Non-Malleable Codes for Partial Functions with Manipulation Detection

Aggelos Kiayias Feng-Hao Liu

Yiannis Tselekounis

Edin. & FAU

CRYPTO 2018

Outline

- Introduction to non-malleable codes
- Adversarial model, motivation
- Results, constructions
- Intuition

Encoding schemes

An *encoding scheme* is a pair of algorithms $(\mathsf{Enc}, \mathsf{Dec})$, satisfying *correctness*:

$$\text{for any message } s,\ \mathsf{Dec}(\mathsf{Enc}(s)) = s$$

Encoding schemes

An *encoding scheme* is a pair of algorithms $(\mathsf{Enc}, \mathsf{Dec})$, satisfying *correctness*:

$$\text{for any message } s,\ \mathsf{Dec}(\mathsf{Enc}(s)) = s$$

Error-correction codes: guarantee correctness in the presence of faults

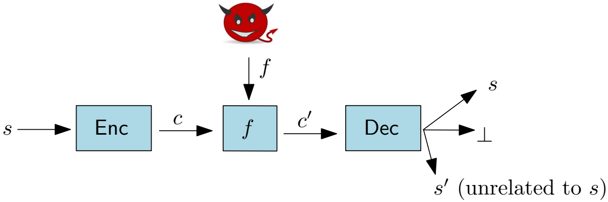
Non-malleable codes [DPW10,18]

Non-malleable codes [DPW10,18]

Non-malleability: any modified codeword does not decode to a message related to/different from, the original

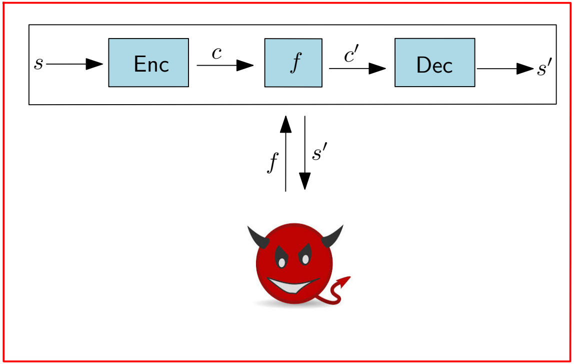
Non-malleable codes [DPW10,18]

Non-malleability: any modified codeword does not decode to a message related to/different from, the original



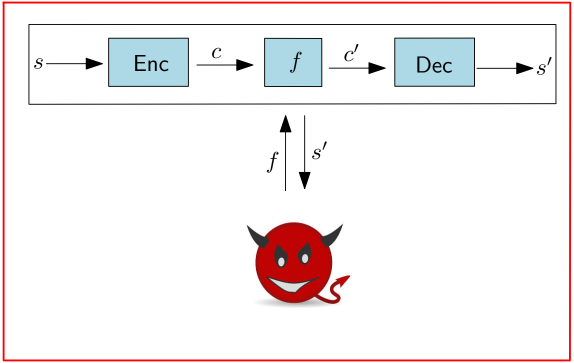
Non-malleability [DPW10,18]

Real



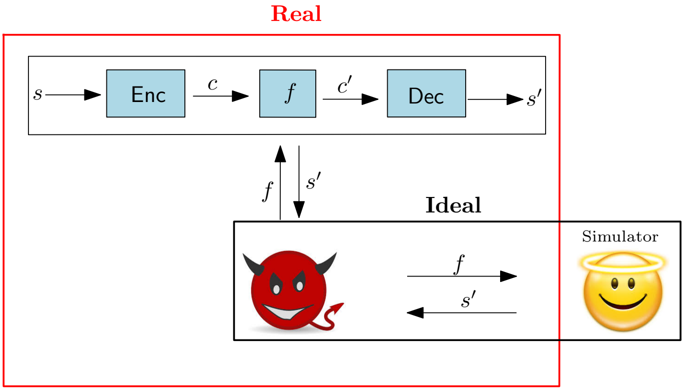
Non-malleability [DPW10,18]

