Securely Instantiating `Half Gates' Garbling in the Standard Model

Anasuya Acharya	Bar-Ilan University	→ Aarhus University
Karen Azari	ETH Zurich	→ University of Vienna
Mirza Ahad Baig	ISTA	
Dennis Hofheinz	ETH Zurich	
Chethan Kamath	IIT Bombay	

Funded by the European Union

European Research Council
Established by the European Commission

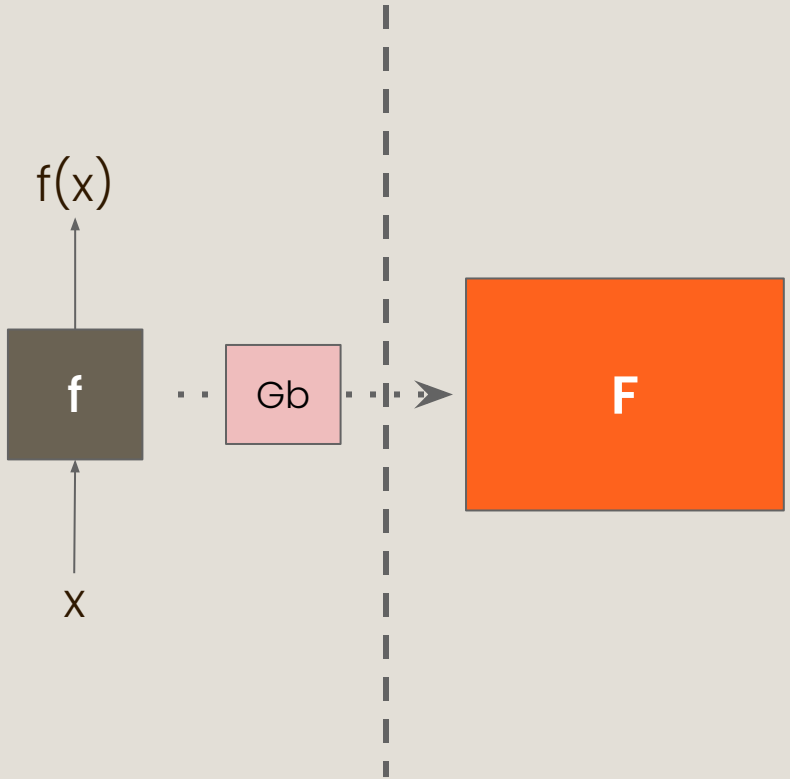
AARHUS UNIVERSITET

GARBLING SCHEMES

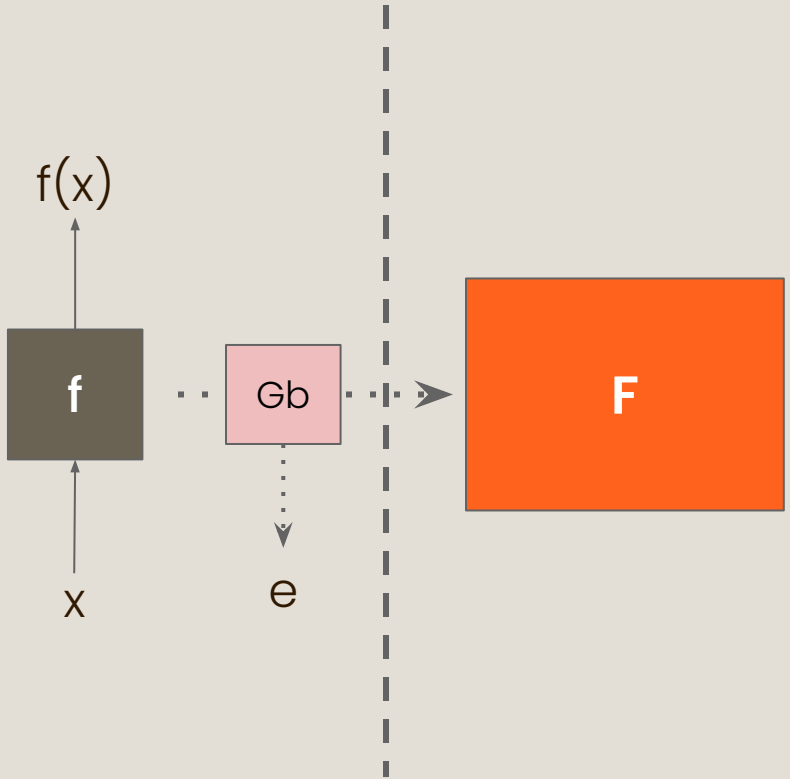
GARBLING SCHEMES

GARBLING SCHEMES

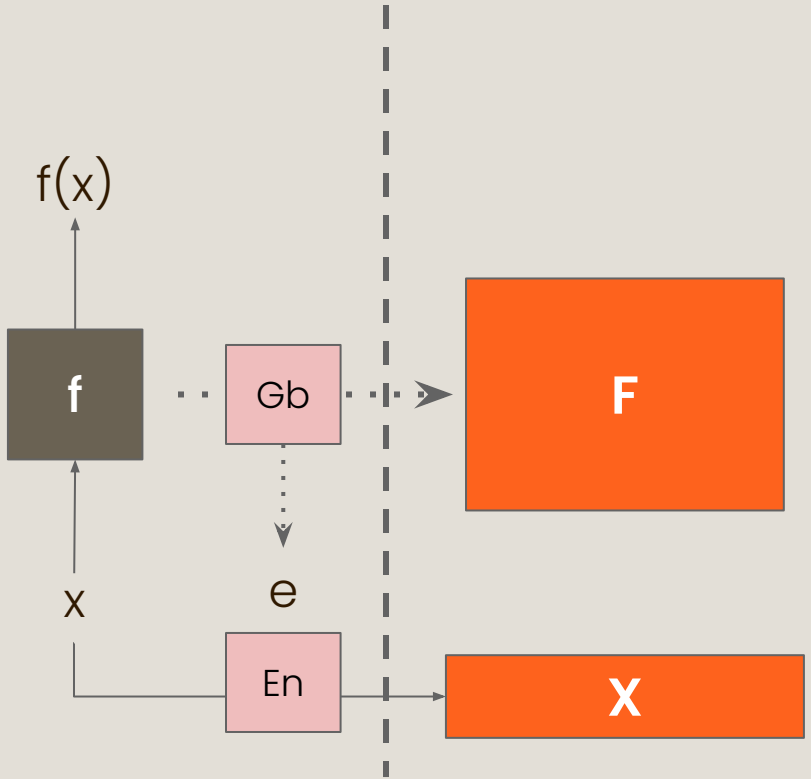
GARBLING SCHEMES



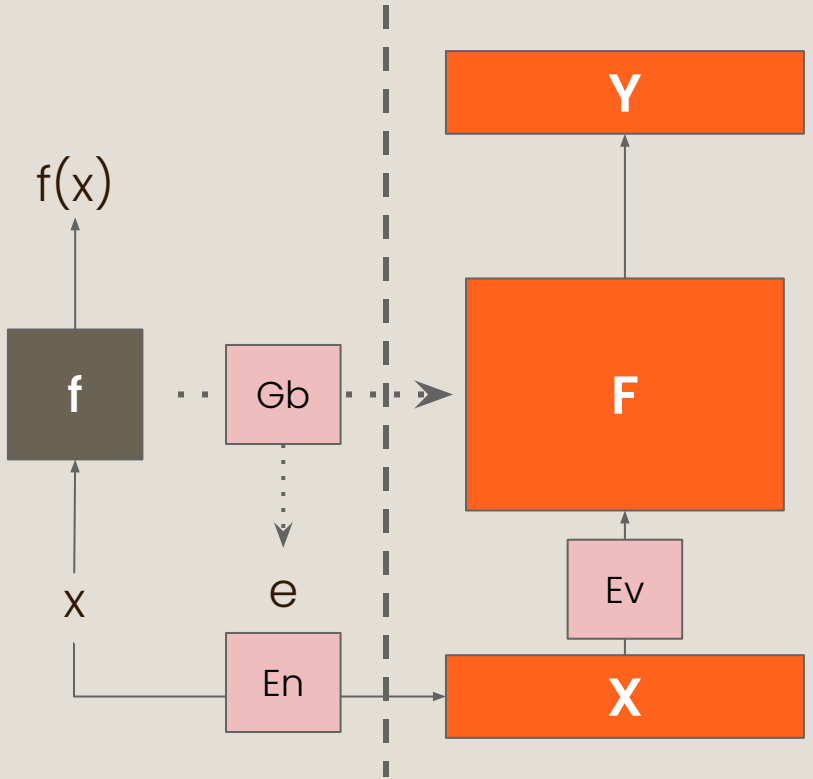
GARBLING SCHEMES



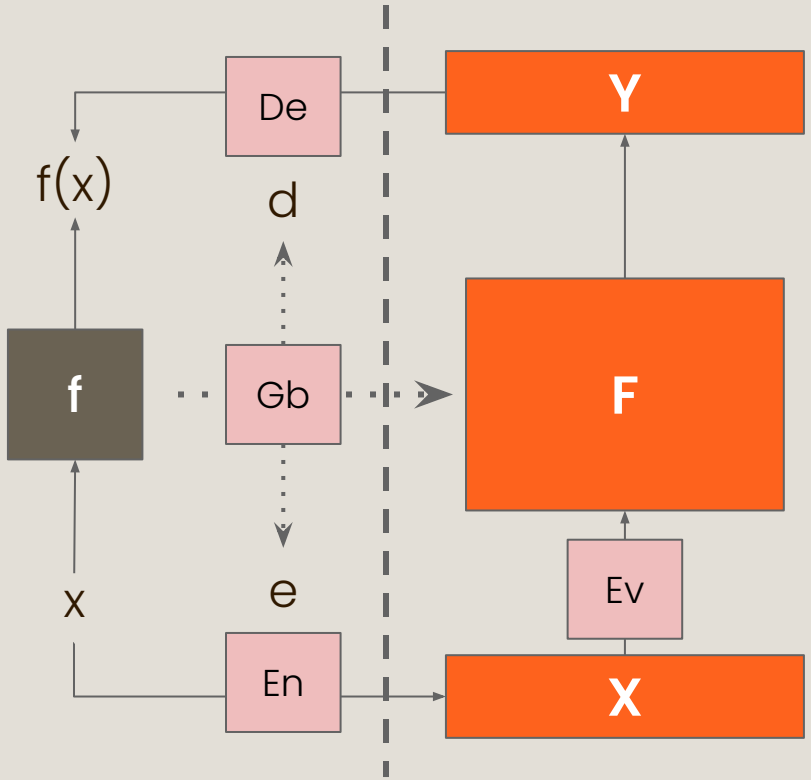
GARBLING SCHEMES



GARBLING SCHEMES

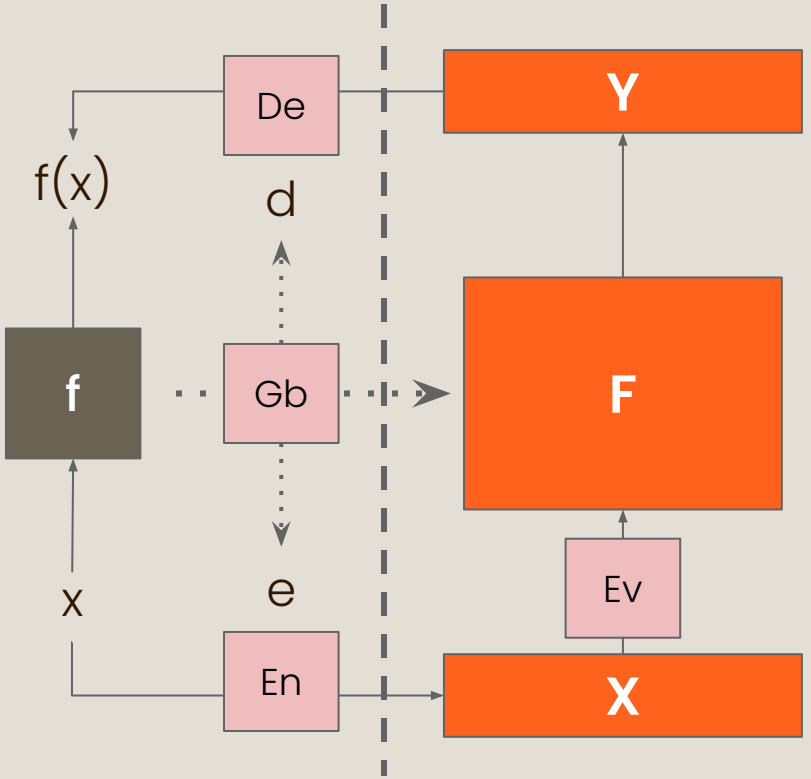


GARBLING SCHEMES

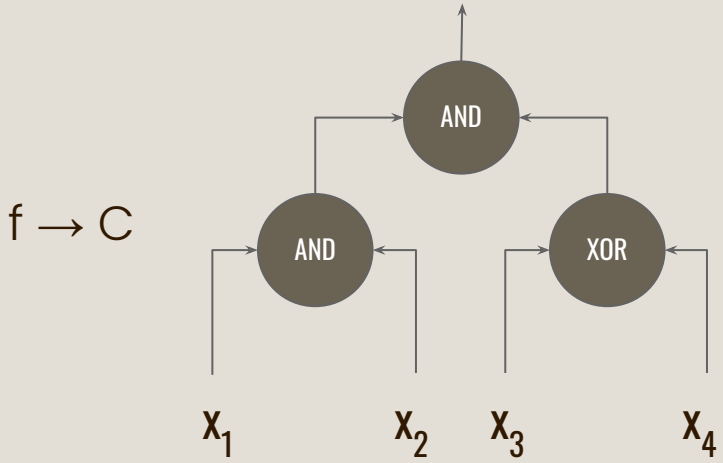


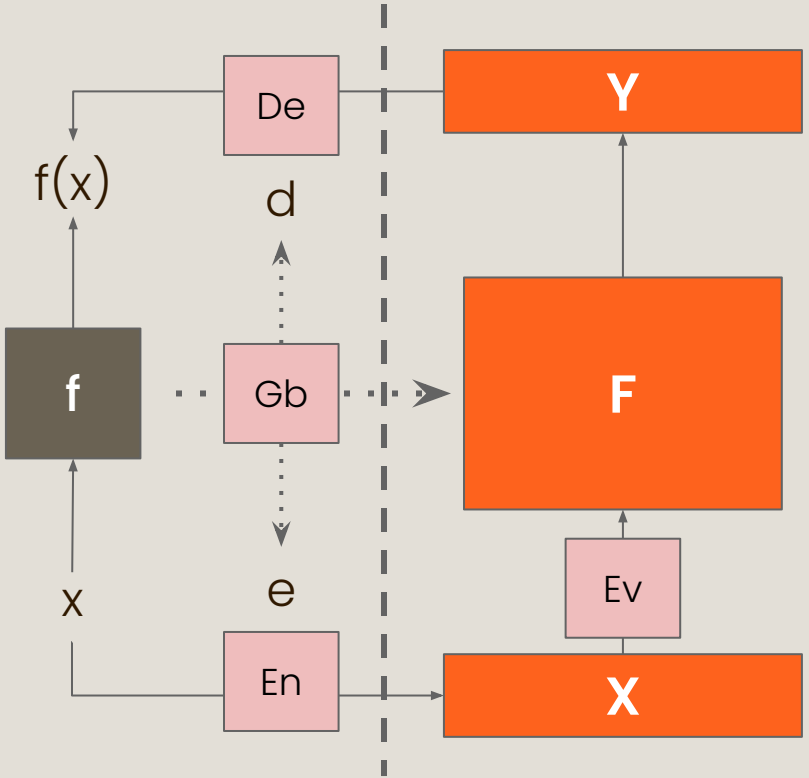
GARBLING SCHEMES

$f \rightarrow C$



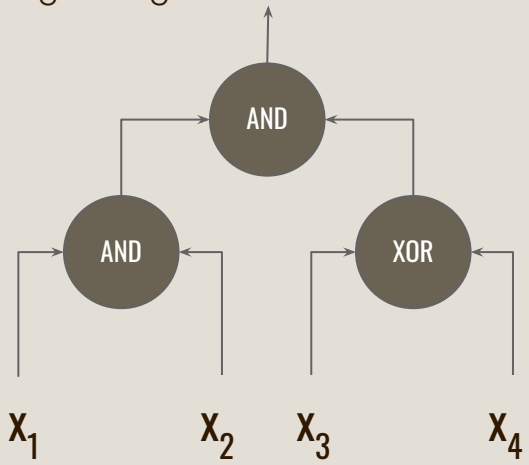
GARBLING SCHEMES





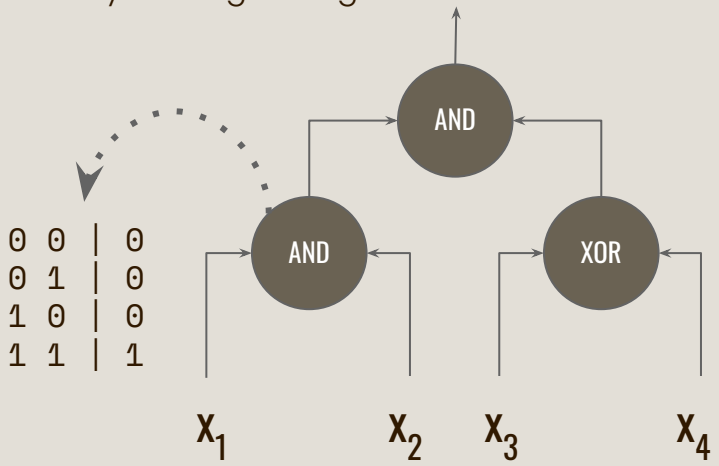
GARBLING SCHEMES

Gate-by-Gate garbling



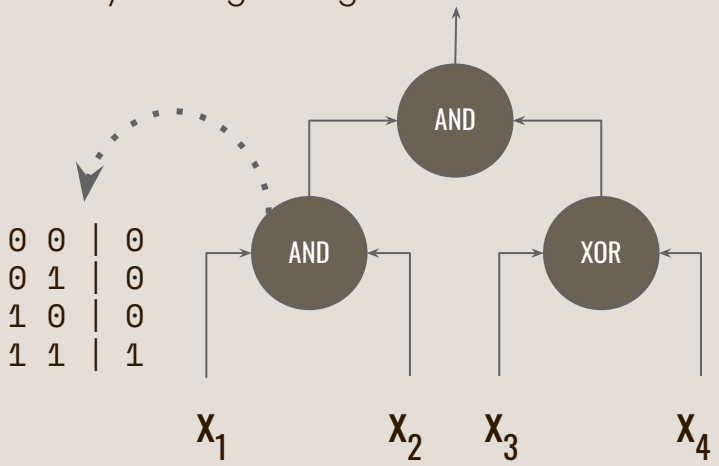
GARBLING SCHEMES

Gate-by-Gate garbling



GARBLING SCHEMES

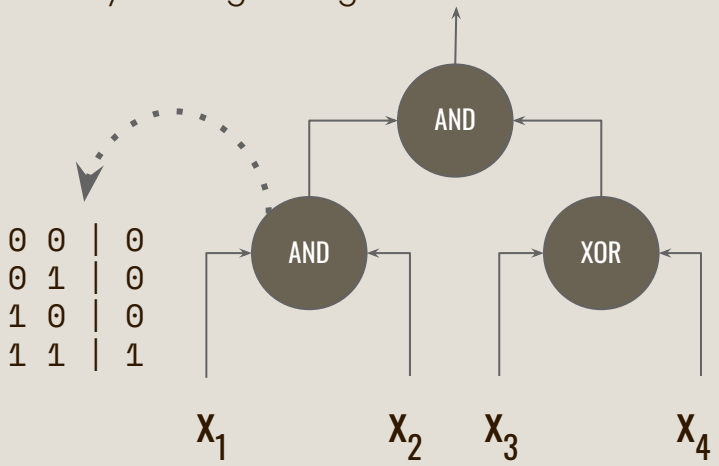
Gate-by-Gate garbling



$Enc(L_a^0 | L_b^0 ; L_c^0)$
$Enc(L_a^0 | L_b^1 ; L_c^0)$
$Enc(L_a^1 | L_b^0 ; L_c^0)$
$Enc(L_a^1 | L_b^1 ; L_c^1)$

GARBLING SCHEMES

Gate-by-Gate garbling



$Enc(L_a^0 | L_b^0 ; L_c^0)$
$Enc(L_a^0 | L_b^1 ; L_c^0)$
$Enc(L_a^1 | L_b^0 ; L_c^0)$
$Enc(L_a^1 | L_b^1 ; L_c^1)$

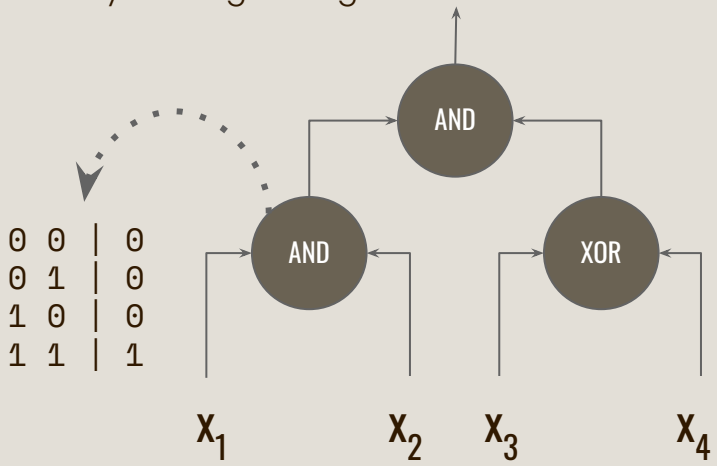
Free-XOR [KS08]

AND gate: 4K bits

