Minimal Assumptions for Cryptographic Tasks and Provable Security in Realistic Models

Dana Dachman-Soled
Microsoft Research New England

Joint Work with Yael Tauman Kalai, Yehuda Lindell, Tal Malkin, Mohammad Mahmoody

Cryptography

- Public Key Encryption

RSA Laboratories RSA

- Digital Signatures

SecurityInnovation®
THE APPLICATION SECURITY COMPANY

NTRU Sign

- Secure Multiparty Computation

RCIM NEWS
online edition

Trading Sugar Beet Quotas - Secure Multiparty Computation in Practice

Provable Secur

probabilistic polynomial time or polynomial sized circuit

- Ideal: Prove that a cryptog
  cannot be broken by any efficient attacker.
- Actually: Our proofs require assumptions.
  - "Factoring is hard": $p \cdot q = n$
- "If an attacker succeeds in breaking PKE the attacker can be used to break factoring."
- This is called a reduction.

Computational Assumptions

- Specific hardness assumptions
  - Factoring is "hard": $n = p \cdot q$; find p, q
  - Discrete log is "hard": $g \in G, g^x$; find x
- Generic hardness a

  Includes factoring, discrete log, others

  - OWF exist: Functions that are easy to compute but hard to invert.
- Constructions based on generic OWF must work when OWF is instantiated with any particular candidate OWF.

Roadmap

- Foundational Questions
  - Limits of Provable Security: Minimal Assumptions
  - OWF vs. Optimally Fair Coin-Tossing
  - New directions
- Towards More Realistic Models
  - Cryptography against Physical Attacks
  - Tamper Resilient Circuits
  - New directions

Roadmap

- Foundational Questions
  - Limits of Provable Security: Minimal Assumptions
  - OWF vs. Optimally Fair Coin-Tossing
  - New directions
- Towards More Realistic Models
  - Cryptography against Physical Attacks
  - Tamper Resilient Circuits
  - New directions

Minimal Assumptions

- What can be constructed assuming only one-way functions (OWF)?
- What requires stronger assumptions?
- Why should we care?

Case: PKE from any OWF?

$(SK, PK) \leftarrow Gen(1^k)$

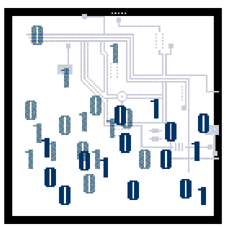
PK

Alice

PK

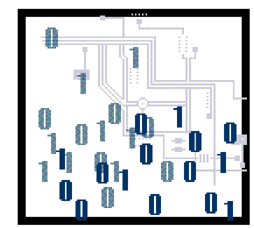
$C = Enc_{PK}(m)$

Bob



OWF: f

$Dec_{sk}(C) = m$



OWF: f

- Despite much effort, no known reduction from PKE to OWF.
- Can we prove that it is impossible?

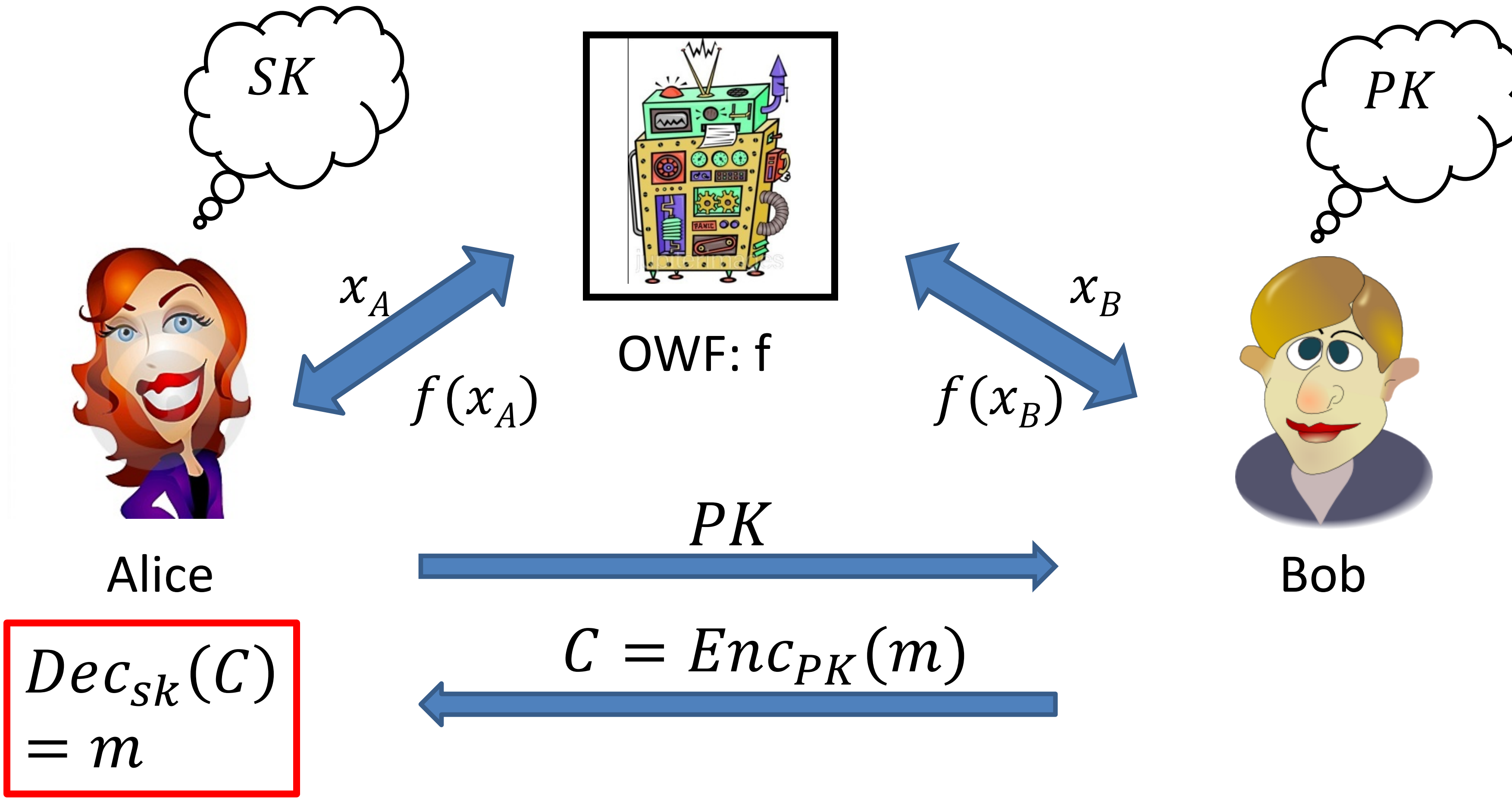
Proving Impossibility Results

- Prove: "There is no construction of PKE from OWF"
- How to formalize?
  - First attempt: Prove $OWF \not\rightarrow PKE$