Real

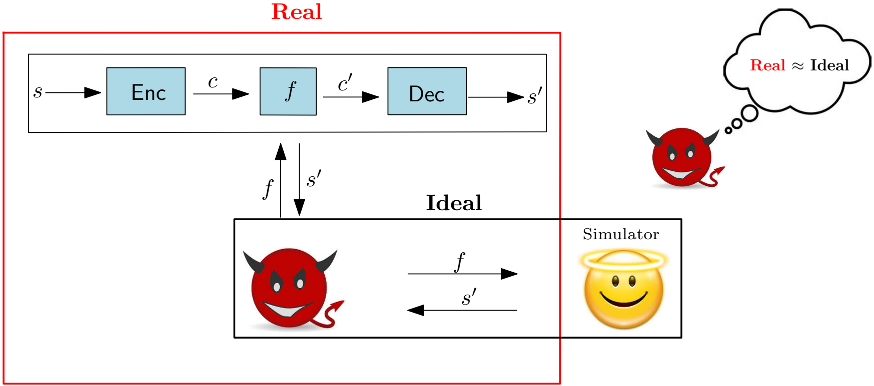


Simulator

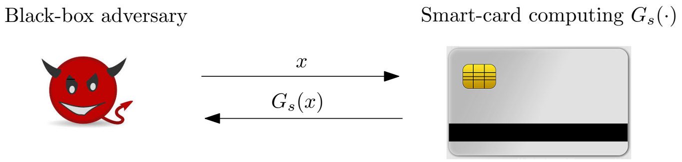
Non-malleability [DPW10,18]



Non-malleability [DPW10,18]



Application of NMC



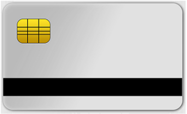
Application of NMC

Black-box adversary

Smart-card computing $G_s(\cdot)$

x →

← $G_s(x)$

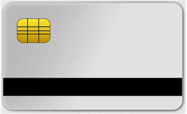


Tampering adversary

Smart-card computing $G_s(\cdot)$

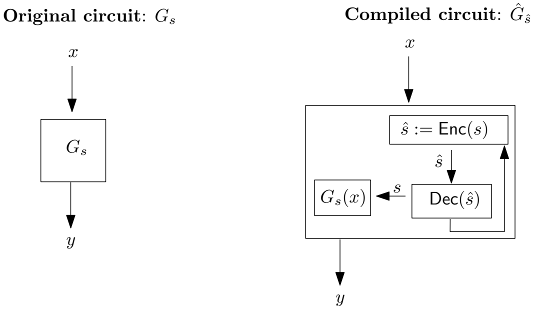
f, x →

← $G_{f(s)}(x)$



Application of NMC

Assuming (Enc, Dec) is a non-malleable code w.r.t. $\mathcal{F}$.



Non-malleability: for any $f \in \mathcal{F}$, $f(\hat{s})$ is simulatable and independent of s

Admissible function classes

Non-malleability is impossible against arbitrary tampering function classes

Admissible function classes

Non-malleability is impossible against arbitrary tampering function classes

For instance, consider a class containing the function $f(c) := \mathsf{Enc}(\mathsf{Dec}(c) + 1)$

Admissible function classes

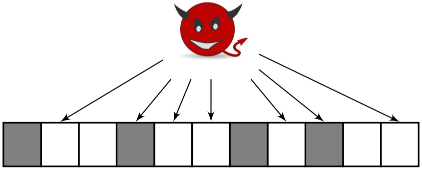
Proposed function classes: Split-state functions [ADL14, DKO13, ADKO15, LL12, AAG+16, DPW10, KLT16], bit-wise tampering and permutations [DPW10, AGM+15a, AGM+15b], bounded-size function classes [FMVW14], bounded depth/fan-in circuits [BDKM16], space-bounded tampering [FHMV17,BDKM18], block-wise tampering [CKM11,CGM+15], AC0 circuits, bounded-depth decision trees and streaming adversaries [BDKM18], small-depth circuits [BDGMT18], and others.

Admissible function classes

Proposed function classes: Split-state functions [ADL14, DKO13, ADKO15, LL12, AAG+16, DPW10, KLT16], bit-wise tampering and permutations [DPW10, AGM+15a, AGM+15b], bounded-size function classes [FMVW14], bounded depth/fan-in circuits [BDKM16], space-bounded tampering [FHMV17,BDKM18], block-wise tampering [CKM11,CGM+15], AC0 circuits, bounded-depth decision trees and streaming adversaries [BDKM18], small-depth circuits [BDGMT18], and others.

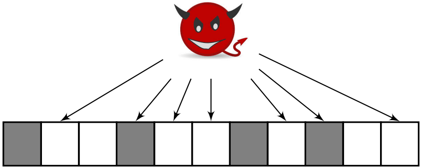
This work: Partial functions

NMC for Partial Functions

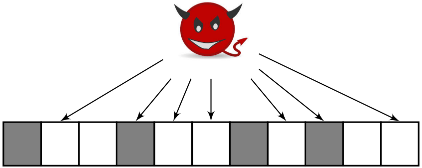


We allow *read/write* access to arbitrary subsets of codeword locations, with bounded cardinality.