XOR gate: 0 bits

GARBLING SCHEMES

Gate-by-Gate garbling



$Enc(L_a^0 | L_b^0 ; L_c^0)$
$Enc(L_a^0 | L_b^1 ; L_c^0)$
$Enc(L_a^1 | L_b^0 ; L_c^0)$
$Enc(L_a^1 | L_b^1 ; L_c^1)$

Free-XOR [KS08]

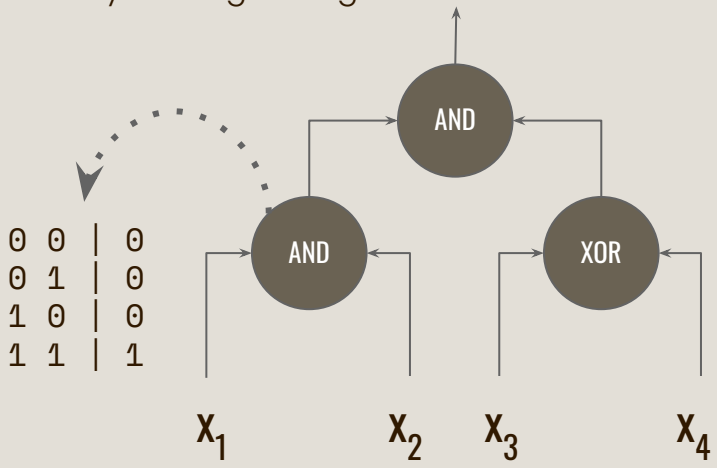
AND gate: 4K bits XOR gate: 0 bits

Two-Halves[ZRE15]

AND gate: 2K bits XOR gate: 0 bits

GARBLING SCHEMES

Gate-by-Gate garbling



$Enc(L_a^0 | L_b^0 ; L_c^0)$
$Enc(L_a^0 | L_b^1 ; L_c^0)$
$Enc(L_a^1 | L_b^0 ; L_c^0)$
$Enc(L_a^1 | L_b^1 ; L_c^1)$

Free-XOR [KS08]

AND gate: 4K bits XOR gate: 0 bits

Two-Halves[ZRE15]

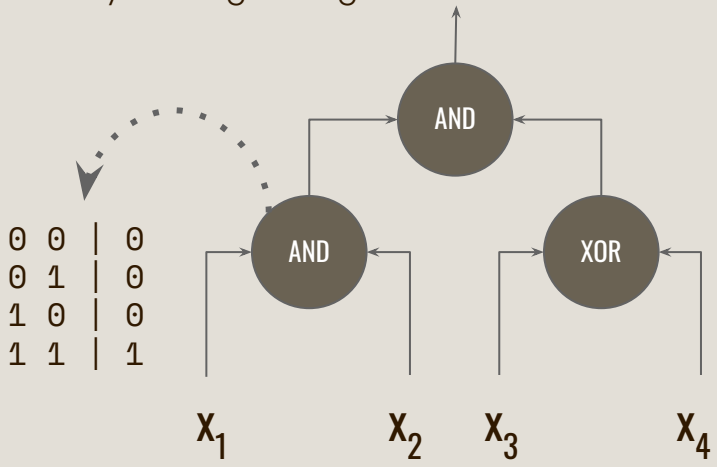
AND gate: 2K bits XOR gate: 0 bits

Three-Halves[RR21]

AND gate: 1.5K+5 bits XOR gate: 0 bits

GARBLING SCHEMES

Gate-by-Gate garbling



$\mathrm{Enc}(L_a^0 | L_b^0 ; L_c^0)$
$\mathrm{Enc}(L_a^0 | L_b^1 ; L_c^0)$
$\mathrm{Enc}(L_a^1 | L_b^0 ; L_c^0)$
$\mathrm{Enc}(L_a^1 | L_b^1 ; L_c^1)$

Free-XOR [KS08]

AND gate: 4K bits XOR gate: 0 bits

Two-Halves[ZRE15]

AND gate: 2K bits XOR gate: 0 bits

Three-Halves[RR21]

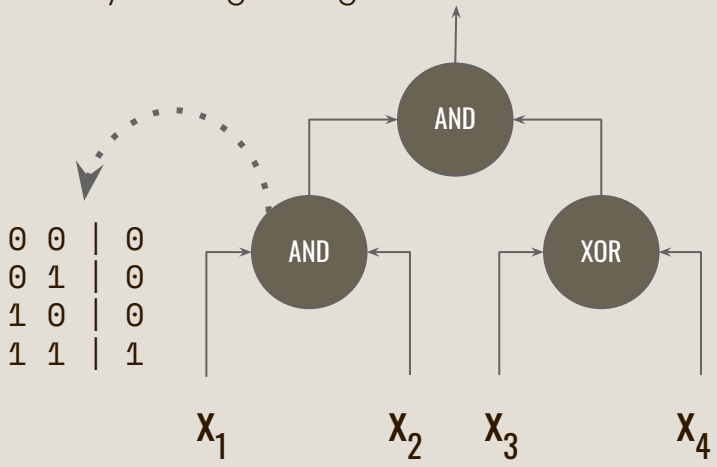
AND gate: 1.5K+5 bits XOR gate: 0 bits

Arithmetic Garbling [Heath24]

Binary gate(L-bit inputs): O(LK) bits

GARBLING SCHEMES

Gate-by-Gate garbling



$\mathrm{Enc}(L_a^0 | L_b^0 ; L_c^0)$
$\mathrm{Enc}(L_a^0 | L_b^1 ; L_c^0)$
$\mathrm{Enc}(L_a^1 | L_b^0 ; L_c^0)$
$\mathrm{Enc}(L_a^1 | L_b^1 ; L_c^1)$

Free-XOR [KS08]

AND gate: 4K bits XOR gate: 0 bits

Two-Halves[ZRE15]

AND gate: 2K bits XOR gate: 0 bits

Three-Halves[RR21]

AND gate: 1.5K+5 bits XOR gate: 0 bits

Arithmetic Garbling [Heath24]