OWF exists and PKE does not exist

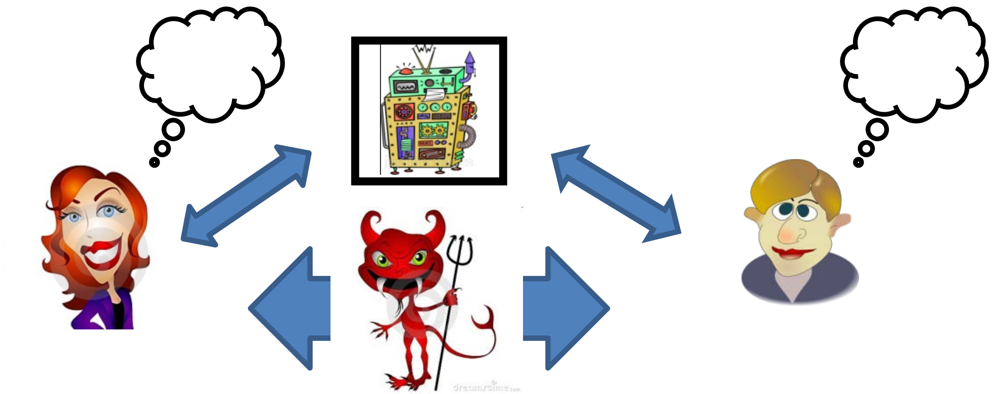
- Problem:
  - Hard to prove OWF exists (implies $P \neq NP$)
  - We believe that PKE exists!
- Instead, we prove "hardness of proving".
- Show that "standard approaches" of proving $OWF \rightarrow PKE$ will fail!

1. Black-Box Construction: PKE scheme E from OWF f



PKE scheme E gets "black-box" access to the OWF f.

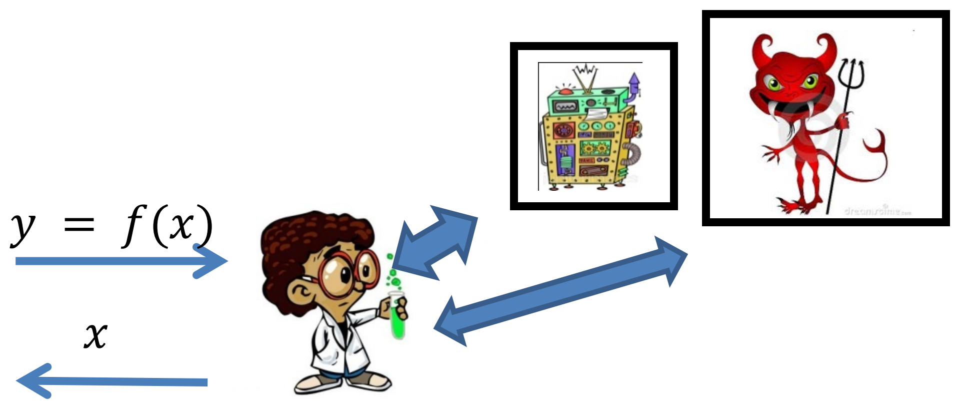
2. Black-Box Analysis: **Reduce Security of E to Security of f**



Present a reduction R such that:

If there is an adversary Eve that breaks security of E

Then R, given oracle access to Eve and f, breaks security of f.



Reduction R

Note: Reduction must work even if f, Eve are inefficient!

Roadmap

- Foundational Questions
  - Limits of Provable Security: Minimal Assumptions
  - OWF vs. Optimally Fair Coin-Tossing
  - New directions
- Towards More Realistic Models
  - Cryptography against Physical Attacks
  - Tamper Resilient Circuits
  - New directions