Basic definitions

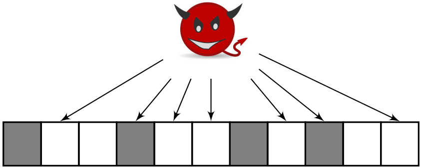


Basic definitions



- **Information rate**: the ratio of message to codeword, length, as the message length goes to infinity.

Basic definitions



- **Information rate**: the ratio of message to codeword, length, as the message length goes to infinity.
- **Access rate**: the fraction of the number of bits (symbols) the attacker is allowed to access over, the total codeword length.

Main Goal

Is it possible to construct efficient (high information rate) non-malleable codes for partial functions, while allowing the attacker to access almost the entire codeword (high access rate)?

Motivation

- Attackers with high access rate could still create correlated codewords

Motivation

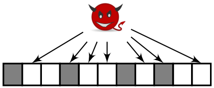
- Attackers with high access rate could still create correlated codewords
- Partial functions comply with existing attacks, e.g., [BDL97, BDL01, BS97]

Motivation

- Attackers with high access rate could still create correlated codewords
- Partial functions comply with existing attacks, e.g., [BDL97, BDL01, BS97]
- The passive analog of the primitive implies All-Or-Nothing-Transforms [Riv97], having numerous applications

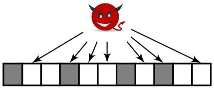
Motivation

- Attackers with high access rate could still create correlated codewords
- Partial functions comply with existing attacks, e.g., [BDL97, BDL01, BS97]
- The passive analog of the primitive implies All-Or-Nothing-Transforms [Riv97], having numerous applications



Motivation

- Attackers with high access rate could still create correlated codewords
- Partial functions comply with existing attacks, e.g., [BDL97, BDL01, BS97]
- The passive analog of the primitive implies All-Or-Nothing-Transforms [Riv97], having numerous applications



- Constant functions are excluded from the model, thus it potentially allows stronger primitives

Results

Results

- **Stronger notion**: Non-malleability with manipulation detection ($\mathrm{MD\text{-}NMC}$),

$$\mathsf{Dec}(f(c)) \in \{s, \perp\}$$

Results

- **Stronger notion**: Non-malleability with manipulation detection ($\mathrm{MD\text{-}NMC}$),

$$\mathsf{Dec}(f(c)) \in \{s, \perp\} \quad (\mathrm{MD} \not\Rightarrow \mathrm{MD\text{-}NMC})$$

Results

- **Stronger notion**: Non-malleability with manipulation detection $(\mathrm{MD\text{-}NMC})$,

$$\mathsf{Dec}(f(c)) \in \{s, \perp\} \quad (\mathrm{MD} \not\Rightarrow \mathrm{MD\text{-}NMC})$$

- Assuming OWF, we construct $\mathrm{MD\text{-}NMC}$ in the CRS model, with information rate 1 and access rate $1 - 1/\Omega(\log k)$

Results

- **Stronger notion**: Non-malleability with manipulation detection $(\mathrm{MD\text{-}NMC})$,

$$\mathsf{Dec}(f(c)) \in \{s, \perp\} \quad (\mathrm{MD} \not\Rightarrow \mathrm{MD\text{-}NMC})$$

- Assuming OWF, we construct $\mathrm{MD\text{-}NMC}$ in the CRS model, with information rate 1 and access rate $1 - 1/\Omega(\log k)$

- Assuming OWF, we construct $\mathrm{MD\text{-}NMC}$ in the standard model, with information rate $1 - 1/\Omega(\log k)$ and access rate $1 - 1/\Omega(\log k)$ (alphabet size: $O(\log k)$)

Results

- **Stronger notion**: Non-malleability with manipulation detection $(\mathrm{MD\text{-}NMC})$,

$$\mathsf{Dec}(f(c)) \in \{s, \perp\} \quad (\mathrm{MD} \;\not\Rightarrow\; \mathrm{MD\text{-}NMC})$$

- Assuming OWF, we construct $\mathrm{MD\text{-}NMC}$ in the CRS model, with information rate 1 and access rate $1 - 1/\Omega(\log k)$

- Assuming OWF, we construct $\mathrm{MD\text{-}NMC}$ in the standard model, with information rate $1 - 1/\Omega(\log k)$ and access rate $1 - 1/\Omega(\log k)$ (alphabet size: $O(\log k)$)

- Our results imply efficient All-Or-Nothing-Transforms under standard assumptions