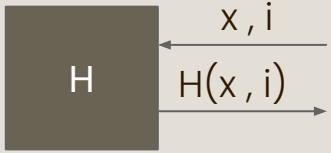
Binary gate(L-bit inputs): O(LK) bits

Requires Circular Security

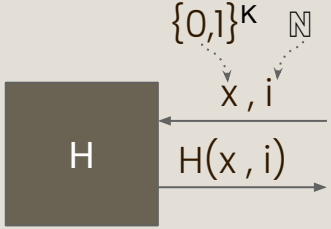
Circular Correlation Robustness

Circular Correlation Robustness

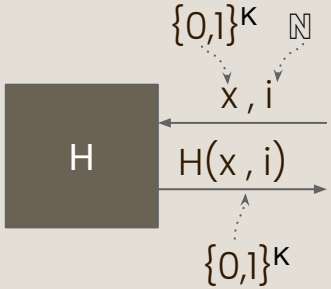
Circular Correlation Robustness



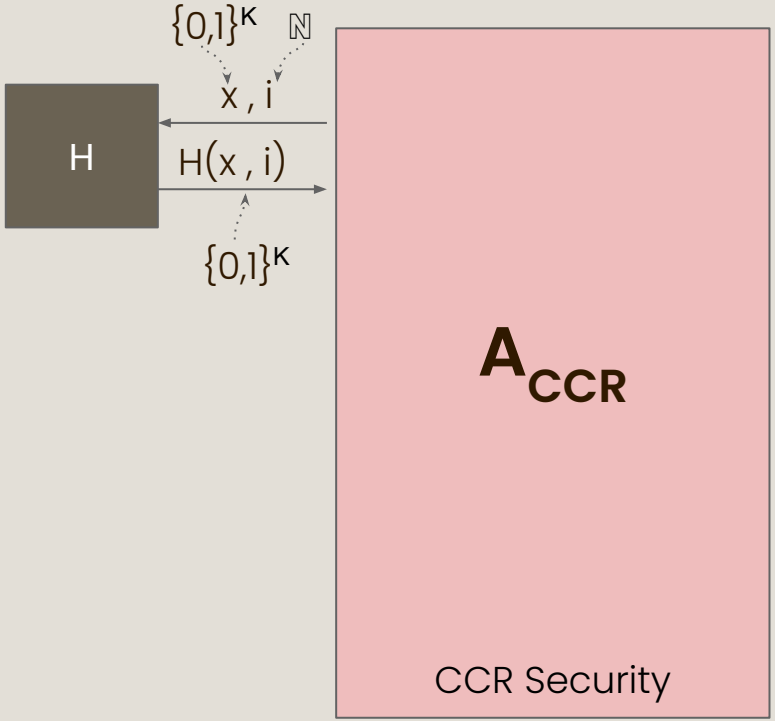
Circular Correlation Robustness



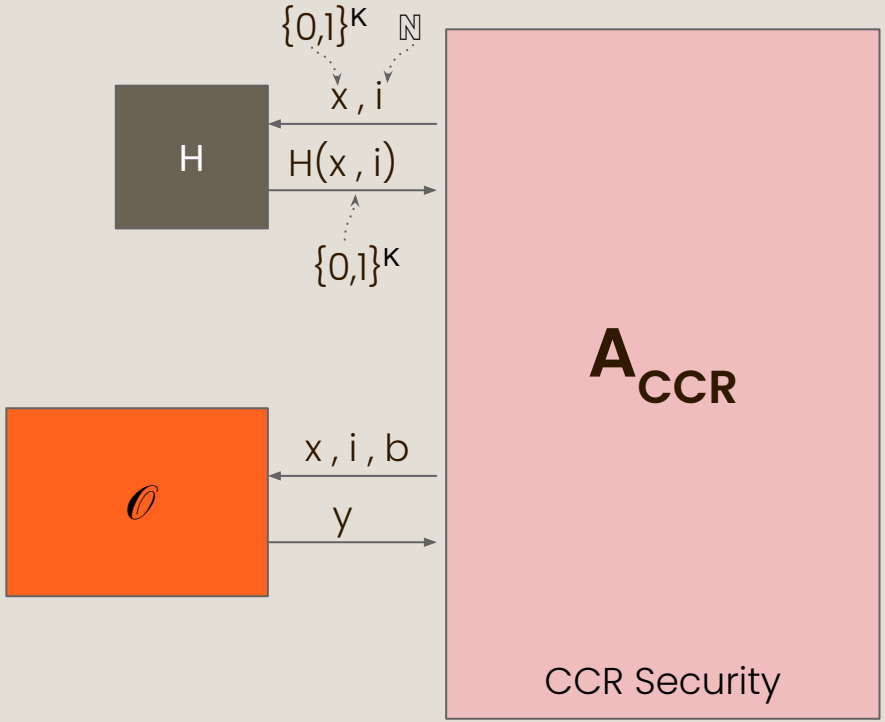
Circular Correlation Robustness



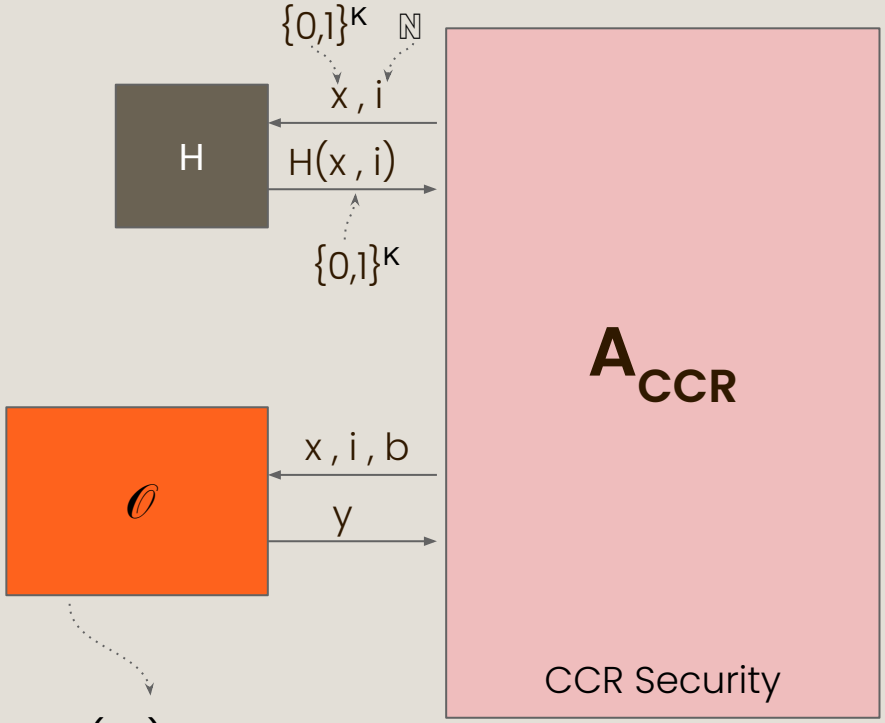
Circular Correlation Robustness



Circular Correlation Robustness



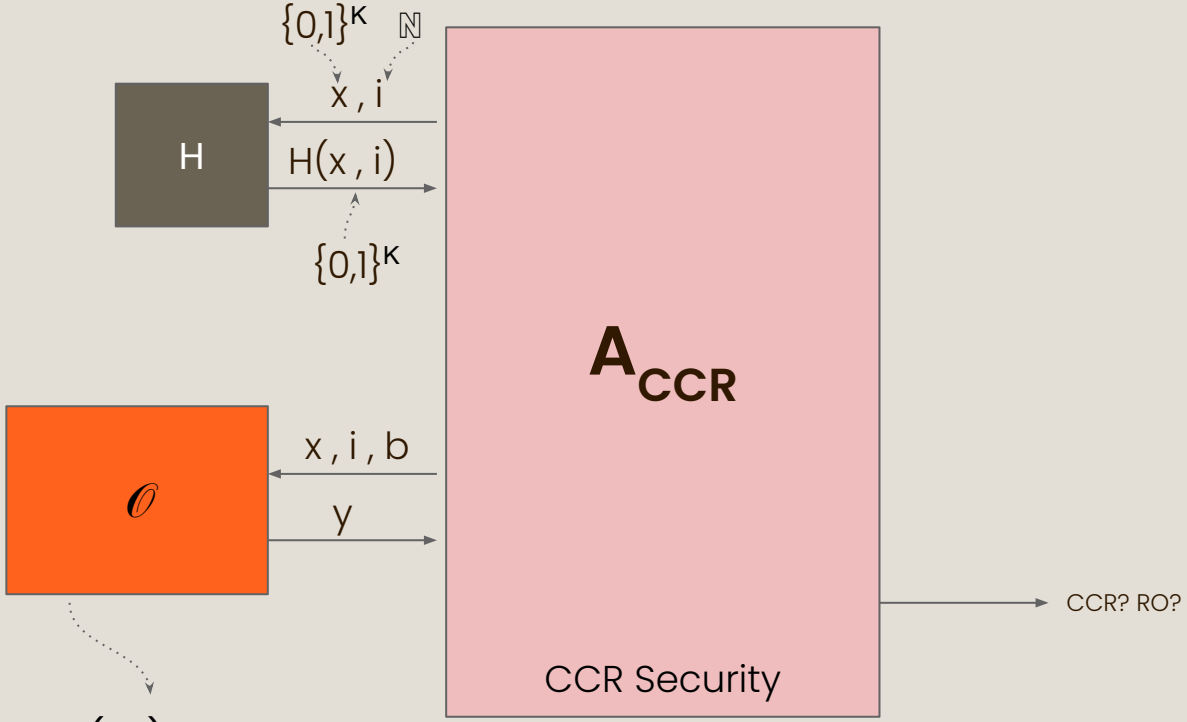
Circular Correlation Robustness



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$

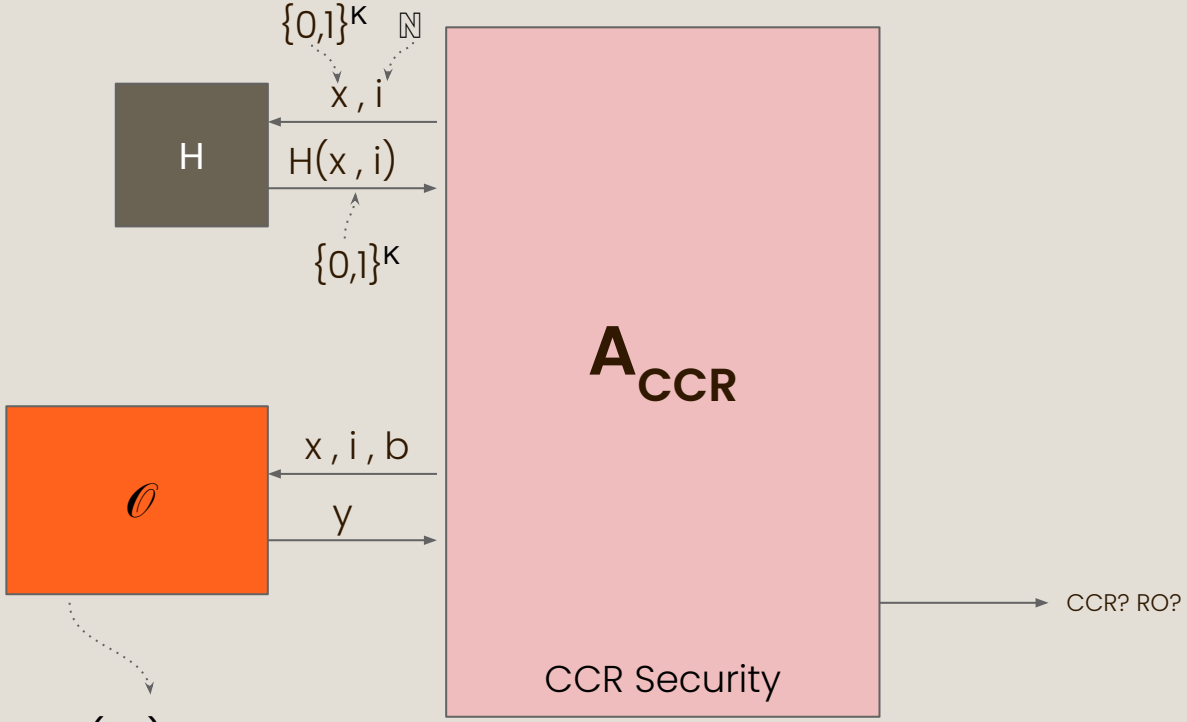
Circular Correlation Robustness



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$

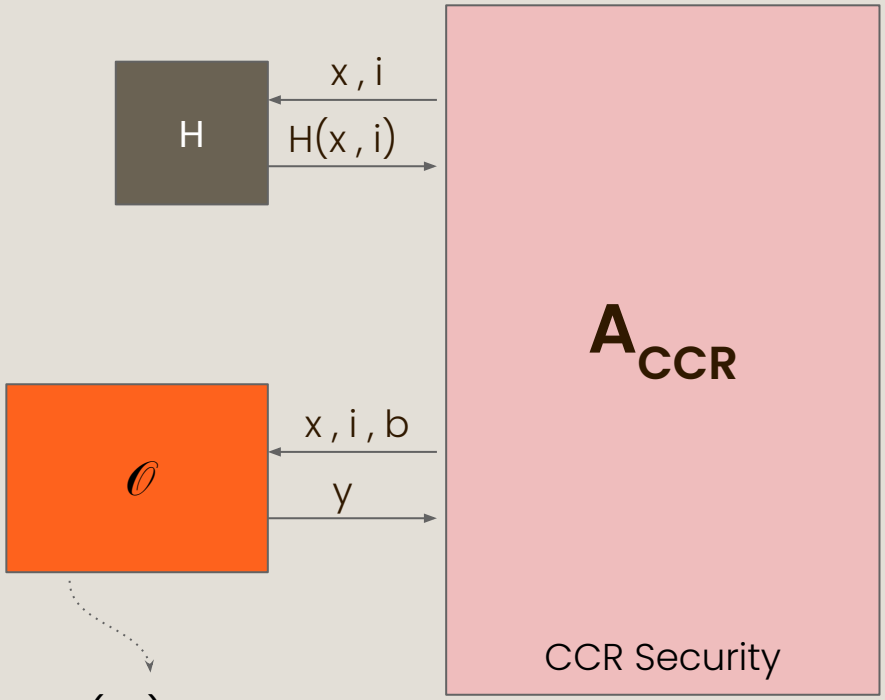
Circular Correlation Robustness



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$ (legal queries)

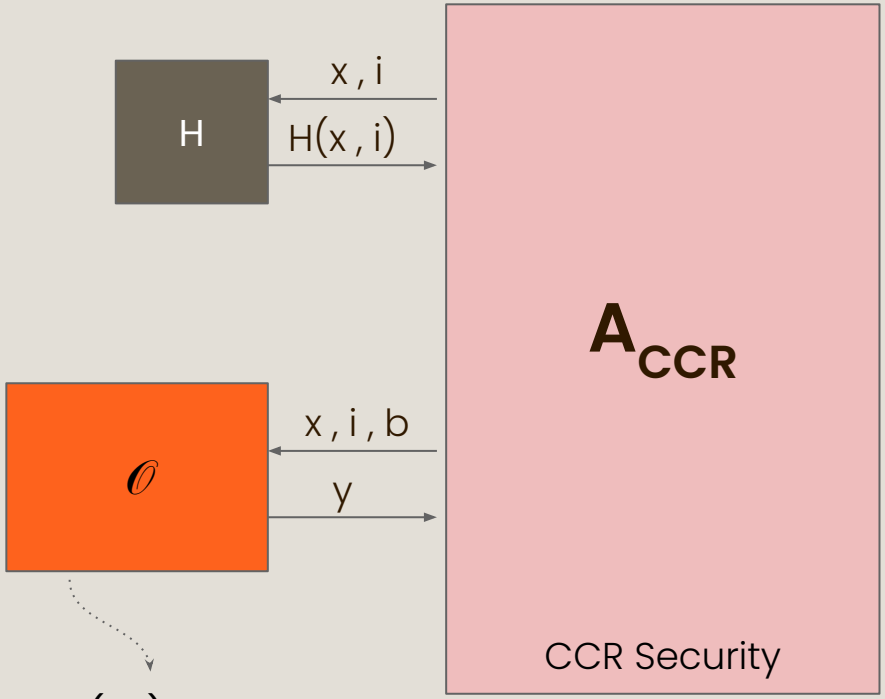
Circular Correlation Robustness



$\mathcal{O}_{RO}(x,i) \rightarrow y$

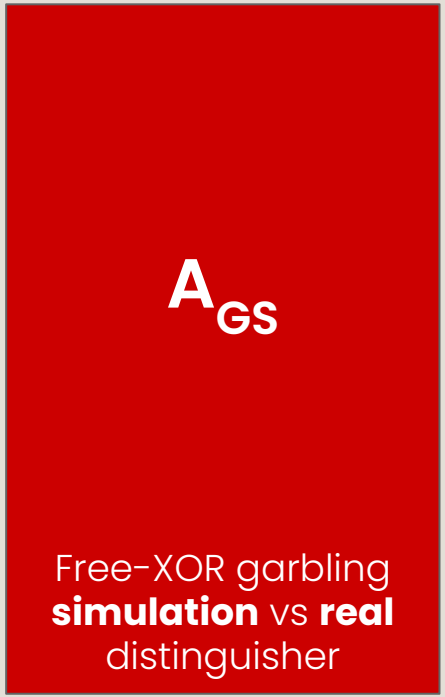
$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$ (legal queries)

Circular Correlation Robustness

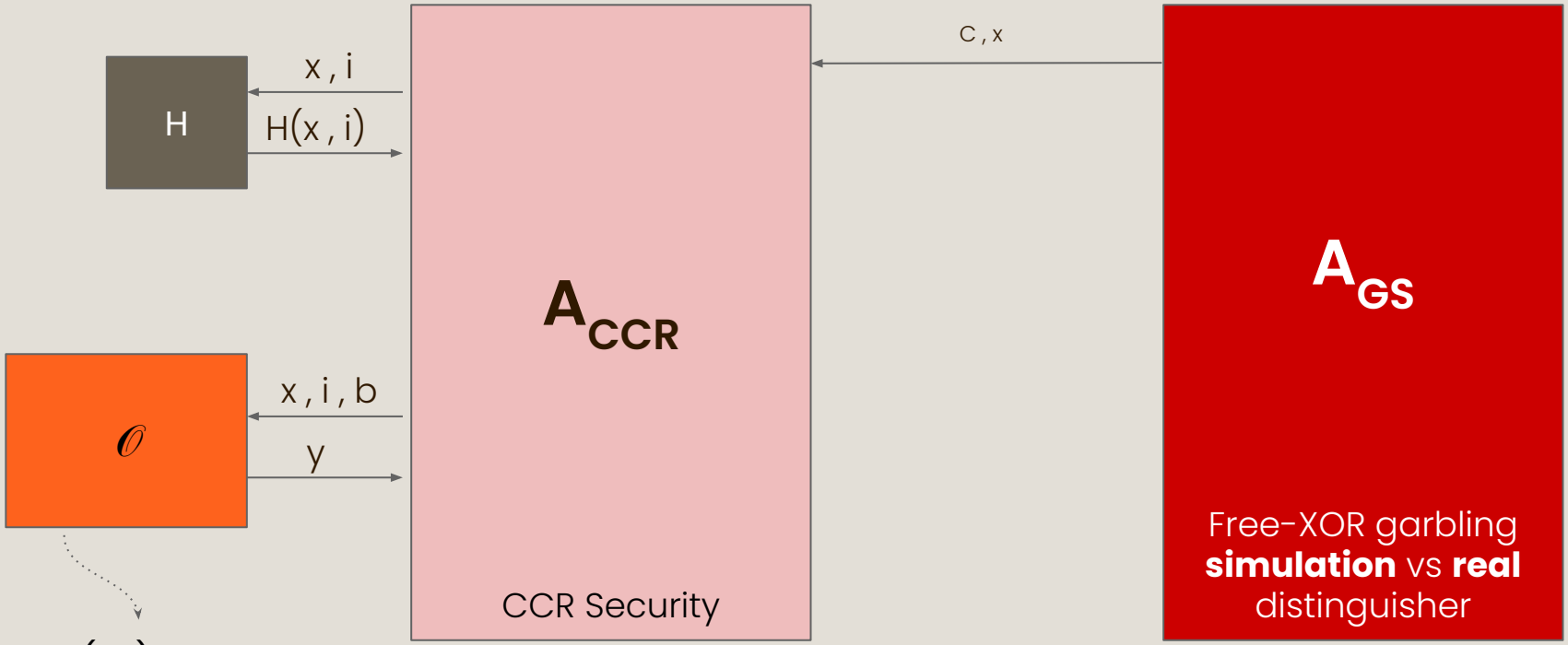


$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$ (legal queries)