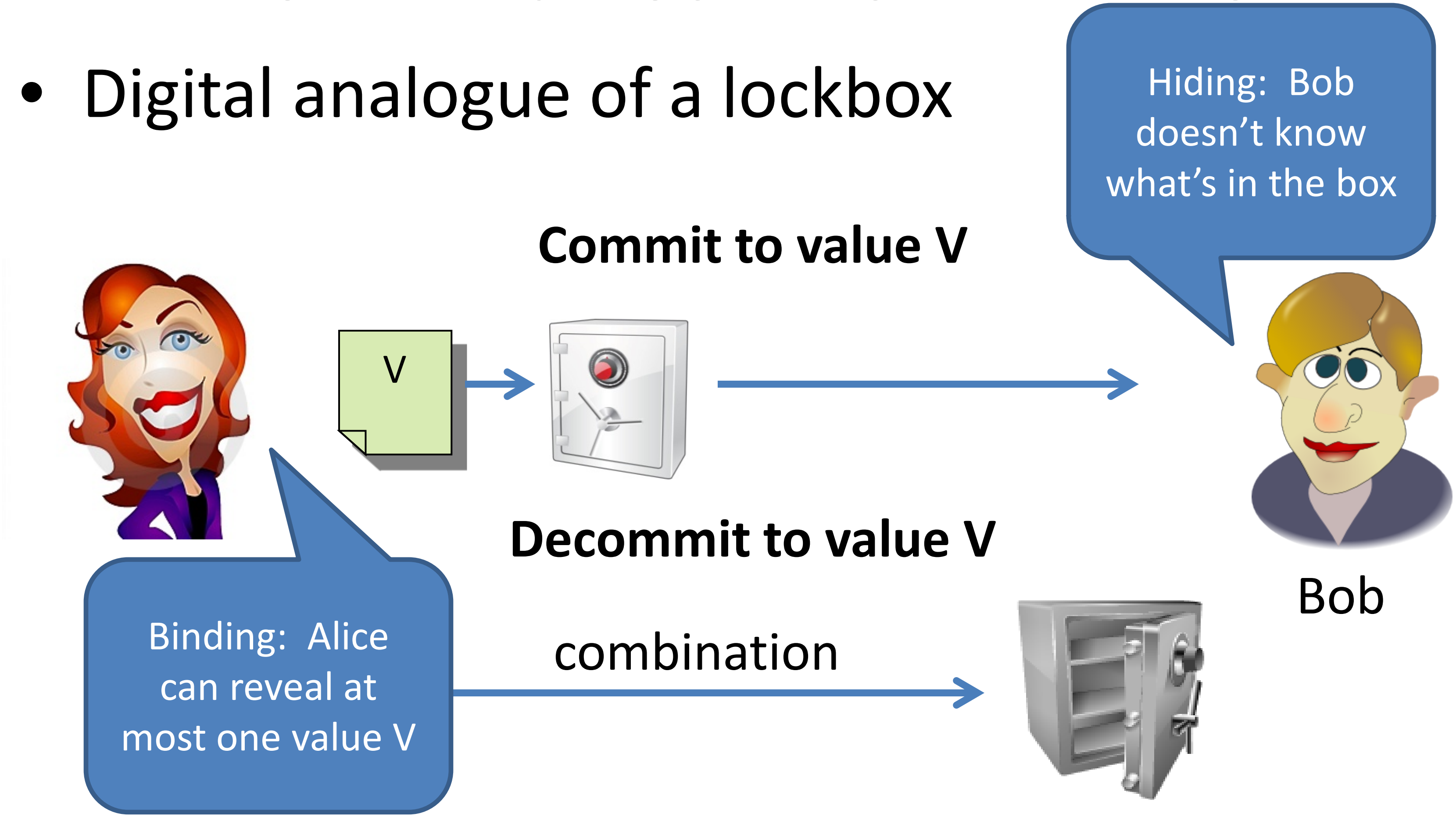
Our Focus: Coin Tossing

- Is there a black-box reduction from Optimally Fair Coin Tossing to OWF.
- Coin Tossing:
  - The output of an honest party is 0 or 1 with probability ½ (= “Fair coin toss”, “bias = 0”)
  - If both parties follow the protocol, they have the same output.
- Basic Primitive
- Used frequently in MPC protocols.

*Joint work with Yehuda Lindell, Tal Malkin, Mohammad Mahmoody

Preliminaries: Commitment

- Digital analogue of a lockbox



- Can be constructed in a black-box manner from OWF.

Blum's Coin-Tossing Protocol ("Over the Telephone")

Alice

Com($coin_A$) →

← $coin_B$

Open to $coin_A$ →

Bob

Alice Output:

$coin_A \oplus coin_B$

Bob Output:

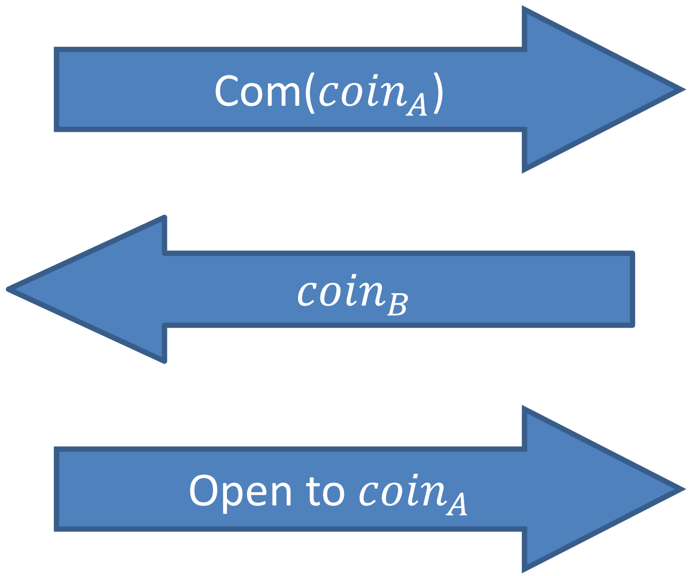
$coin_A \oplus coin_B$

Fairness? If execution completes, Alice cannot bias coin due to binding of commitment. Bob cannot bias coin due to hiding.

But what if Bob must output a value **even in the case that Alice aborts**?

Blum's Coin-Tossing Protocol ("Over the Telephone")

Alice



Bob

Alice Output:

$coin_A \oplus coin_B$

Bob Output:

$coin_A \oplus coin_B$

In this case, Alice can impose bias of ¼.

Note: Black-Box construction from OWF

What is known

- [Cleve86] showed Blum's protocol can be extended to get bias $O(1/\sqrt{r})$ in r rounds from OWF
- [Cleve86] lower bound tells us bias is always at least Ω(1/r) in r rounds
  - Define "optimally-fair coin tossing": coin tossing with bias $O(1/r)$.
- Until recently, not known if it was possible to achieve bias O(1/r)
  - [MNS09] based on work of [GHKL08] constructed protocol that achieves O(1/r) bias.
  - Protocol uses generic MPC, and thus relies on stronger assumptions.

Open questions

- Can we get bias of O(1/r) in r rounds from just OWF?
- Are stronger assumptions necessary for bias of O(1/r)?

In our work, we focus on the question:

Is there a black-box construction of optimally-fair coin-tossing from OWF?

