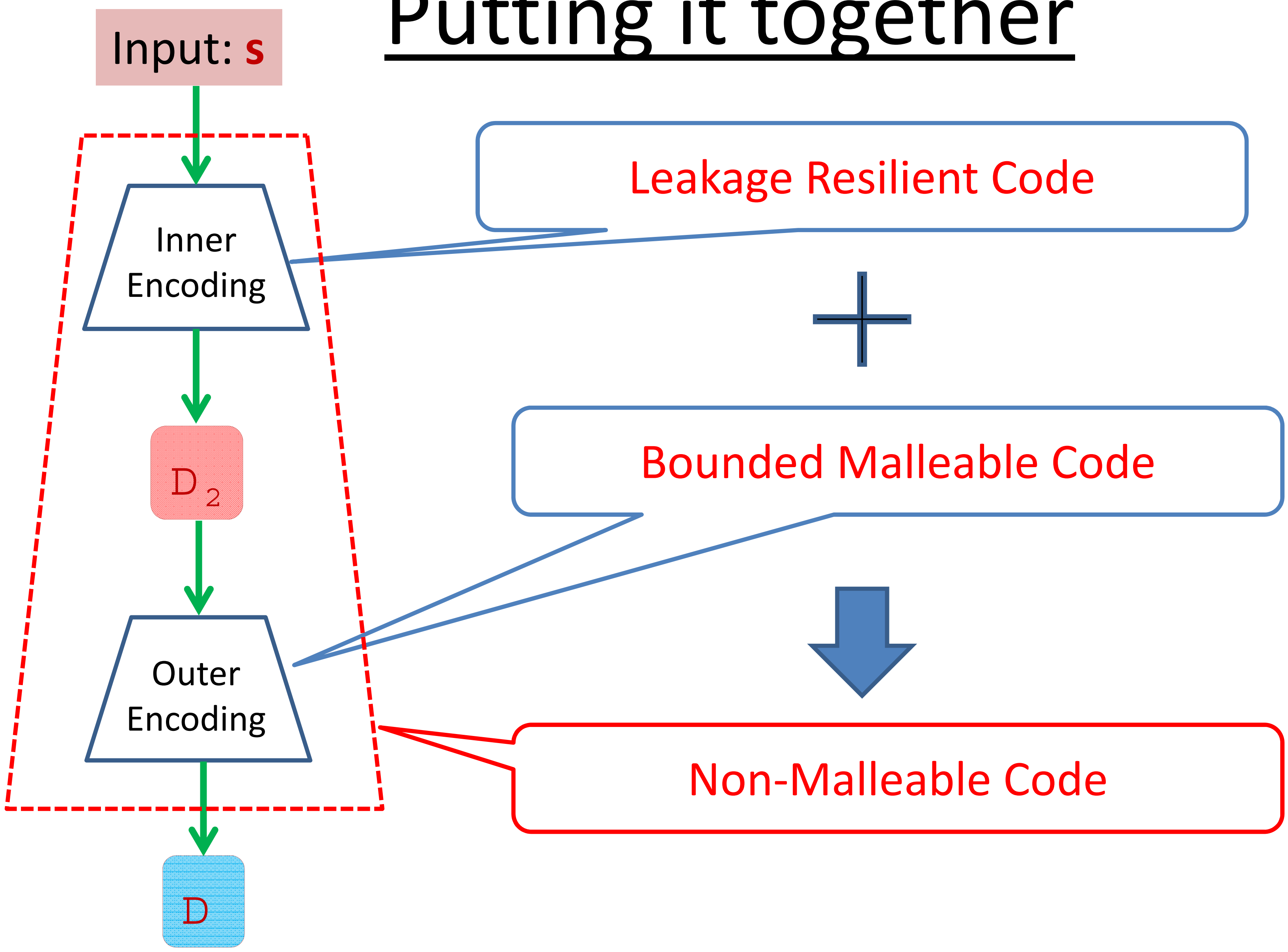
Our Inner Encoding

Analysis by [DDV 10] uses bound for extractor and therefore, **r** ≥ **s** (rate ≤ 1/2) even if the leakage ℓ is small

We show: The construction is an LRC as long as: **r** > ℓ even if **r** <<**s**

We use the same construction but improved analysis to achieve optimal rate ≈ 1.