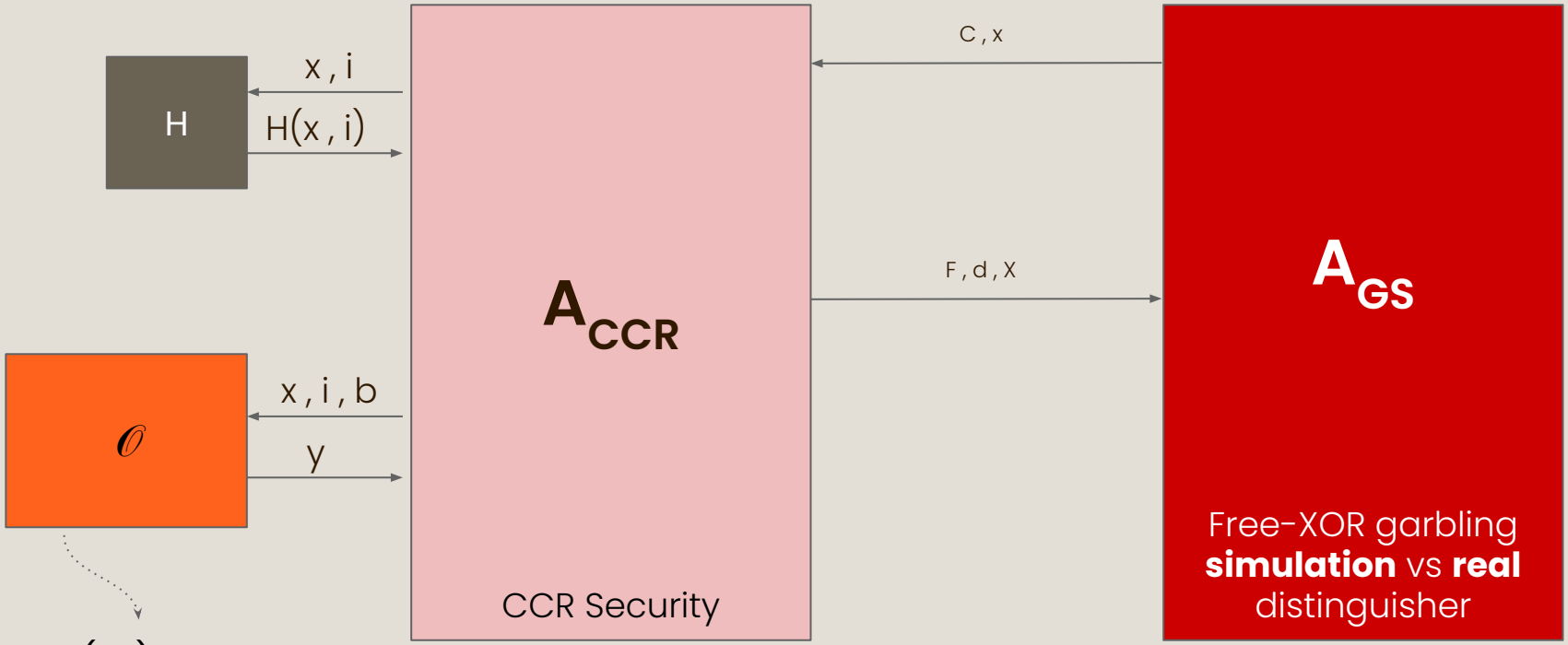
Circular Correlation Robustness



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$ (legal queries)

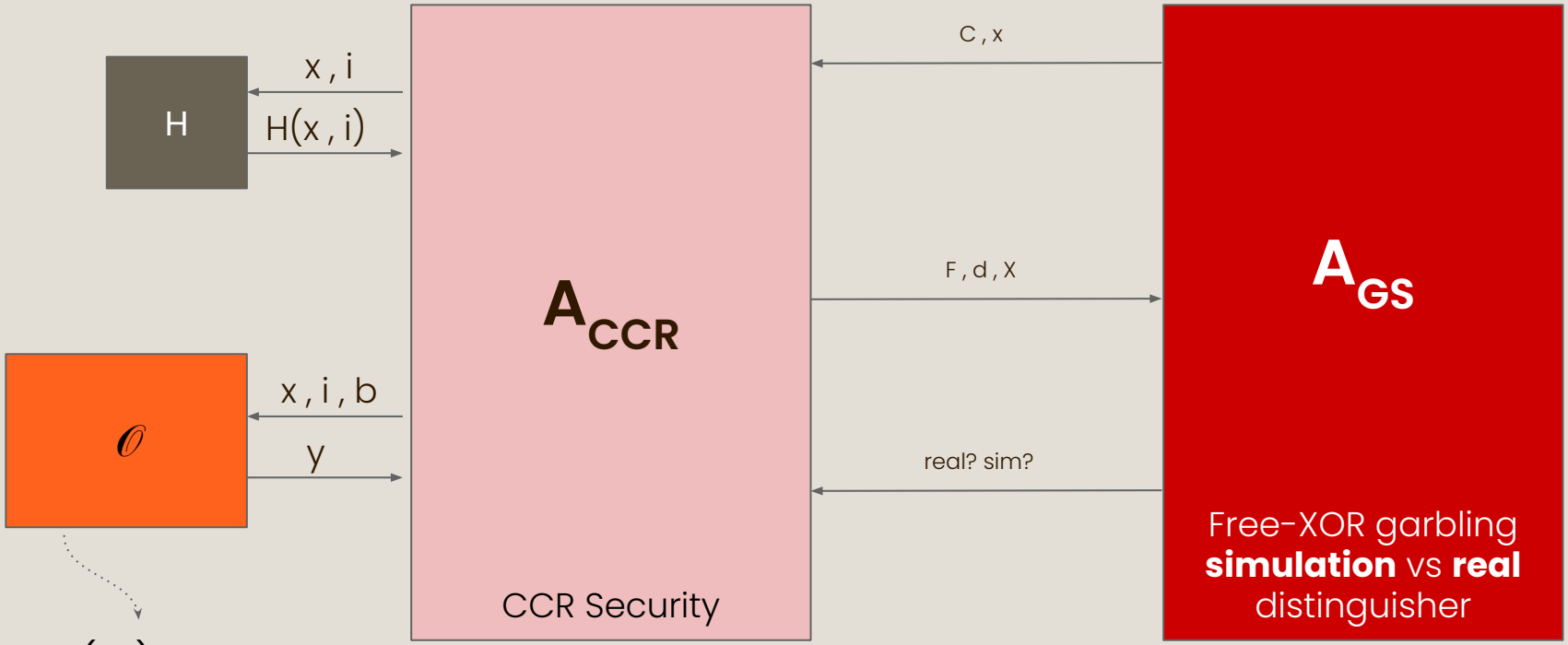
Circular Correlation Robustness



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$ (legal queries)

Circular Correlation Robustness

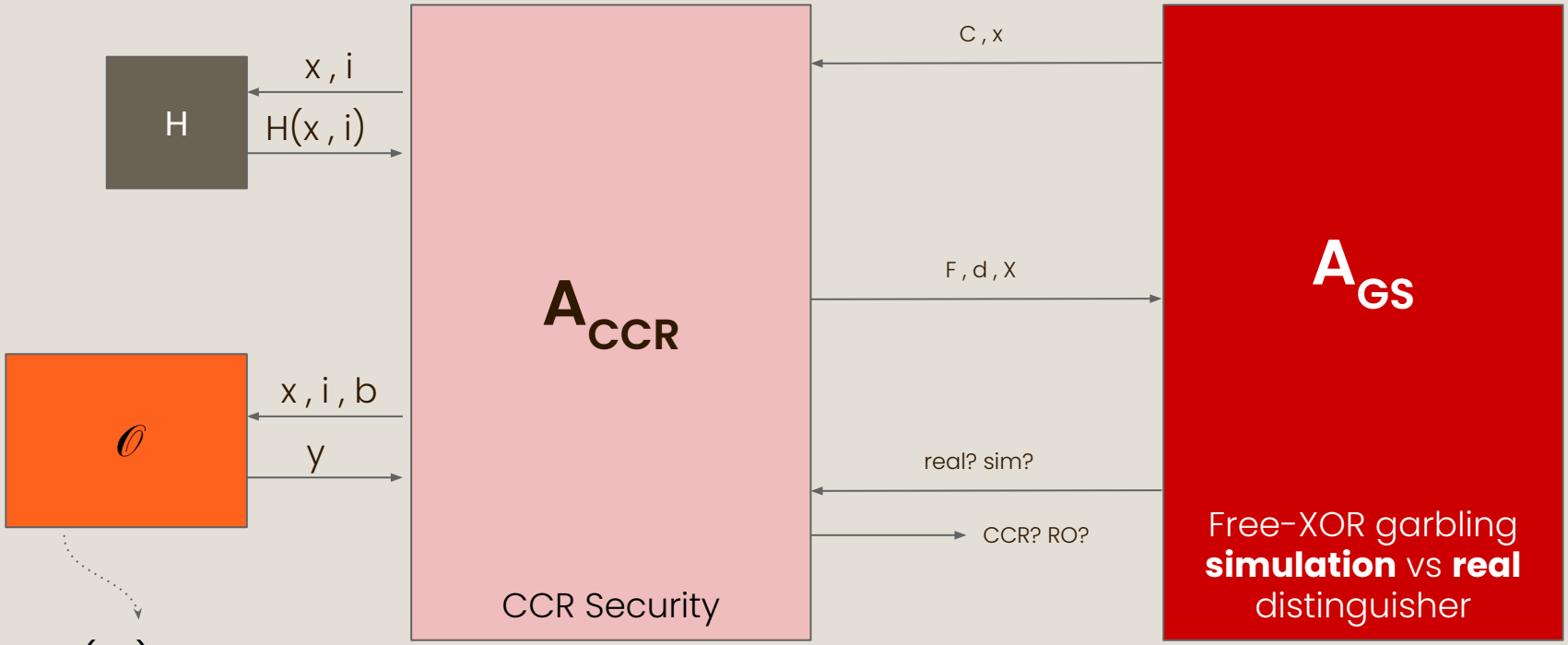


$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$

(legal queries)

Circular Correlation Robustness



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$

(legal queries)

CONTRIBUTIONS

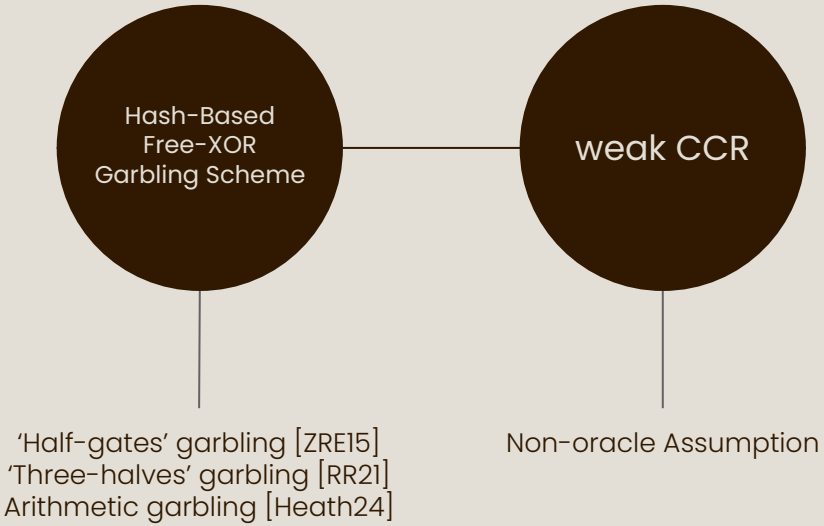
Instantiating **hash-based free-XOR** garbling in the **plain model**