Main Result:

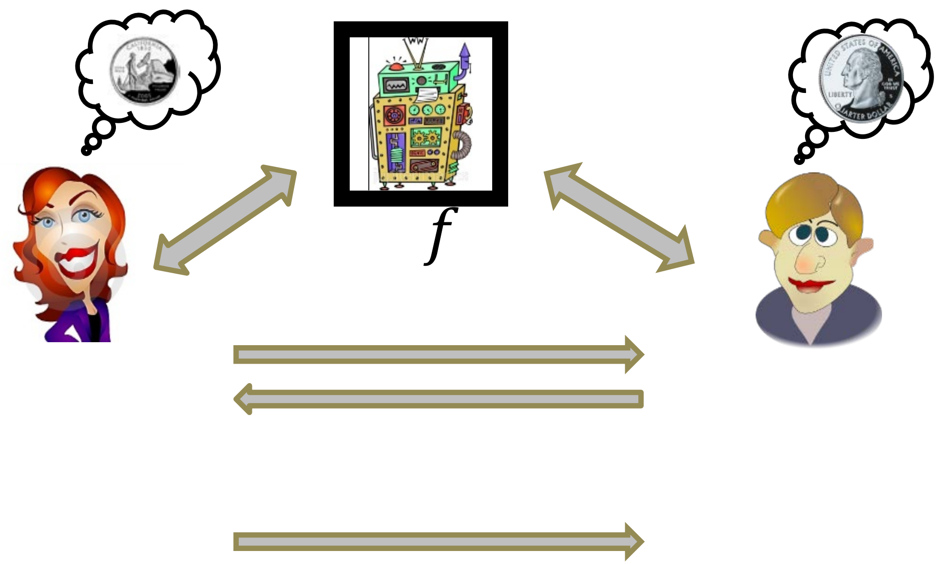
Theorem (informal): Any black-box construction of Optimally Fair Coin-Tossing from OWF will require at least $\Omega(n/\log n)$ rounds.

- Regular coin-tossing requires only 1 round.
- Optimally fair coin-tossing for any number of rounds can be constructing using stronger assumptions.

Proof Intuition

Consider: Random Oracle Model

Goal:
Given any BB construction, show strategies for either Alice or Bob to impose bias $\Omega(\frac{1}{\sqrt{r}})$.



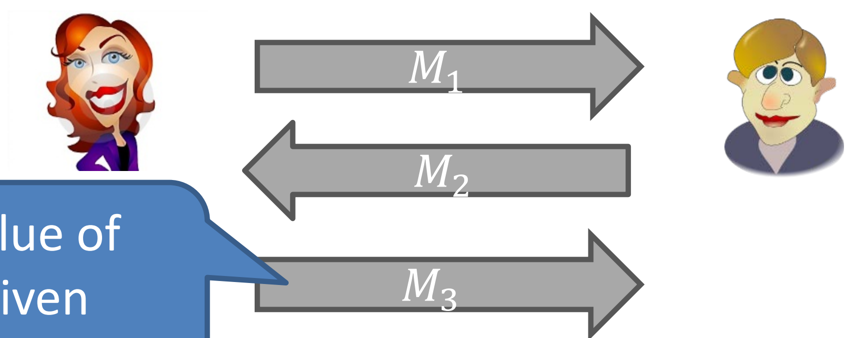
Note: Alice and Bob may be computationally unbounded, but must make "few" queries to oracle f.

Cleve, Impagliazzo 93 Result

[CI93]: For every r-round coin-tossing protocol, there is a strategy for either Alice or Bob to impose bias $\Omega(\frac{1}{\sqrt{r}})$.

Alice and Bob are assumed to be Computationally Unbounded

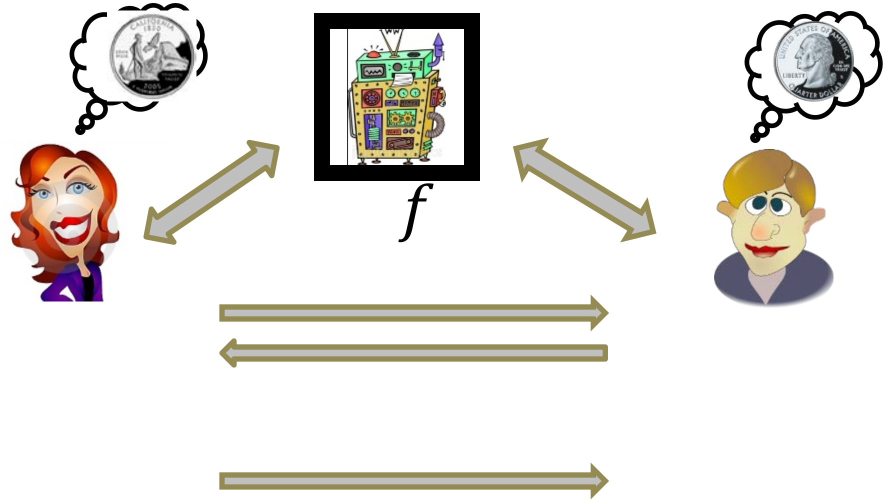
Strategies for A, B involve computing *expected value of coin toss* conditioned on *current transcript* at each pass.



Expected value of outcome given M_1, M_2, M_3

Proof Intuition

Goal:
Given any BB construction, show strategies for either Alice or Bob to impose bias $\Omega(\frac{1}{\sqrt{r}})$.



Idea: Use [CI93] result in the Random Oracle Model

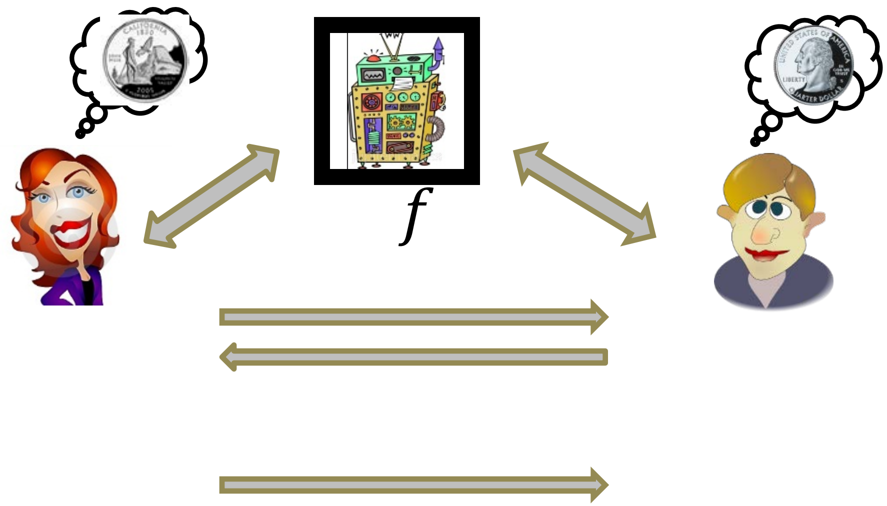
- Issue: Recall [CI93] strategies involve computing *expected values*

  Computing these values must involve making "many" queries since it may involve *inverting* f.