Challenges

Challenges

- Non-malleability for partial functions with concrete access rate 1 is impossible

Challenges

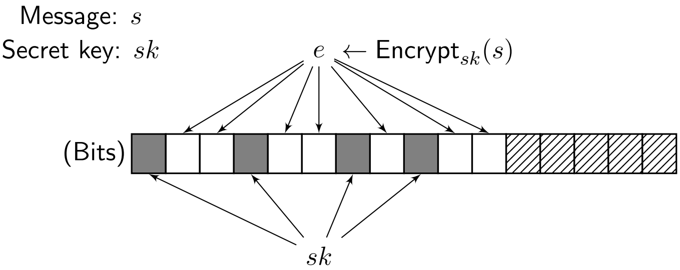
- Non-malleability for partial functions with concrete access rate 1 is impossible
- **Impossibility on the information-theoretic setting [CG14]**: assuming constant access/information rate, security is achievable only with constant probability

Challenges

Towards an encryption-based solution:

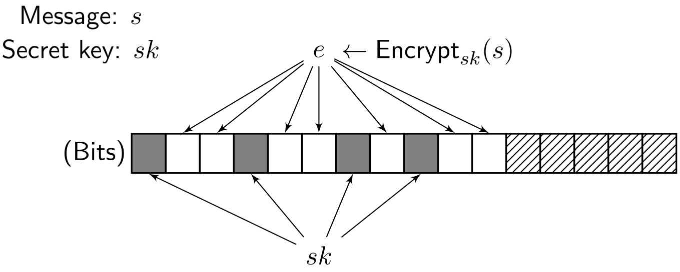
Challenges

Towards an encryption-based solution:



Challenges

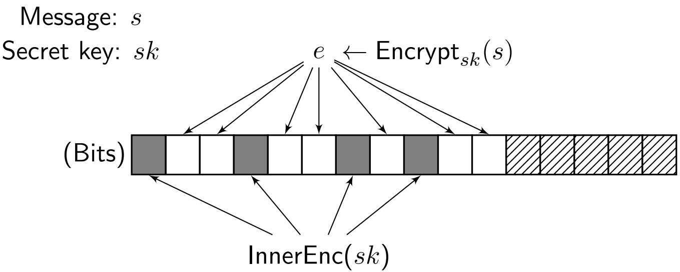
Towards an encryption-based solution:



Security breaks by accessing $O(|sk|/|s|)$ codewords bits

Challenges

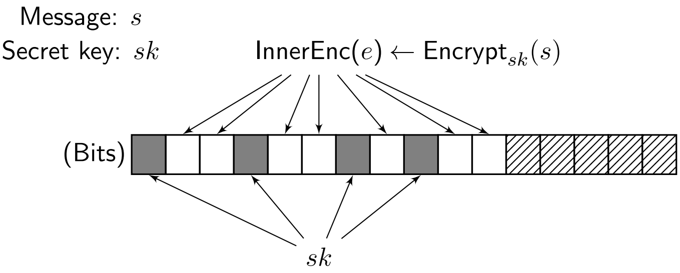
Towards an encryption-based solution:



Security breaks by accessing $O(|sk|/|s|)$ codewords bits

Challenges

Towards an encryption-based solution:



Challenges

Question: Is it possible to achieve access rate greater than $O(|sk|/|c|)$?

Challenges

Question: Is it possible to achieve access rate greater than $O(|sk|/|c|)$?

More generally: Can we achieve access rate greater than what our weakest primitive sustains?

Challenges

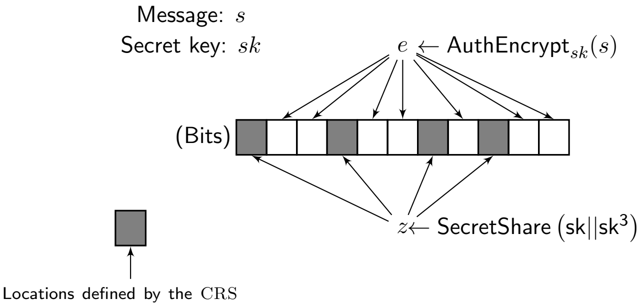
Main observation: the structure of the codeword is fixed and known to the attacker