CONTRIBUTIONS

Instantiating **hash-based free-XOR** garbling in the **plain model**

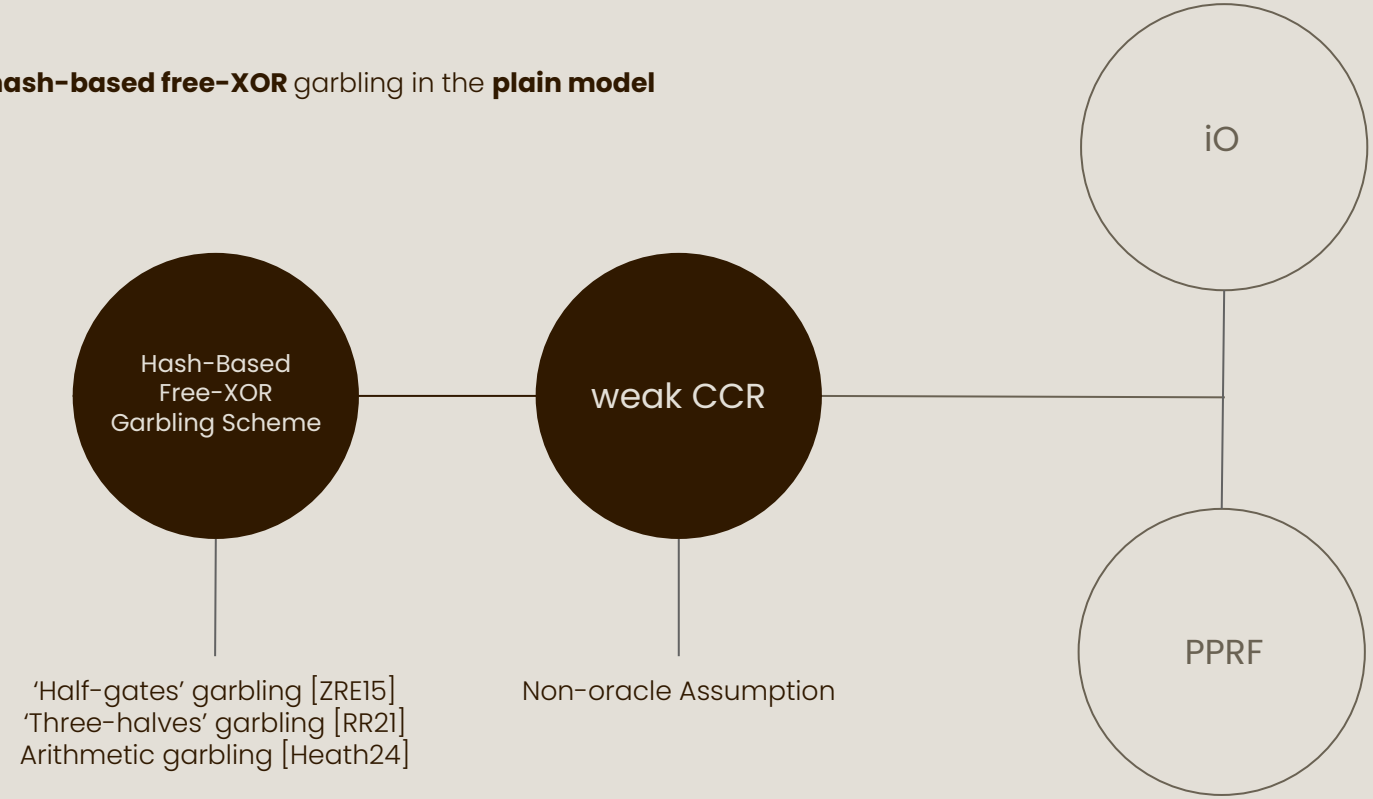
CONTRIBUTIONS

Instantiating **hash-based free-XOR** garbling in the **plain model**

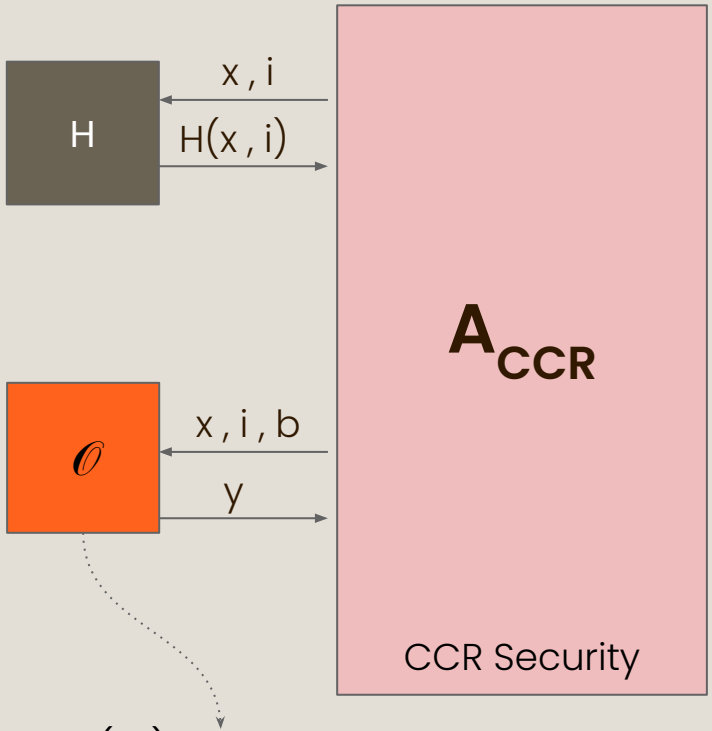


CONTRIBUTIONS

Instantiating **hash-based free-XOR** garbling in the **plain model**



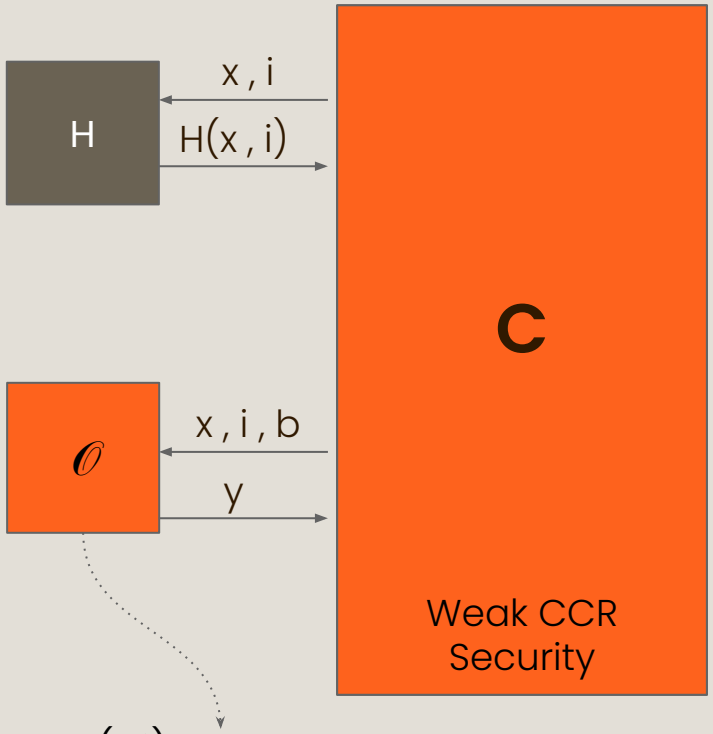
Weak CCR



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$

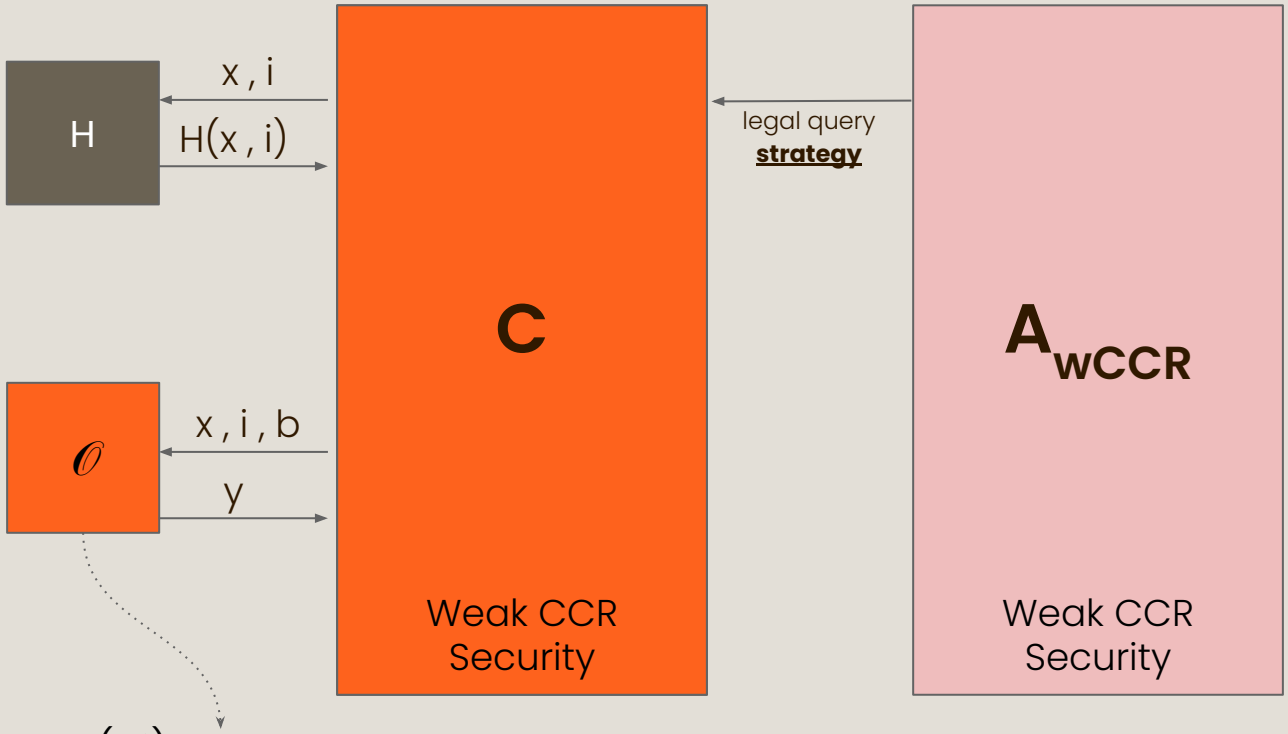
Weak CCR



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$

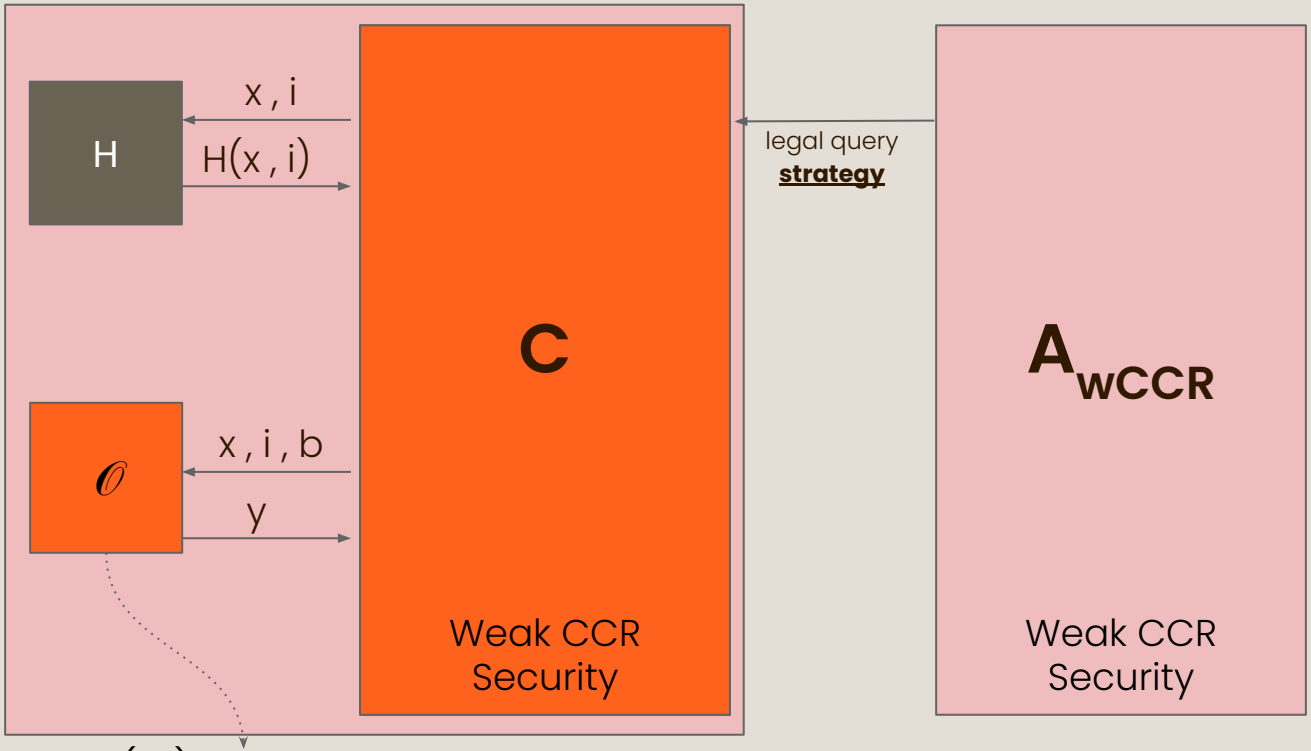
Weak CCR



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$

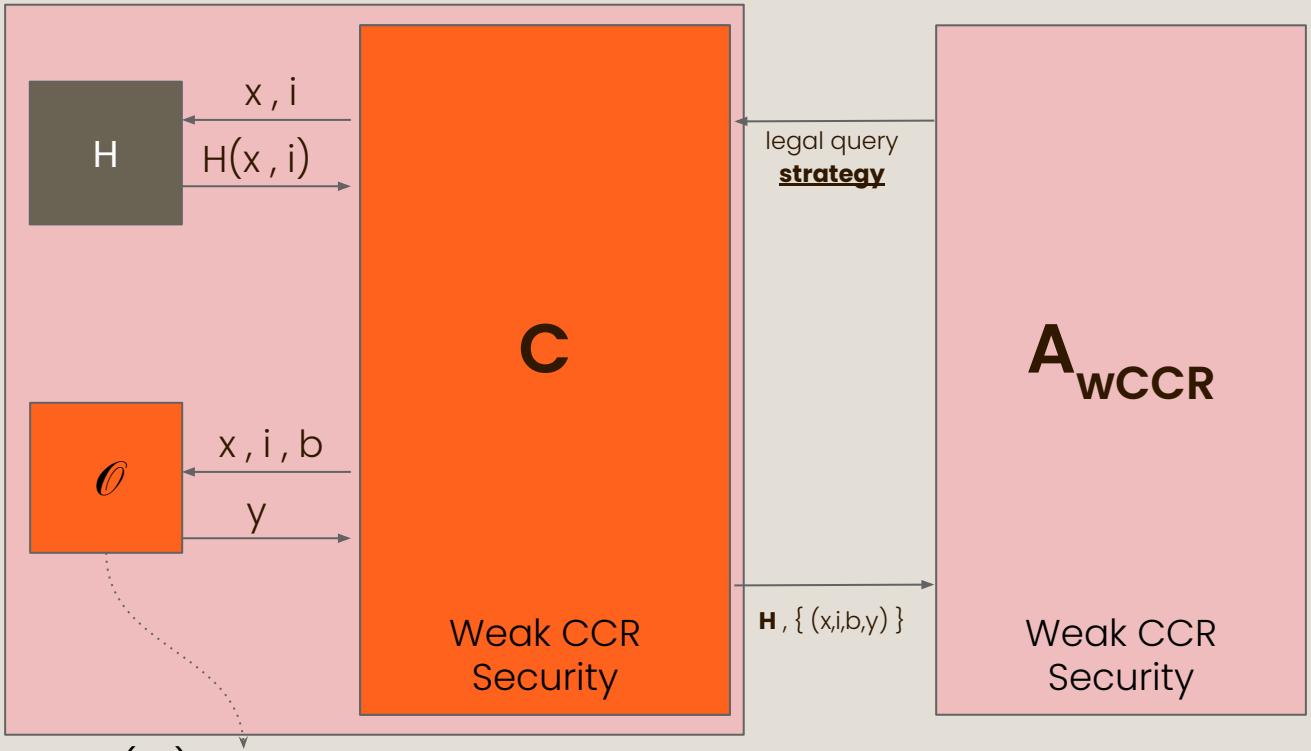
Weak CCR



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$

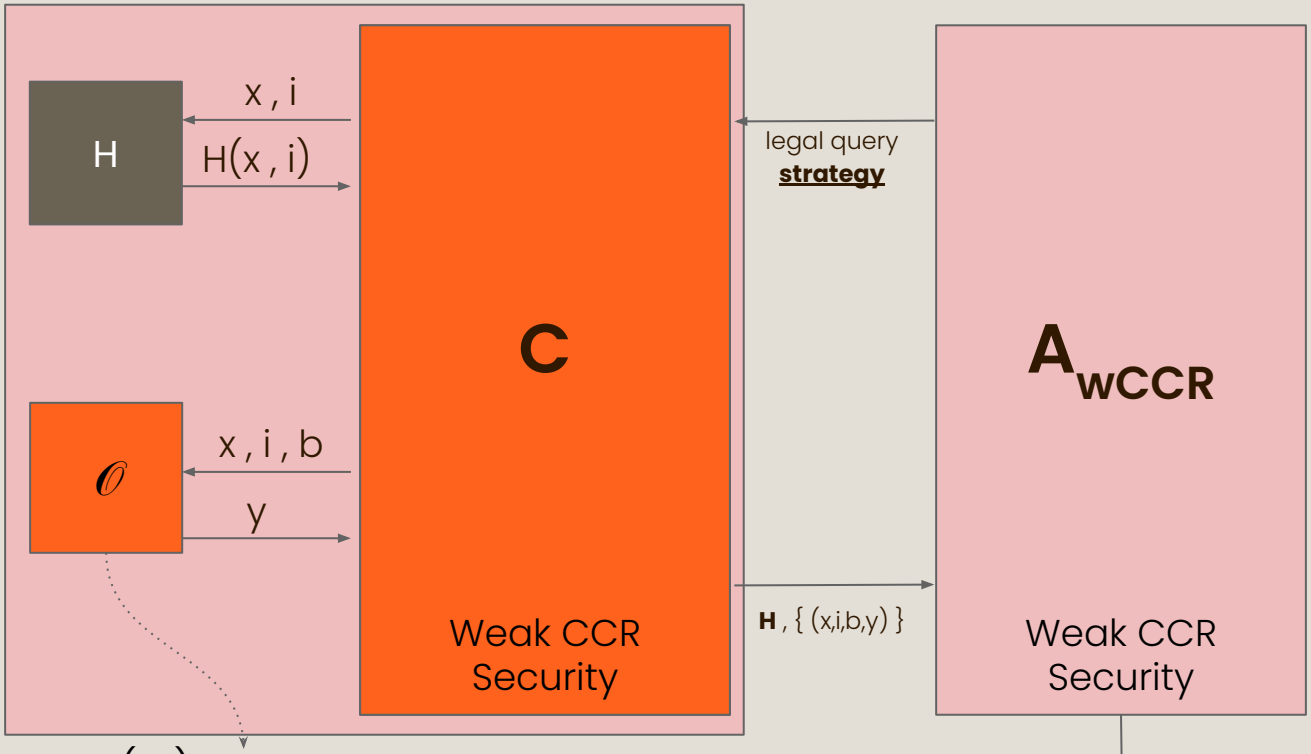
Weak CCR



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$

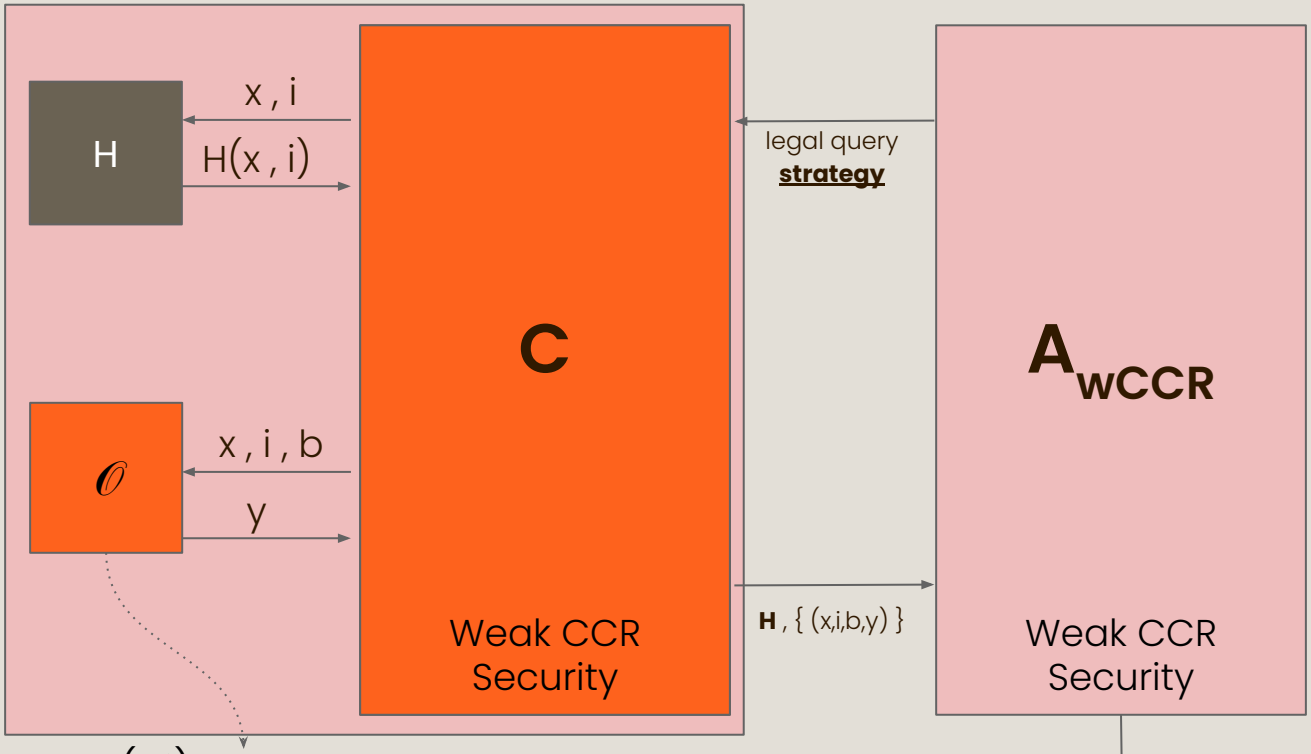
Weak CCR



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$

Weak CCR