- Solution: Instead of taking expectations over a fixed oracle, we include the randomness of the oracle in the expectation.

Proof Intuition

Goal:
Given any BB construction, show strategies for either Alice or Bob to impose bias $\Omega(\frac{1}{\sqrt{r}})$.



Idea: Use [CI93] result in the Random Oracle Model

- Issue: [CI93] result critically relies on the fact that the views of A and B are *independent* conditioned on the *current transcript*.

  This is not true in the presence of a random oracle.

- Solution: Idea—add queries to transcript to ensure that the views of A and B are (nearly) independent conditioned on the current transcript.

Summary

- We prove that any black-box construction of optimally-fair coin-tossing from OWF will require at least $\Omega(n/\log n)$ rounds.
- This is in contrast to (unfair) coin-tossing which can be constructed from OWF in 1 round.
- Our techniques extend to rule out constructions for a general class of 2-party protocols with $o(n/\log n)$ rounds.

More Impossibility Results

- OWF / KA [IR89], [BM09]
- OWF / CRHF [Simon98]
- PKE / OT [GMRV00]
- PKE / TDF [GMR01]
- OWF / stat. commitment with $o(n/\log n)$ rounds [HHRS07]
- TDP / IBE [BPRVW08]
- TDF / correlated products [Vahlis10]
- Simulatable PKE / Deniable PKE [D12]
- . . .

Roadmap

- Foundational Questions
  - Limits of Provable Security: Minimal Assumptions
  - OWF vs. Optimally Fair Coin-Tossing
  - New directions
- Towards More Realistic Models
  - Cryptography against Physical Attacks
  - Tamper Resilient Circuits
  - New directions

New Directions—Turing Reductions

1. Arbitrary Construction: **PKE scheme E from OWF f**

SK

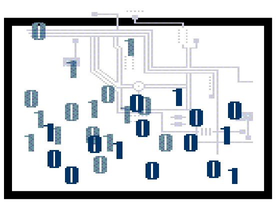
PK

PK

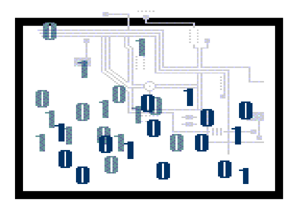
$C = Enc_{PK}(m)$

Alice

Bob



$Dec_{sk}(C) = m$

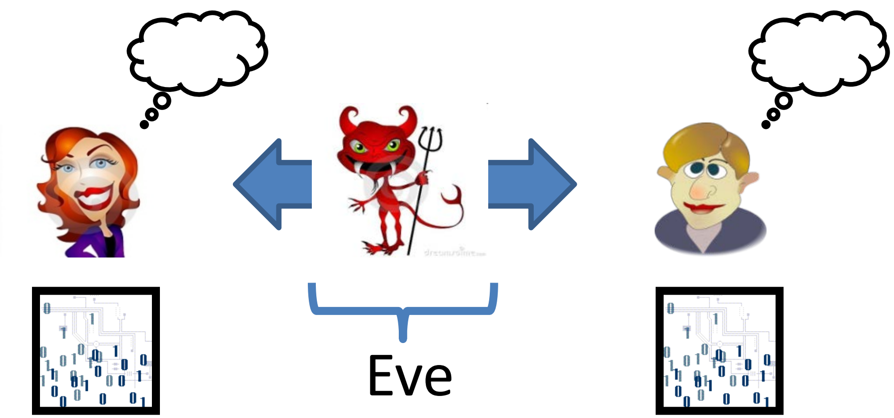


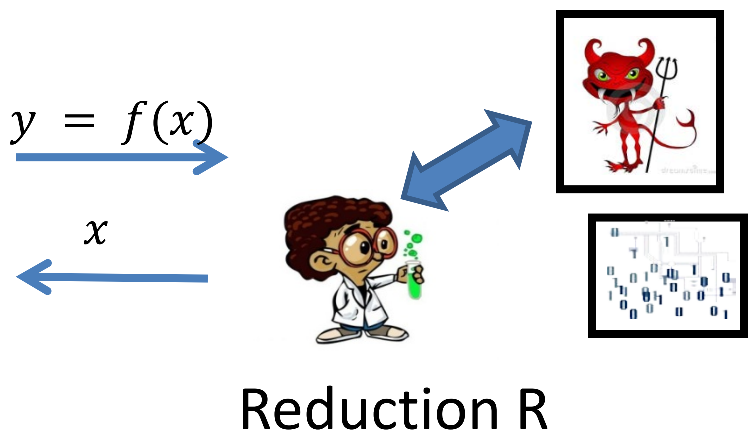
OWF: f

OWF: f

PKE scheme E gets access to the code of the OWF f.

2. Semi BB Analysis: **Reduce Security of E to Security of f**





Present an efficient reduction R such that:

If there is an adversary Eve that breaks security of E

↓

Then R, using BB access to Eve and (nonBB access to) code of f, breaks security of f.

Note: Reduction must work even if Eve is inefficient.

New Directions—Turing Reductions

- [Pass, Tseng, Venkit., 11] showed that under very strong assumptions can rule out Turing reductions between some primitives.
  - Rule out Turing reductions from OWP to OWF
  - Rule out Turing reductions from CRHF to OWF
- Note: In this setting assumptions are necessary
  - Minimally, must assume OWF exists.
- [PTV11] assume existence of OWF with specific strong properties.

Open Questions

- Can we rule out Turing reductions of PKE to OWF?
- Can we rule out other general types of reductions that go beyond BB reductions?
- New proof techniques for both positive and negative results:
  - Positive: New ways to leverage code of OWF or code of adversary?
  - Negative: New results on obfuscation?

Roadmap

- Foundational Questions
  - Limits of Provable Security: Minimal Assumptions
  - OWF vs. Optimally Fair Coin-Tossing
  - New directions
- Towards More Realistic Models
  - Cryptography against Physical Attacks
  - Tamper Resilient Circuits
  - New directions

Roadmap

- Foundational Questions
  - Limits of Provable Security: Minimal Assumptions
  - OWF vs. Optimally Fair Coin-Tossing
  - New directions
- Towards More Realistic Models
  - Cryptography against Physical Attacks
  - Tamper Resilient Circuits
  - New directions

Protecting Circuits against Physical Attacks

Traditional view of cryptography:

Attacker interacts with honest parties in a black-box manner.