Challenges

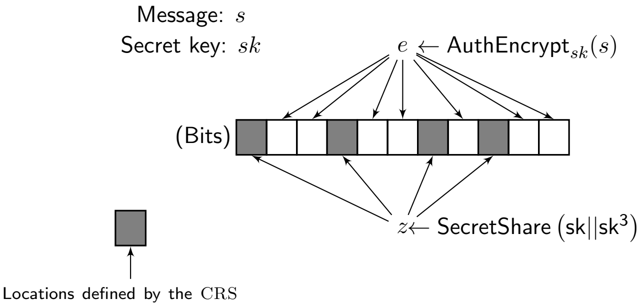
Main observation: the structure of the codeword is fixed and known to the attacker

Idea: hide the structure via randomization

Construction in the CRS model

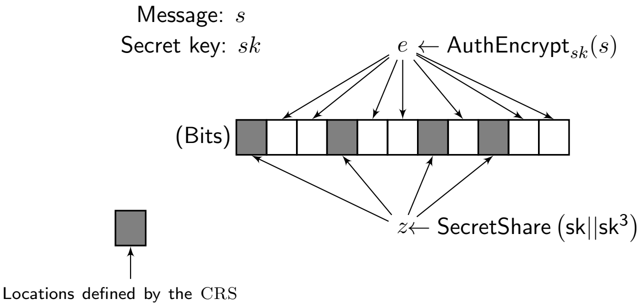


Construction in the CRS model



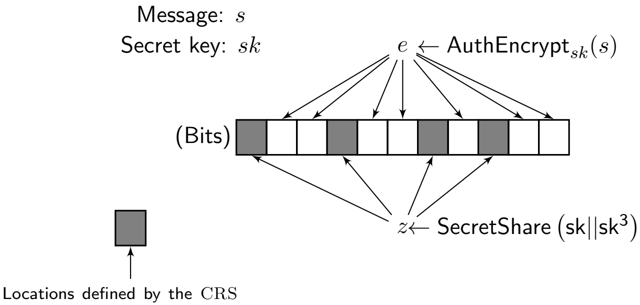
- Due to the shuffling, the attacker learns nothing about sk, sk^3. Let $(sk, sk^3) \xrightarrow{f} (sk', sk'')$

Construction in the CRS model



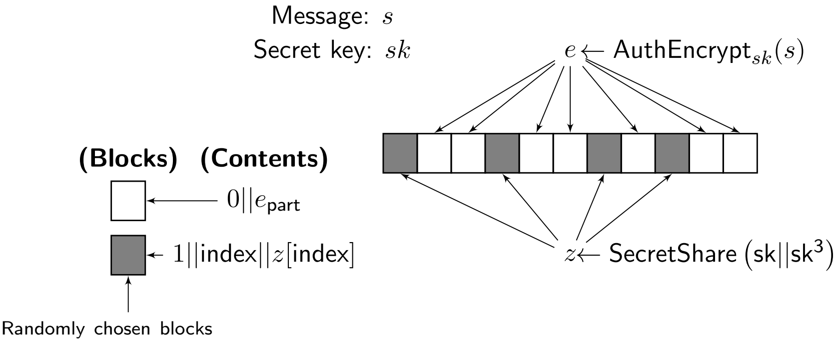
- Due to the shuffling, the attacker learns nothing about sk, sk^3. Let $(sk, sk^3) \xrightarrow{f} (sk', sk'')$
- If $(sk, sk^3) \neq (sk', sk'')$, then $\Pr[sk'^3 = sk''] \leq$ negl, otherwise we can recover sk

Construction in the CRS model



- Due to the shuffling, the attacker learns nothing about sk, sk^3. Let $(sk, sk^3) \xrightarrow{f} (sk', sk'')$
- If $(sk, sk^3) \neq (sk', sk'')$, then $\Pr[sk'^3 = sk''] \leq$ negl, otherwise we can recover sk
- Thus, if $sk \neq sk'$ or $sk^3 \neq sk''$, the simulator outputs $\perp$, otherwise, security follows by the authenticity property of the encryption scheme

Removing the CRS



Block size: $\log(k)$

Conclusions

- **Stronger notion**: Non-malleable codes with manipulation detection $(\mathrm{MD\text{-}NMC})$

Conclusions

- **Stronger notion**: Non-malleable codes with manipulation detection $(\mathrm{MD\text{-}NMC})$
- **Constructions**: efficient $\mathrm{MD\text{-}NMC}$ for partial functions

Conclusions

- **Stronger notion**: Non-malleable codes with manipulation detection $(\mathrm{MD\text{-}NMC})$

- **Constructions**: efficient $\mathrm{MD\text{-}NMC}$ for partial functions

- **Applications**: tamper-resilient cryptography (boolen/aritmetic circuits), secure communication over adversarial channels (Wire-Tap channels), AONTs

Thank you!