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$

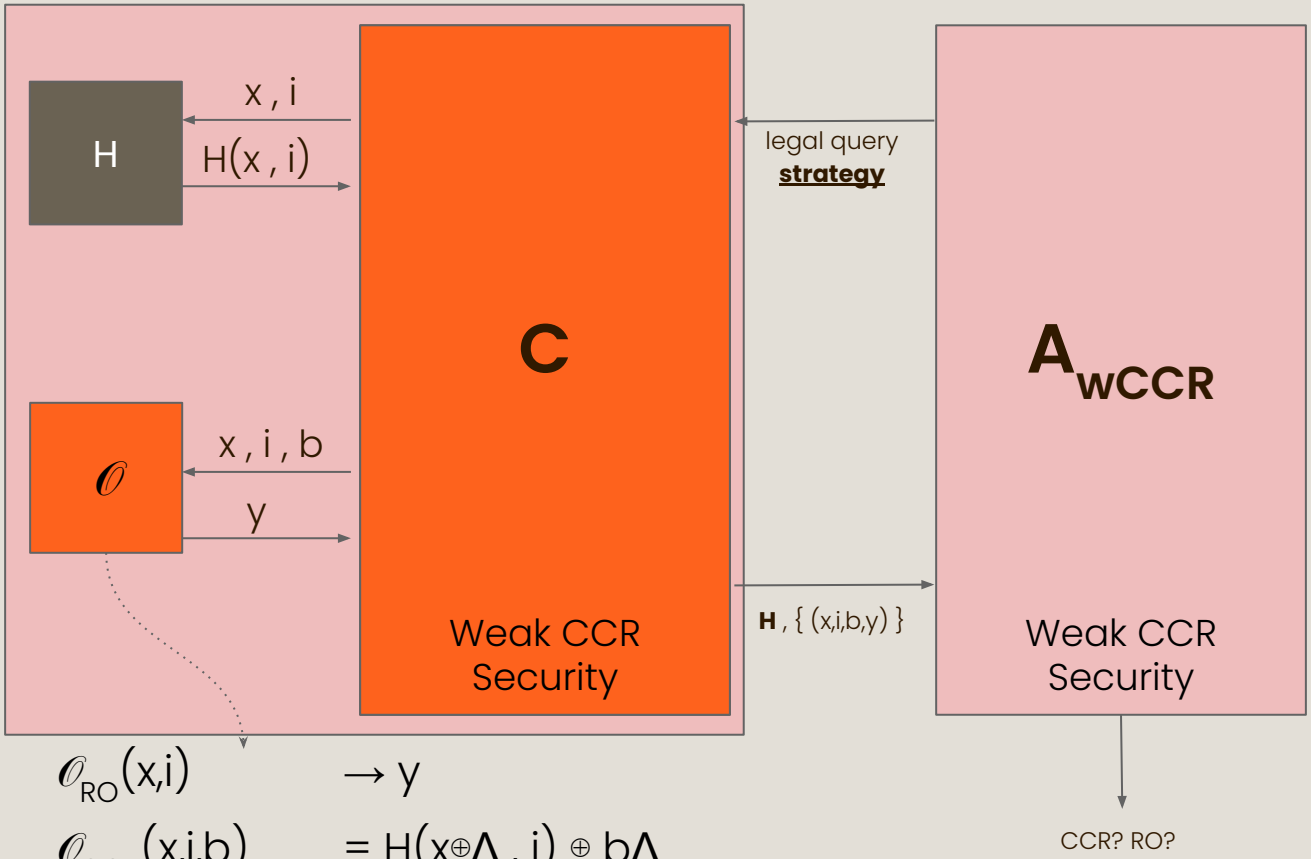
Difference from CCR

non-oracle assumption

non-black-box in query strategy

A receives H

Weak CCR



Difference from CCR

non-oracle assumption

non-black-box in query strategy

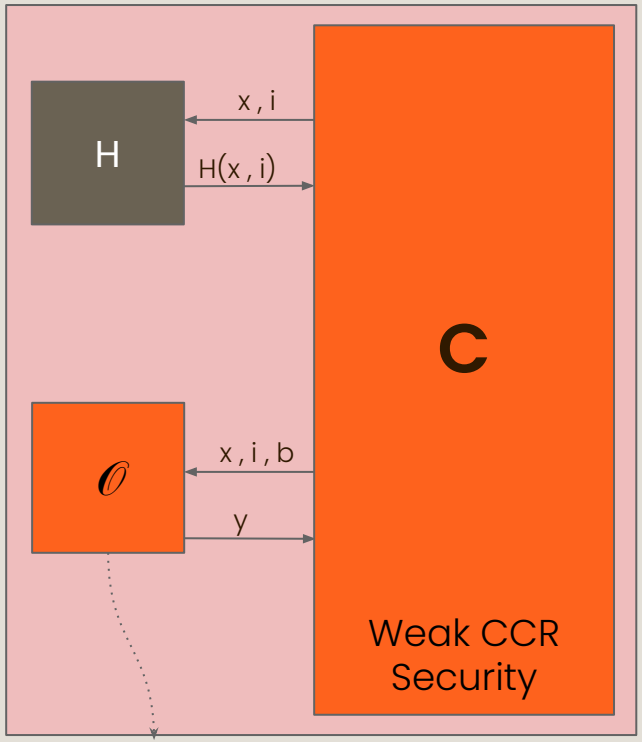
A receives H

Difference from RTCCR

|output| = |input|

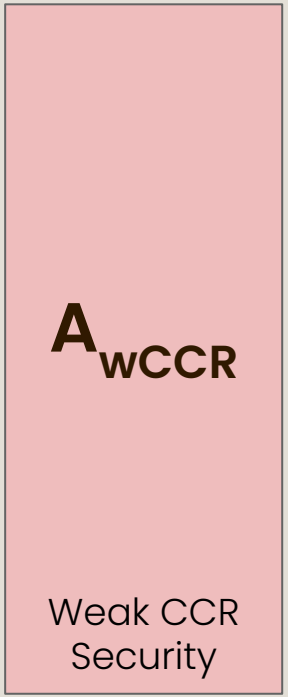
linear combination defined by b

Weak CCR

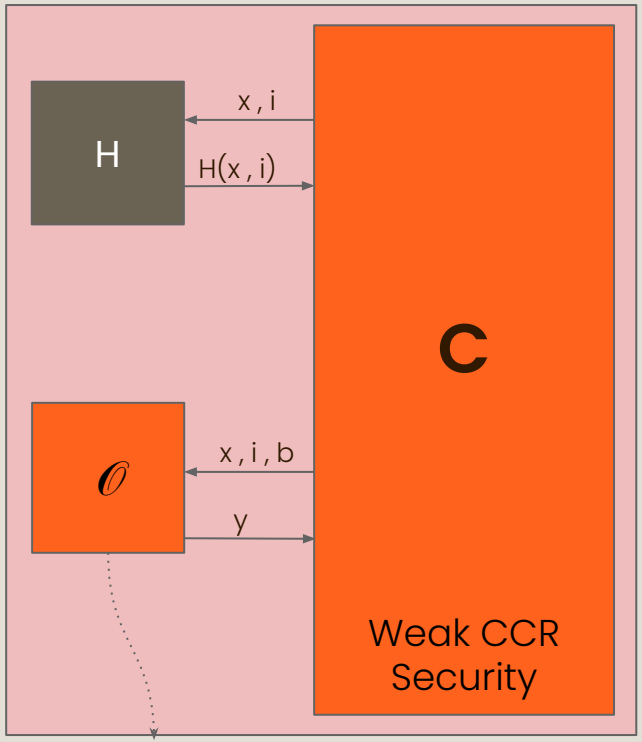


$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$



Weak CCR



$\mathcal{O}_{RO}(x,i) \rightarrow y$

$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$

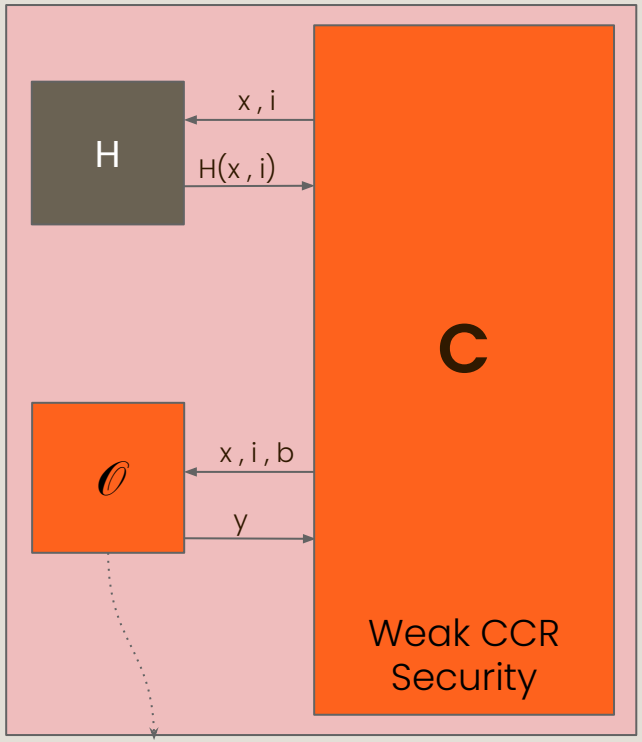
$\mathbf{A}_{wCCR}$

Weak CCR Security

$\mathbf{A}_{GS}$

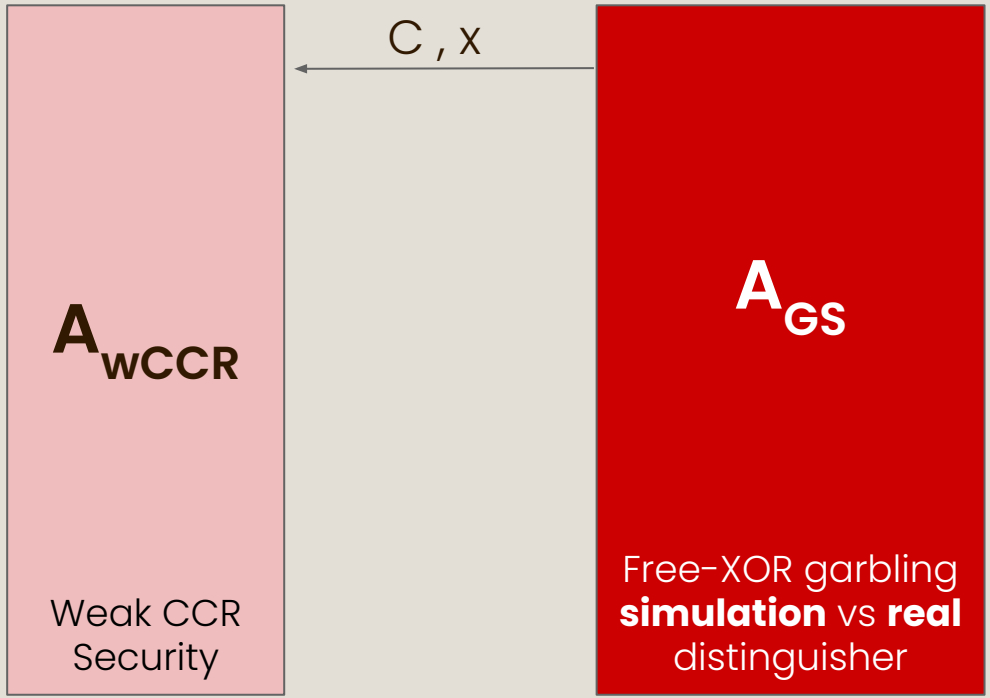
Free-XOR garbling **simulation** vs **real** distinguisher

Weak CCR

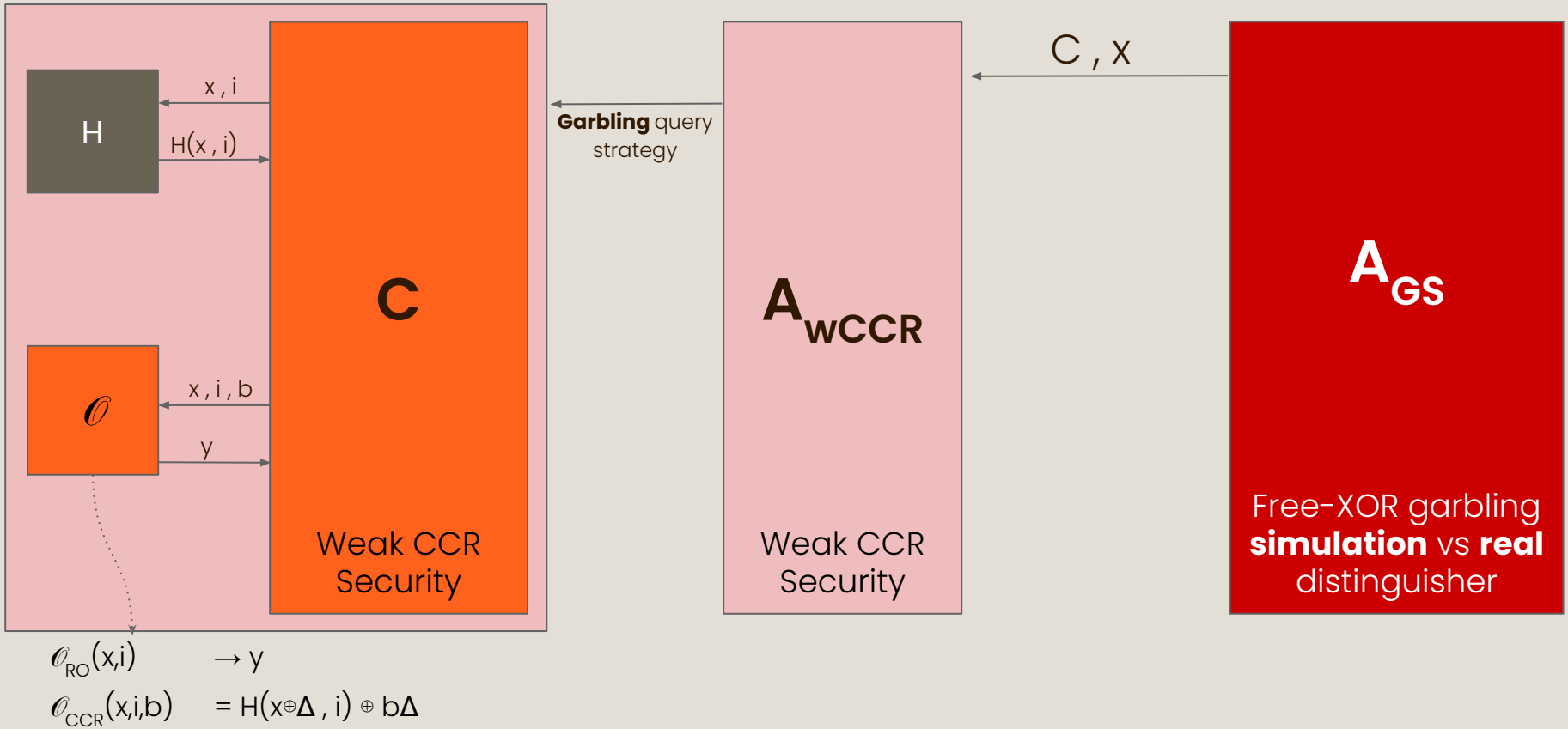


$\mathcal{O}_{RO}(x,i) \rightarrow y$

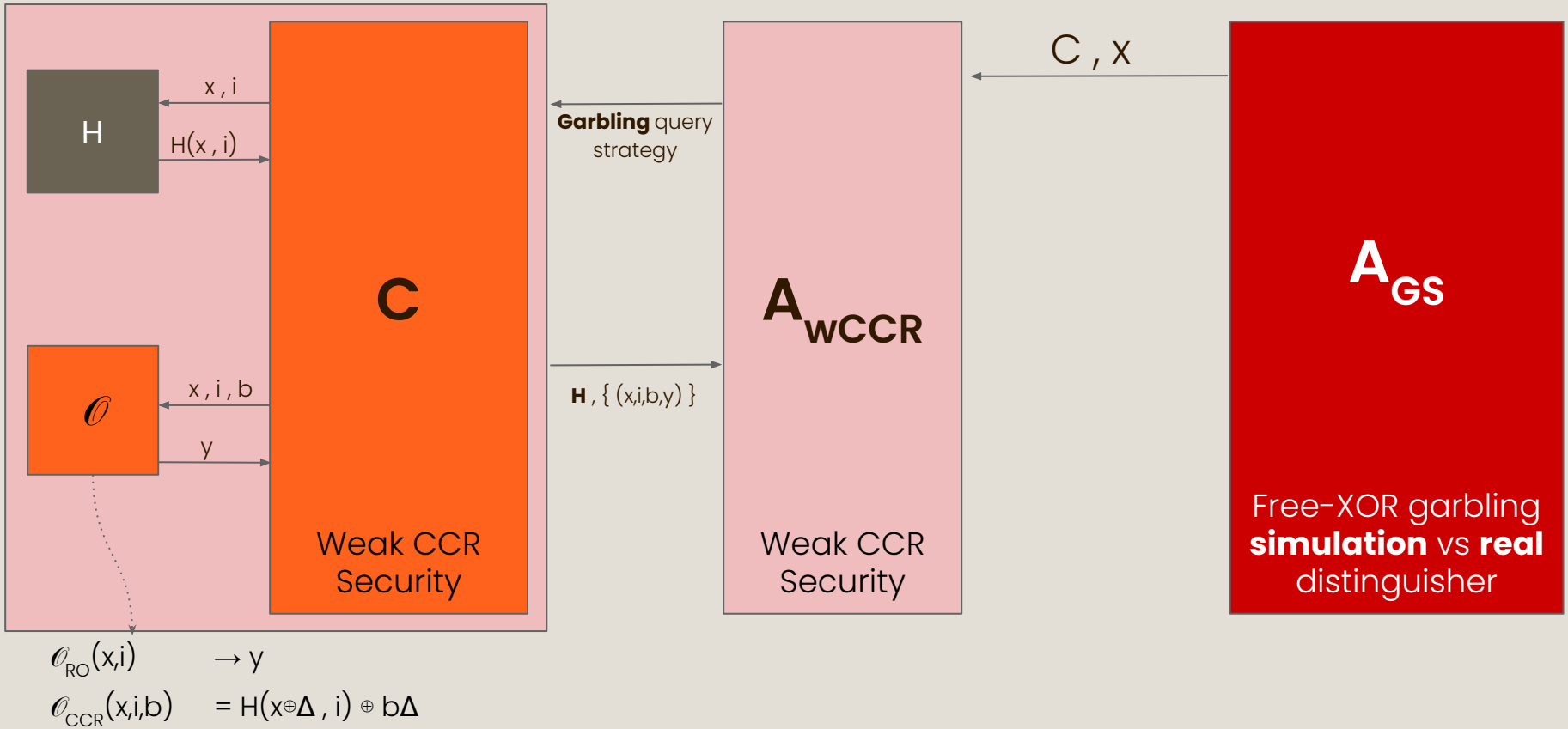
$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$