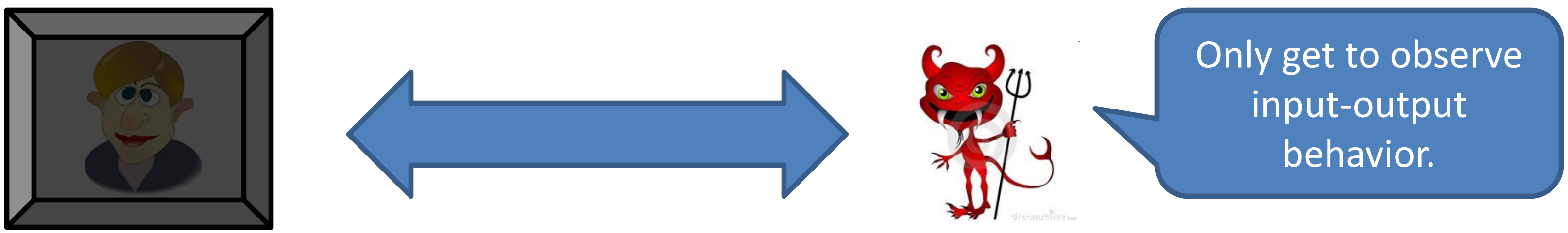
Only get to observe input-output behavior.

* Joint work with Yael Tauman Kalai.

Protecting Circuits against Physical Attacks

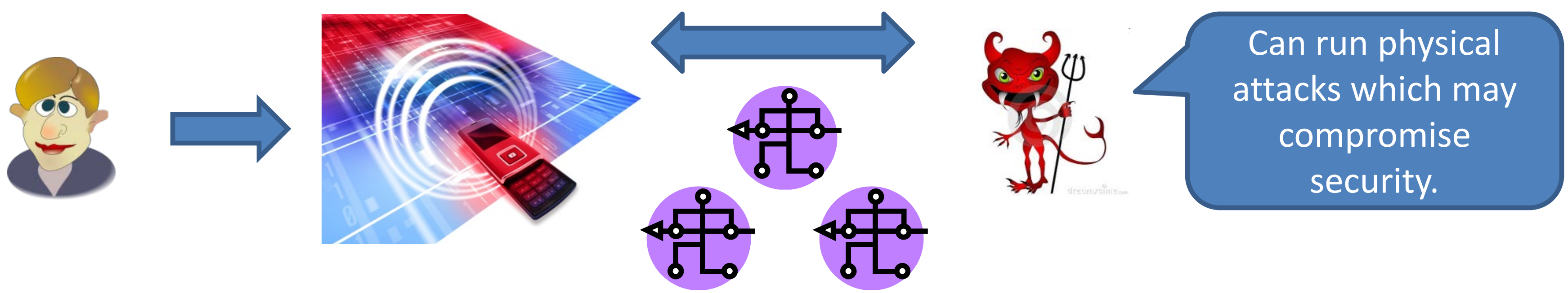
Traditional view of cryptography:

Attacker interacts with honest parties in a black-box manner.



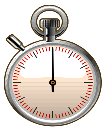
Towards more realistic models:

Attacker may have physical access to honest party.



Examples of Physical Attacks

- Leakage attacks--passively leak some function of the honest party's secret state:
  - Timing attacks [Kocher96,…]
  - Power attacks [Kocher-Jaffe-Jun99,…]
  - Acoustic attacks [Shamir-Tromer04]



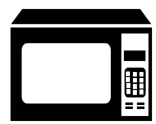
Remote RSA Timing Attacks Practical

Posted by CowboyNeal on Thursday March 13 2003, @08:06PM
from the all-in-the-timing dept.

Examples of Physical Attacks

- Tampering attacks—actively disrupt honest party's computation while observing input/output behavior.
  - Fault attacks [Boneh-DeMillo-Lipton97, Biham-Shamir98, ..]

  - Radiation attacks



1024-bit RSA encryption cracked by carefully starving CPU of electricity

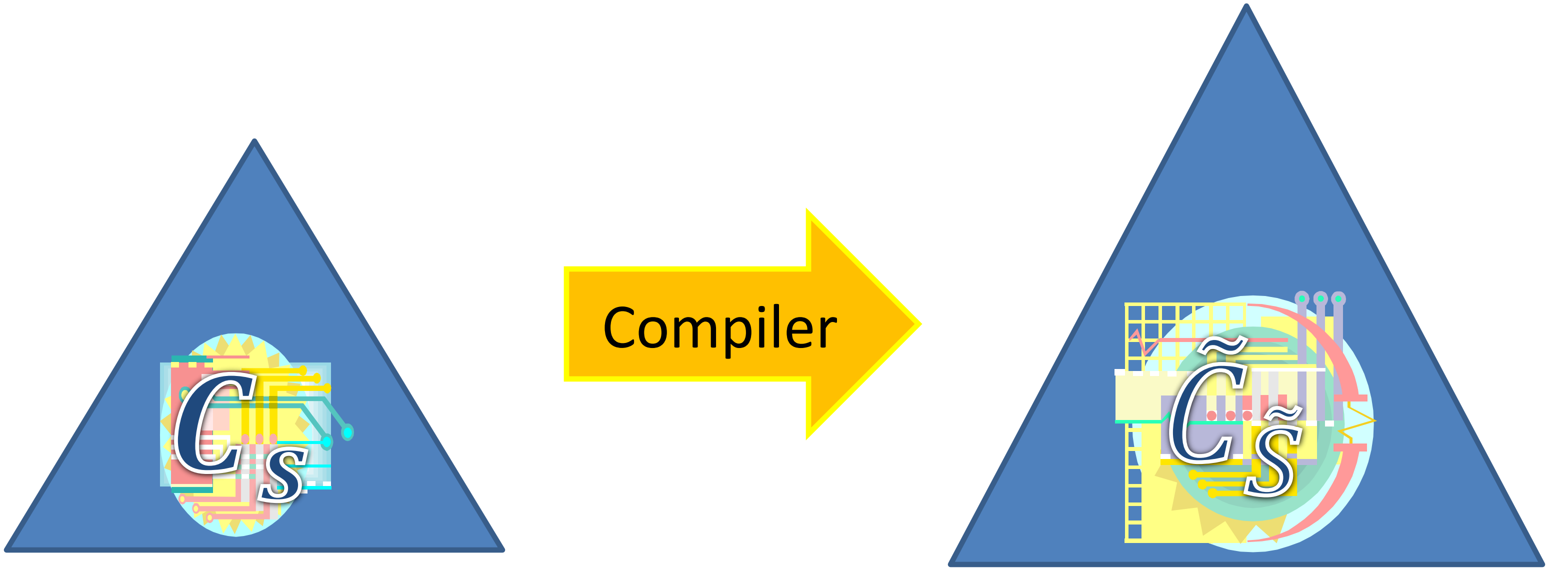
By Sean Hollister posted Mar 9th, 2010 at 2:47 AM