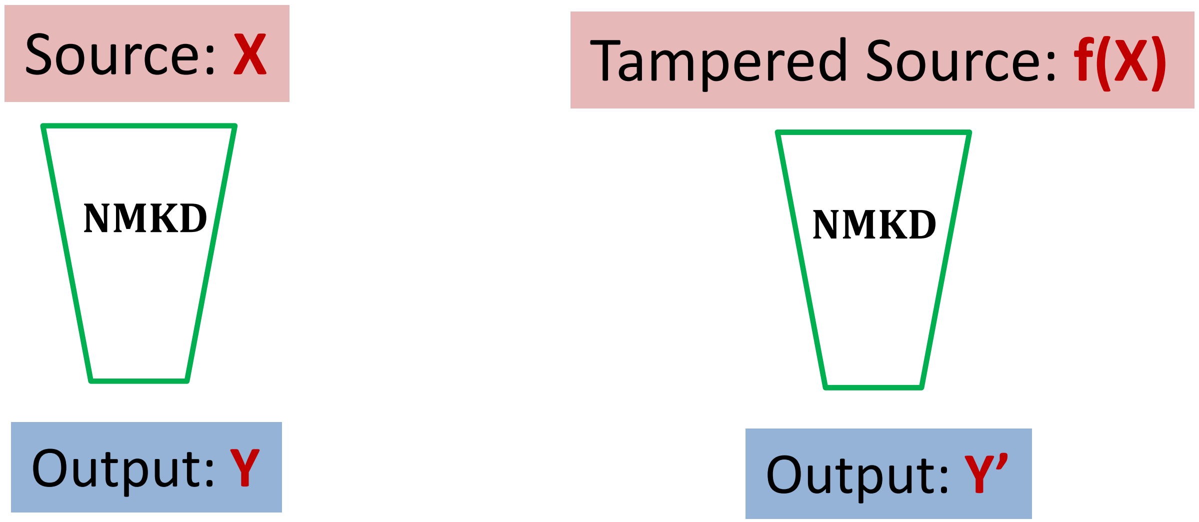
Putting it together



Part-2

Efficient Non-malleable **Key-derivation** **(NMKD)** against Poly-size circuits

NMKD: A new primitive



NMKD guarantees that if **f(X)** $\neq$ **X** then **(Y, Y')** $\approx$ **(U, Y')**

A dual of Non-Malleable Extractor

NMKD: Defintion

Real $^{\phi, f}$

Sample $x \leftarrow U$

If $f(x) = x$

return $(\phi(x), \underline{\textbf{same}})$

Else return $(\phi(x), \phi(f(x)))$

$\approx$

Ideal $^{\phi, f}$

Sample $x \leftarrow U$; $y \leftarrow U'$

If $f(x) = x$

return $(y, \underline{\textbf{same}})$

Else return $(y, \phi(f(x)))$

Definition: A function ϕ is **NMKD** w.r.t. $\mathtt{I}$ iff

; $f \in \mathtt{I}$ if above holds

Theorem (NMKD)

For any $\mathtt{I}$ of size $\leq 2^P$, a randomly chosen **t**-wise independent hash function is an NMKD w.h.p. as long as **t > P**

Conclusion

- The first construction of non-granular efficient Non-malleable code.
  - Our construction is information theoretic and achieves optimal rate.
- A new primitive Non-Malleable Key-derivation.
  - Application to construct Tamper-resilient Stream Cipher.
- Open:
  - New Application of NMKD.

Thank You !