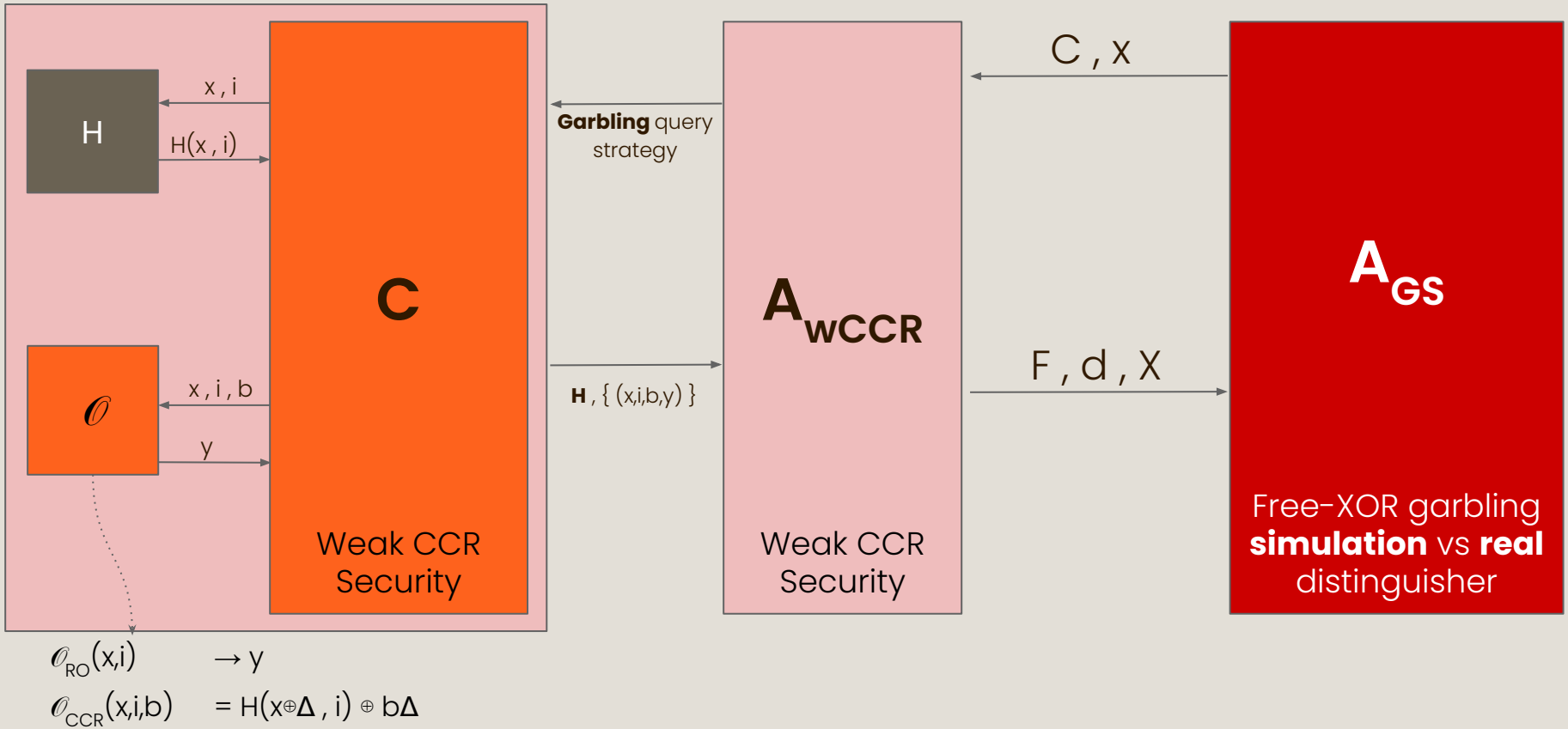
Weak CCR


H
x , i
H(x , i)
C
𝒪
x , i , b
y
Weak CCR Security
Garbling query strategy
A_wCCR
Weak CCR Security
C , x
A_GS
Free-XOR garbling **simulation** vs **real** distinguisher
$\mathcal{O}_{RO}(x,i) \rightarrow y$
$\mathcal{O}_{CCR}(x,i,b) = H(x \oplus \Delta , i) \oplus b\Delta$

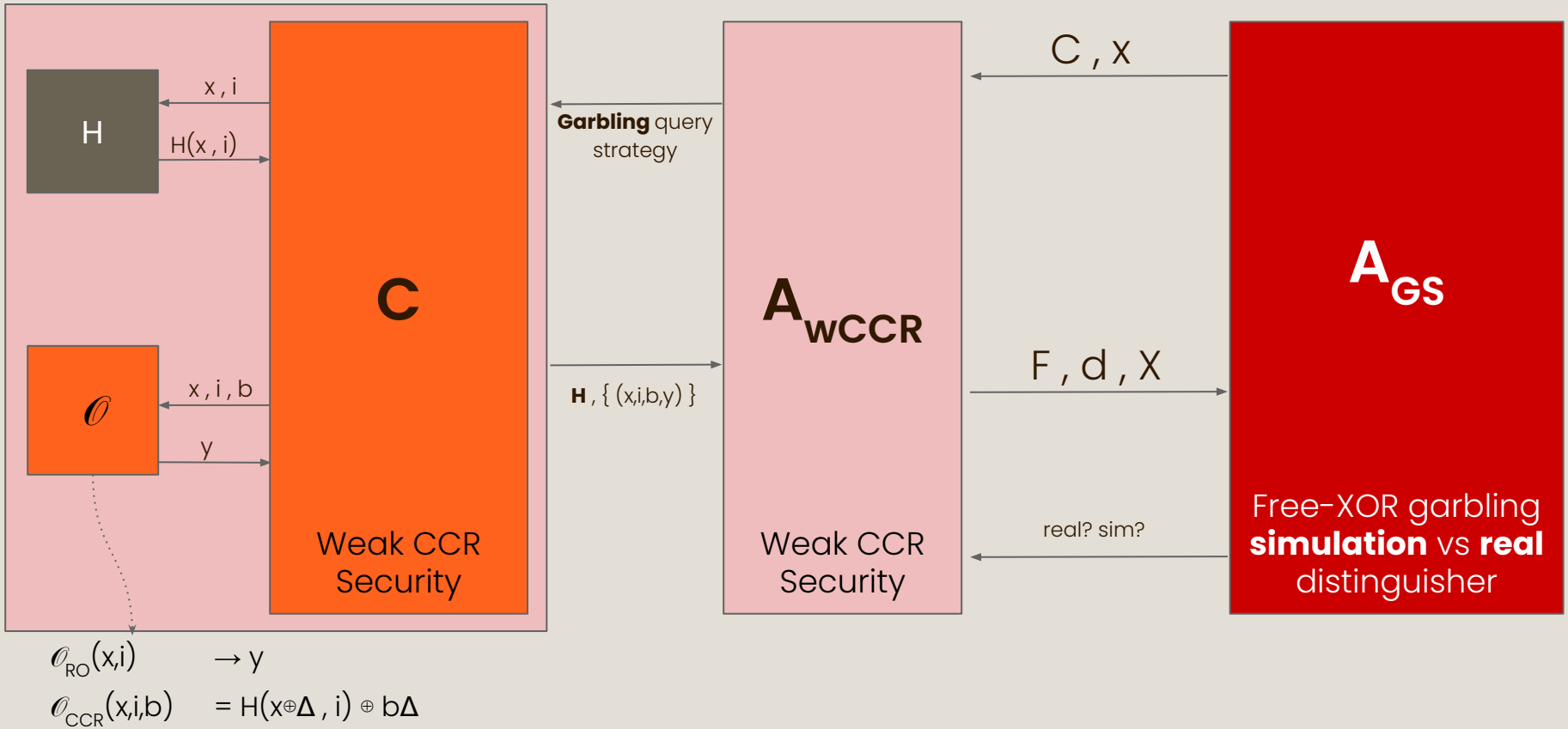
Weak CCR


H
x , i
H(x , i)
C
𝒪
x , i , b
y
Weak CCR Security
Garbling query strategy
H , { (x,i,b,y) }
A wCCR
Weak CCR Security
C , x
A GS
Free-XOR garbling simulation vs real distinguisher
𝒪RO(x,i) → y
𝒪CCR(x,i,b) = H(x⊕Δ , i) ⊕ bΔ

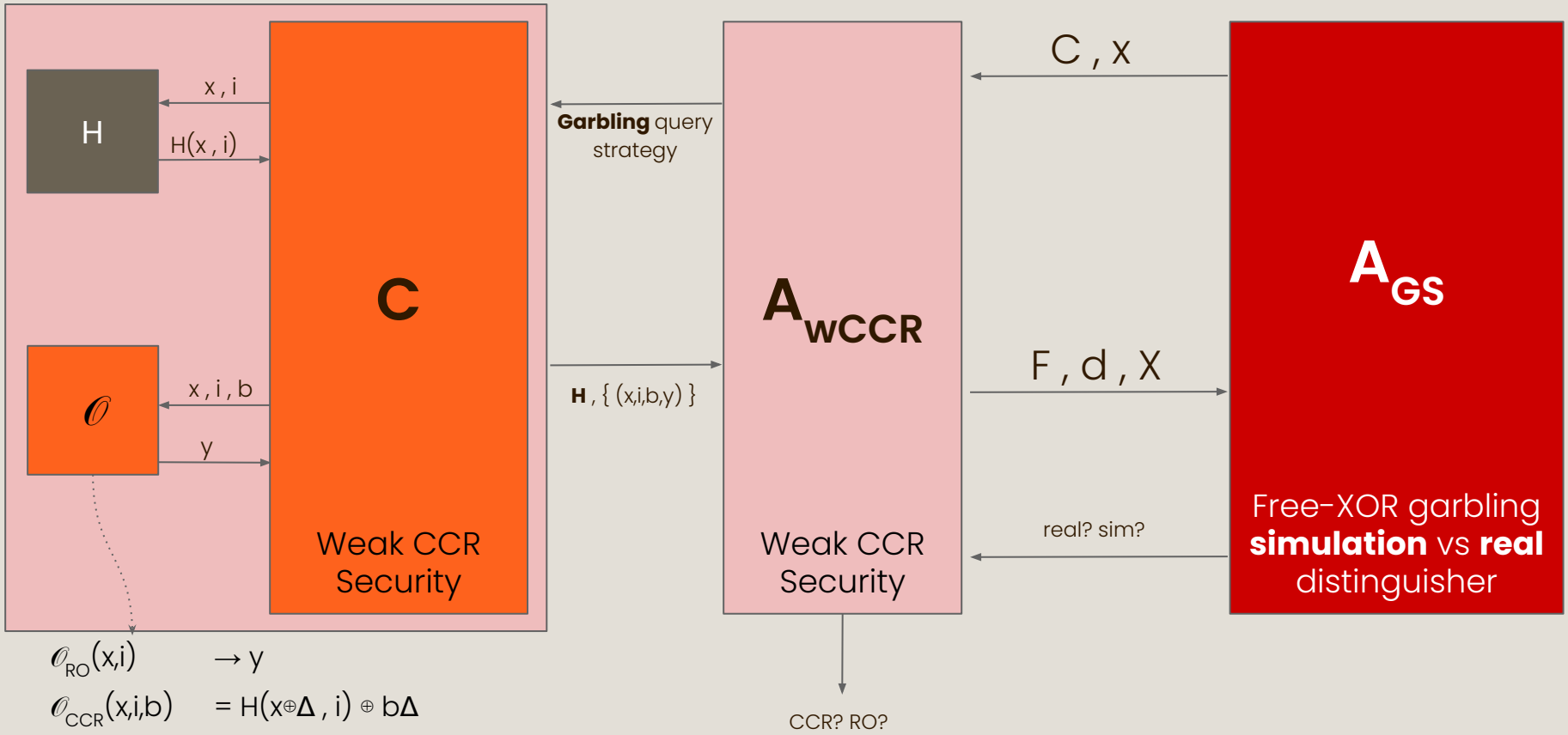
Weak CCR


H
x , i
H(x , i)
C
𝒪
x , i , b
y
Weak CCR Security
Garbling query strategy
H , { (x,i,b,y) }
A wCCR
Weak CCR Security
C , x
F , d , X
A GS
Free-XOR garbling simulation vs real distinguisher
𝒪RO(x,i) → y
𝒪CCR(x,i,b) = H(x⊕Δ , i) ⊕ bΔ

Weak CCR

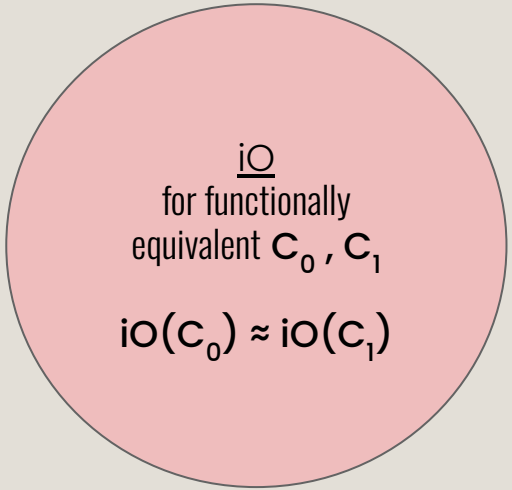

H
x , i
H(x , i)
C
𝒪
x , i , b
y
Weak CCR Security
Garbling query strategy
H , { (x,i,b,y) }
A_wCCR
Weak CCR Security
C , x
F , d , X
real? sim?
A_GS
Free-XOR garbling simulation vs real distinguisher
𝒪_RO(x,i) → y
𝒪_CCR(x,i,b) = H(x⊕Δ , i) ⊕ bΔ

Weak CCR

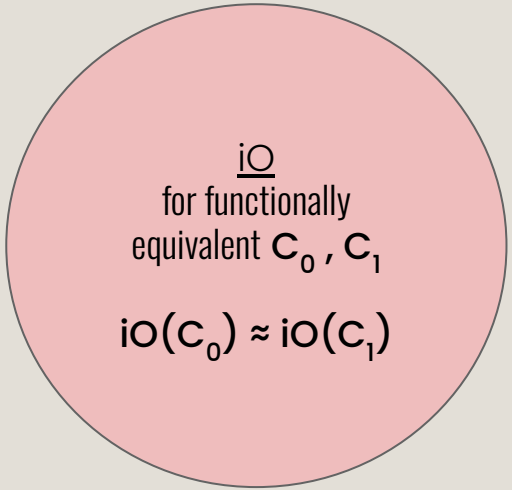

H
x , i
H(x , i)
C
$\mathcal{O}$
x , i , b
y
Weak CCR Security
$\mathcal{O}_{RO}$(x,i) → y
$\mathcal{O}_{CCR}$(x,i,b) = H(x⊕Δ , i) ⊕ bΔ
Garbling query strategy
H , { (x,i,b,y) }
A_{wCCR}
Weak CCR Security
CCR? RO?
C , x
F , d , X
real? sim?
A_{GS}
Free-XOR garbling
simulation vs real
distinguisher

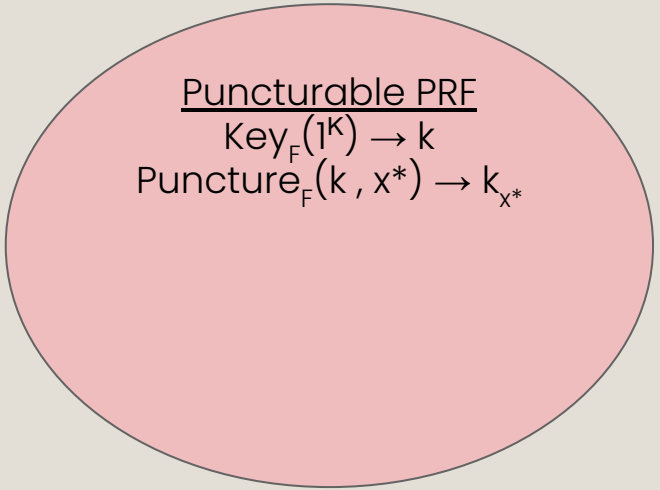
CONSTRUCTION

CONSTRUCTION

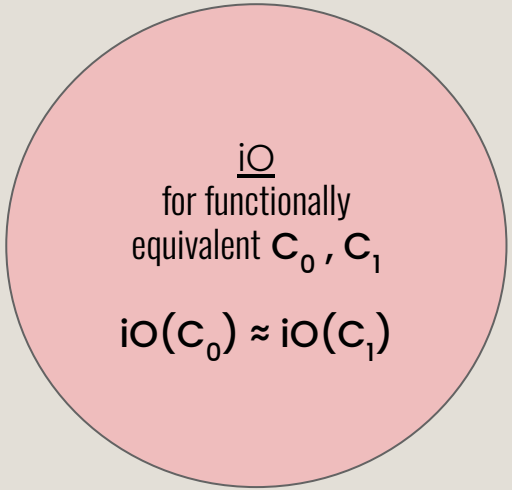


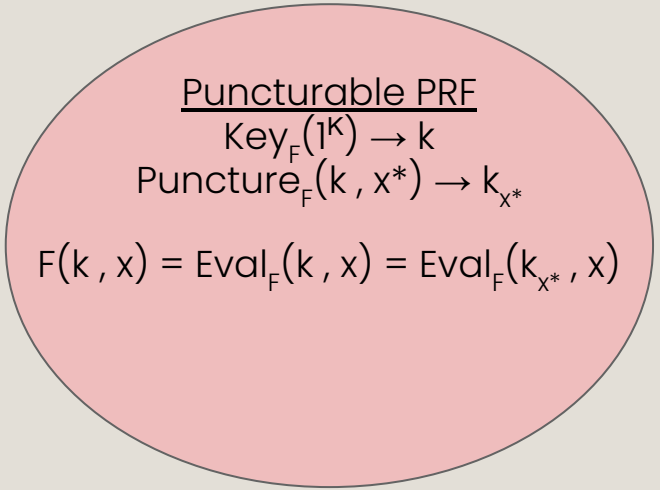
CONSTRUCTION



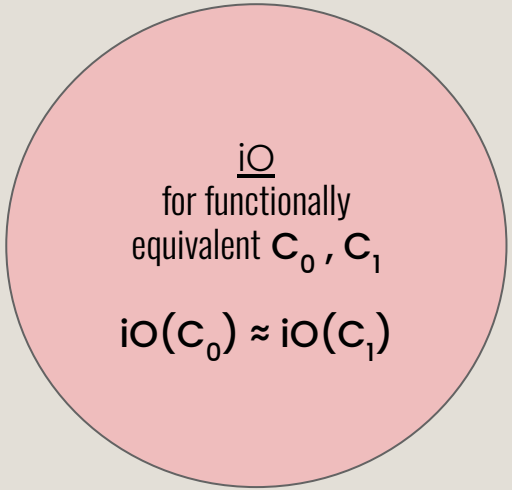


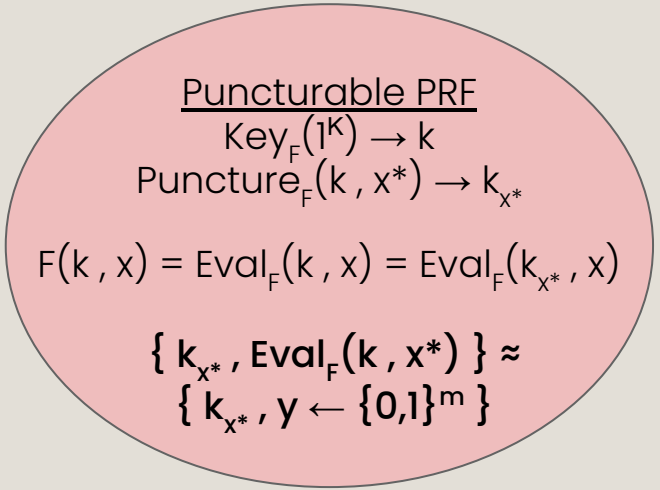
CONSTRUCTION



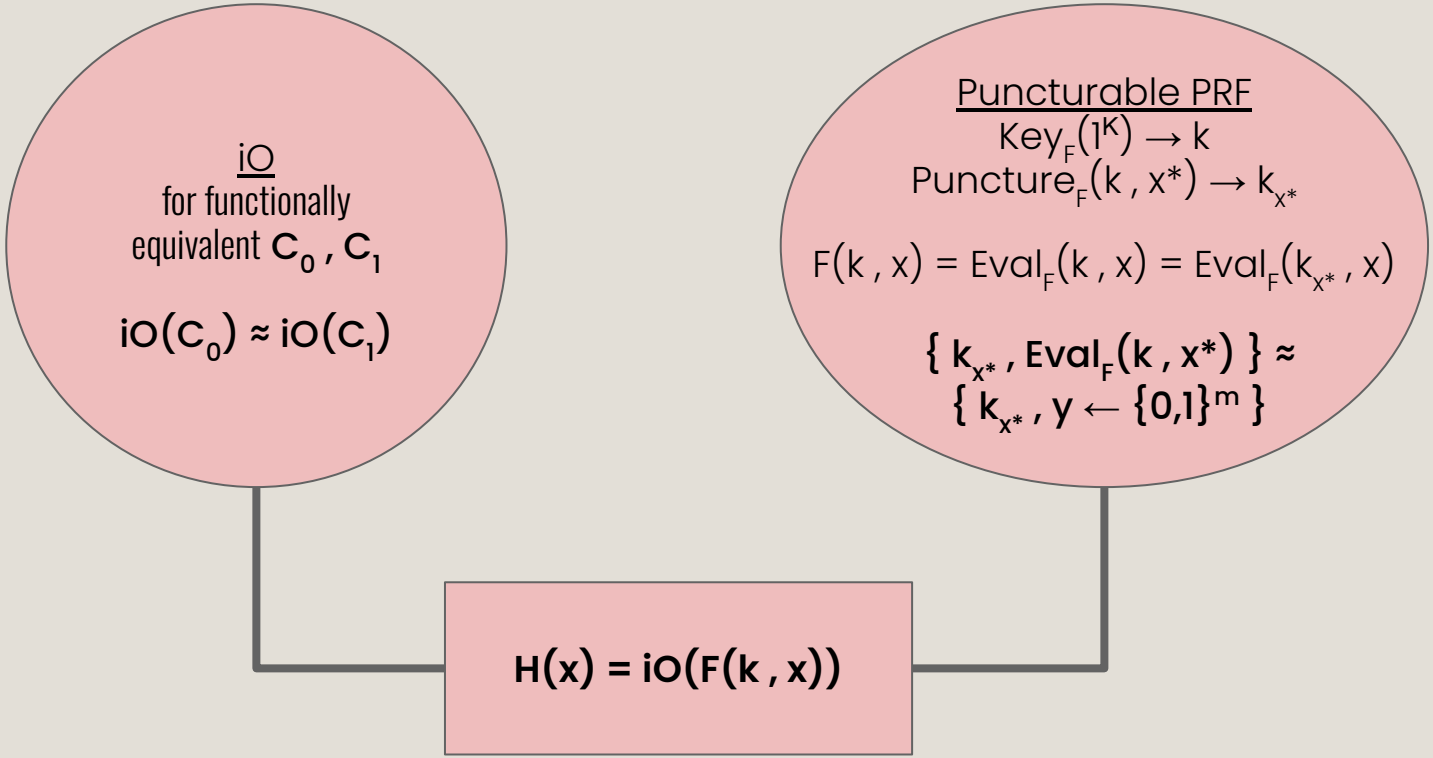


CONSTRUCTION





CONSTRUCTION



PROOF

PROOF

$H(\cdot) = iO(F(k, \cdot))$

$\text{QueryStrategy}^{\mathbf{CCR}}(1^K, r)$

$H(\cdot) = iO(F(k, \cdot))$

$\text{QueryStrategy}^{\mathbf{RO}}(1^K, r)$

PROOF

Hybrid 0

$H(\cdot) = iO(F(k, \cdot))$

$\text{QueryStrategy}^{\text{CCR}}(1^K, r)$

Hybrid 7

$H(\cdot) = iO(F(k, \cdot))$

$\text{QueryStrategy}^{\text{RO}}(1^K, r)$

PROOF

Hybrid 0

$H(\cdot) = iO(F(k, \cdot))$

$\text{QueryStrategy}^{\text{CCR}}(1^K, r)$

Hybrid 1

$\mathcal{O}_{\text{CCR}}$ outputs → random strings

$y^*_i = \text{QueryStrategy}(1^K, r, \{y^*_{i'}\}_{i'<i-1})$

$H^*(\cdot) = iO(F(k^*, \cdot))$

(programmed at outputs of $\mathcal{O}_{\text{CCR}}$)

Hybrid 7

$H(\cdot) = iO(F(k, \cdot))$

$\text{QueryStrategy}^{\text{RO}}(1^K, r)$

PROOF

Hybrid 0

$H(\cdot) = iO(F(k, \cdot))$

$\text{QueryStrategy}^{\text{CCR}}(1^K, r)$

Hybrid 1

$\mathcal{O}_{CCR}$ outputs → random strings

$y^*_i = \text{QueryStrategy}(1^K, r, \{y^*_{i'}\}_{i'<i-1})$

$H^*(\cdot) = iO(F(k^*, \cdot))$

(programmed at outputs of $\mathcal{O}_{CCR}$)

Hybrid 7

$H(\cdot) = iO(F(k, \cdot))$

$\text{QueryStrategy}^{\text{RO}}(1^K, r)$

Hybrid 6

$y^*_i = \text{QueryStrategy}(1^K, r, \{y^*_{i'}\}_{i'<i-1})$

$H(\cdot) = iO(F(k, \cdot))$

(H independent of QueryStrategy outputs)

PROOF

PROOF

Hybrid 1

$F^*(x,i) = y_i^* \oplus b_i \Delta$

if $x = x_i \oplus \Delta$

(random strings y_i^*)

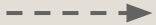
PROOF

Hybrid 1

$F^*(x,i) = y_i^* \oplus b_i \Delta$

if $x = x_i \oplus \Delta$

(random strings y_i^*)



Hybrid 2

$F^*(x,i) = y_i^* \oplus b_i (x \oplus x_i)$

if $x \oplus x_i = \Delta$

(isolate Δ)

PROOF

Hybrid 1

$F^*(x,i) = y_i^* \oplus b_i \Delta$

if $x = x_i \oplus \Delta$

(random strings y_i^*)

Hybrid 2

$F^*(x,i) = y_i^* \oplus b_i (x \oplus x_i)$

if $x \oplus x_i = \Delta$

(isolate Δ)

Hybrid 3

$F^*(x,i) = y_i^* \oplus b_i (x \oplus x_i)$

if $\boldsymbol{\delta}_{\Delta}(x \oplus x_i) = 1$

(using point function $\boldsymbol{\delta}_{\Delta}$)

PROOF

Hybrid 1

$F^*(x,i) = y_i^* \oplus b_i \Delta$

if $x = x_i \oplus \Delta$

(random strings y_i^*)

Hybrid 2

$F^*(x,i) = y_i^* \oplus b_i (x \oplus x_i)$

if $x \oplus x_i = \Delta$

(isolate Δ)

Hybrid 3

$F^*(x,i) = y_i^* \oplus b_i (x \oplus x_i)$

if $\boldsymbol{\delta}_{\Delta}(x \oplus x_i) = 1$

(using point function $\boldsymbol{\delta}_{\Delta}$)

Hybrid 4

$F^*(x,i) = y_i^* \oplus b_i (x \oplus x_i)$

if $\boldsymbol{\delta}_{\boldsymbol{\pi}(\Delta)} \circ \boldsymbol{\pi}(x \oplus x_i) = 1$

(using hard-core predicate within $\boldsymbol{\pi}$)

PROOF

Hybrid 1

$F^*(x,i) = y_i^* \oplus b_i\ \Delta$

if $x = x_i \oplus \Delta$

(random strings y_i^*)

Hybrid 2

$F^*(x,i) = y_i^* \oplus b_i\ (x \oplus x_i)$

if $x \oplus x_i = \Delta$

(isolate Δ)

Hybrid 3

$F^*(x,i) = y_i^* \oplus b_i\ (x \oplus x_i)$

if $\boldsymbol{\delta}_{\Delta}(x \oplus x_i) = 1$

(using point function $\boldsymbol{\delta}_{\Delta}$)

Hybrid 4

$F^*(x,i) = y_i^* \oplus b_i\ (x \oplus x_i)$

if $\boldsymbol{\delta}_{\boldsymbol{\pi}(\Delta)} \circ \boldsymbol{\pi}(x \oplus x_i) = 1$

(using hard-core predicate within $\boldsymbol{\pi}$)

Hybrid 5

$F^*(x,i) = y_i^* \oplus b_i\ (x \oplus x_i)$

if $\boldsymbol{\delta}_{\boldsymbol{\pi}(\Delta)} \circ \boldsymbol{\pi}(x \oplus x_i) = 1$

(using the complement of the hard-core predicate within $\boldsymbol{\pi}$)

PROOF

Hybrid 1

$F^*(x,i) = y_i^* \oplus b_i \Delta$

if $x = x_i \oplus \Delta$

(random strings y_i^*)

Hybrid 2

$F^*(x,i) = y_i^* \oplus b_i (x \oplus x_i)$

if $x \oplus x_i = \Delta$

(isolate Δ)

Hybrid 3

$F^*(x,i) = y_i^* \oplus b_i (x \oplus x_i)$

if $\boldsymbol{\delta}_{\Delta}(x \oplus x_i) = 1$

(using point function $\boldsymbol{\delta}_{\Delta}$)

Hybrid 4

$F^*(x,i) = y_i^* \oplus b_i (x \oplus x_i)$

if $\boldsymbol{\delta}_{\boldsymbol{\pi}(\Delta)} \circ \boldsymbol{\pi}(x \oplus x_i) = 1$

(using hard-core predicate within $\boldsymbol{\pi}$)

Hybrid 5

$F^*(x,i) = y_i^* \oplus b_i (x \oplus x_i)$

if $\boldsymbol{\delta}_{\boldsymbol{\pi}(\Delta)} \circ \boldsymbol{\pi}(x \oplus x_i) = 1$

(using the complement of the hard-core predicate within $\boldsymbol{\pi}$)

Hybrid 6

$y^*_i = \text{QueryStrategy}(1^k, r, \{y^*_{i'}\}_{i'<i-1})$

$H(\cdot) = iO(F(k, \cdot))$

(H not programmed)

IMPLICATIONS and FUTURE WORK

IMPLICATIONS and FUTURE WORK

01

Feasibility Result:

Hash based free-XOR garbling schemes are secure even without assuming RO

IMPLICATIONS and FUTURE WORK

01

Feasibility Result:

Hash based free-XOR garbling schemes are secure even without assuming RO

02

Hierarchy of Cryptographic Primitives:

Weak CCR $\not\Rightarrow$ collision resistance

IMPLICATIONS and FUTURE WORK

01

Feasibility Result:

Hash based free-XOR garbling schemes are secure even without assuming RO

02

Hierarchy of Cryptographic Primitives:

Weak CCR $\not\Rightarrow$ collision resistance

03

Future Work:

de-obfuscation and efficient instantiation?

IMPLICATIONS and FUTURE WORK

01

Feasibility Result:

Hash based free-XOR garbling schemes are secure even without assuming RO

02

Hierarchy of Cryptographic Primitives:

Weak CCR $\not\Rightarrow$ collision resistance

03

Future Work:

de-obfuscation and efficient instantiation?

04

Open Question:

Adaptive Security of hash-based free-XOR garbling schemes in the plain model

Thank You

https://eprint.iacr.org/2025/281