- Our main result focuses on protecting circuits against tampering.

Roadmap

- Foundational Questions
  - Limits of Provable Security: Minimal Assumptions
  - OWF vs. Optimally Fair Coin-Tossing
  - New directions
- Towards More Realistic Models
  - Cryptography against Physical Attacks
  - Tamper Resilient Circuits
  - New directions

Our Results

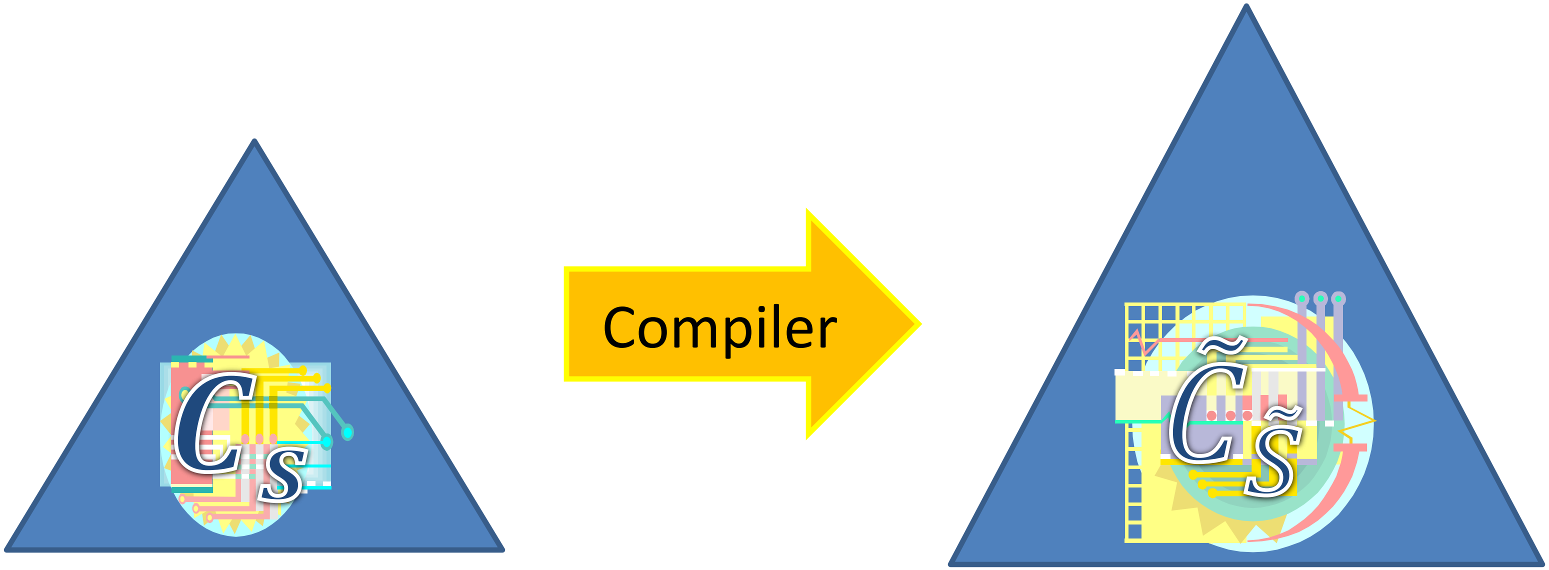


tamper
resilient

Need to define:

1. Tampering model
2. Security guarantee

Our Results



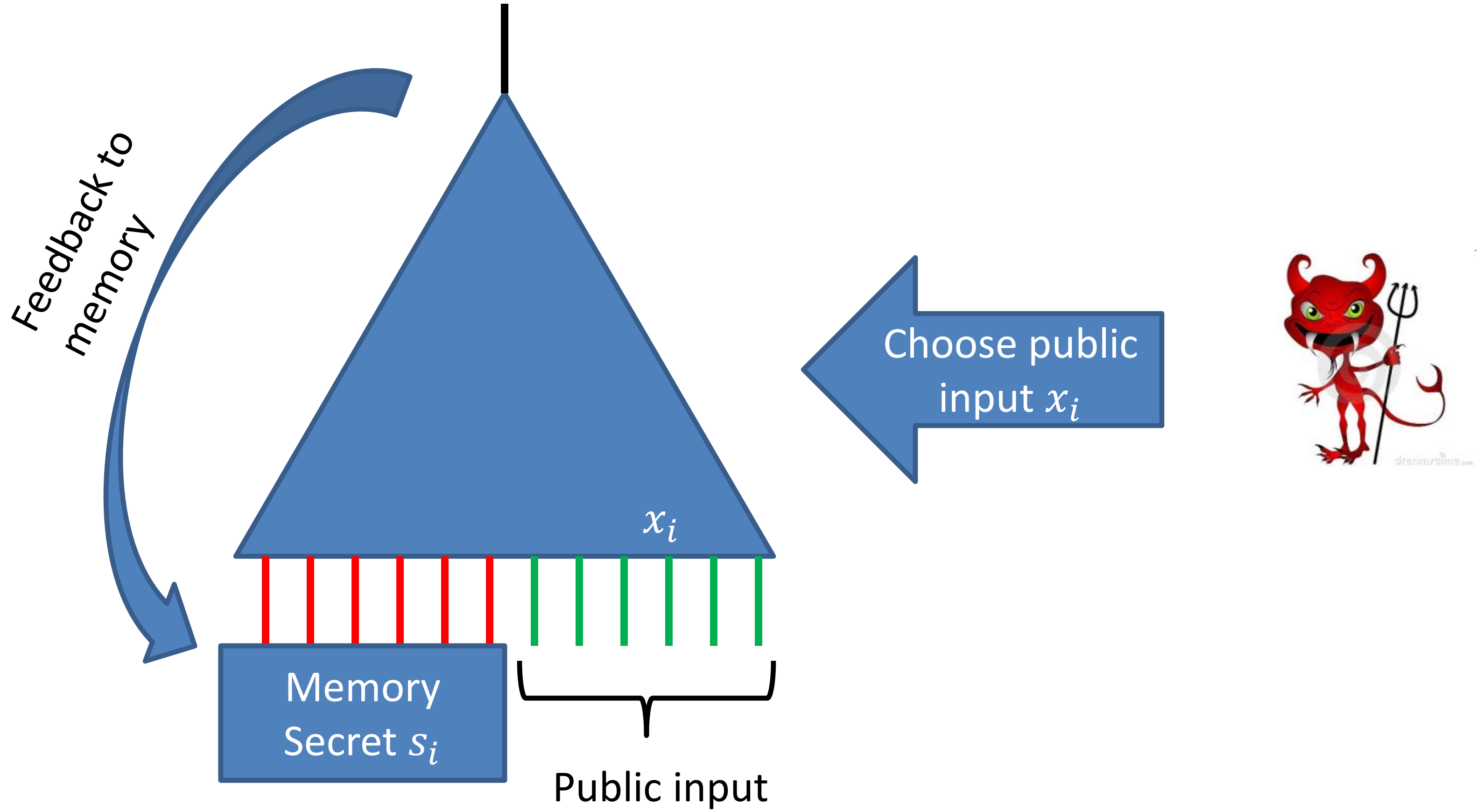
tamper
resilient

Need to define:

1. Tampering model
2. Security guarantee

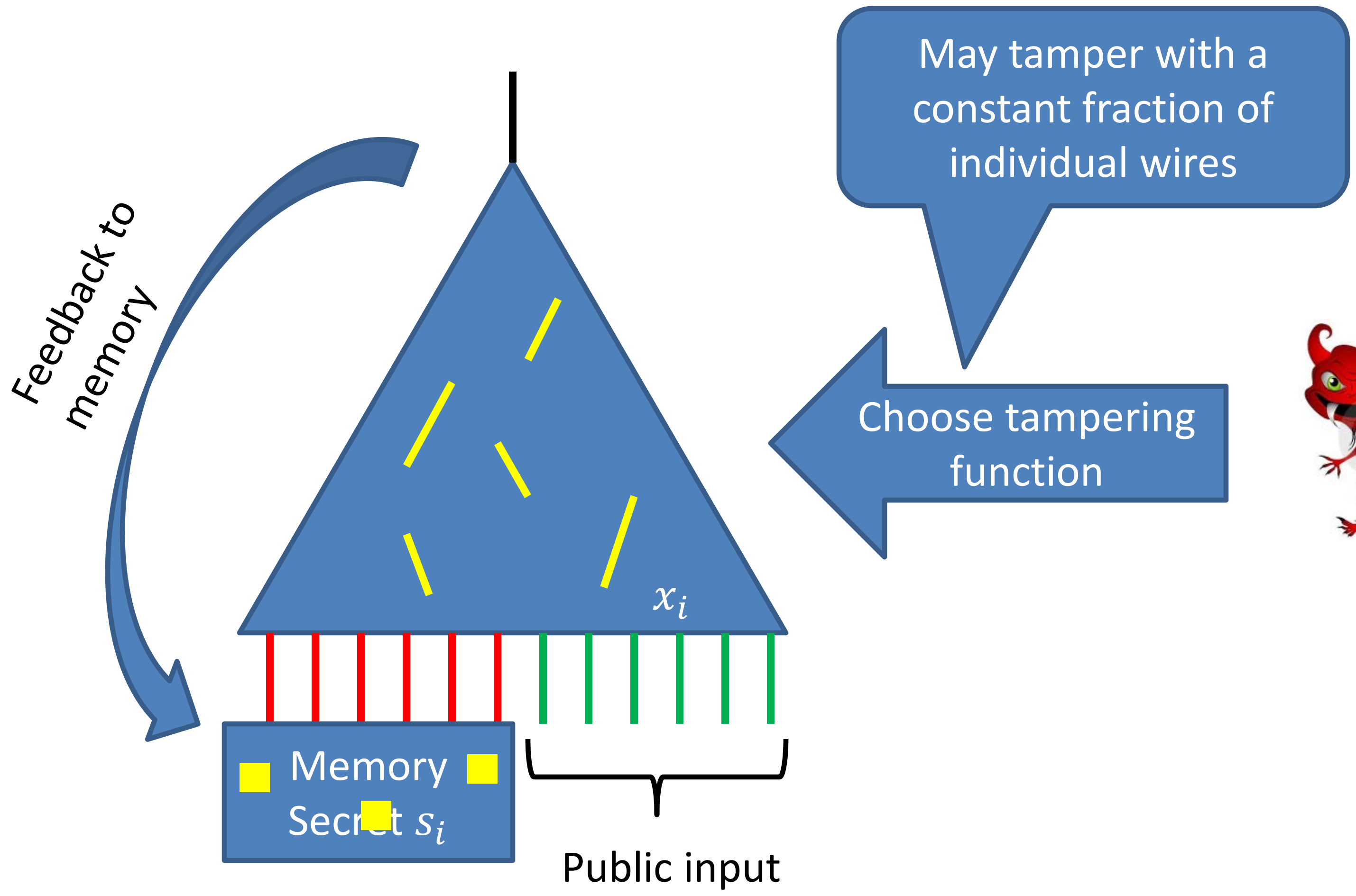
Our Model: Private Circuits

- Introduced by Ishai, Prabhakaran, Sahai, Wagner 2006
- Attack Model: i-th run of circuit C_s



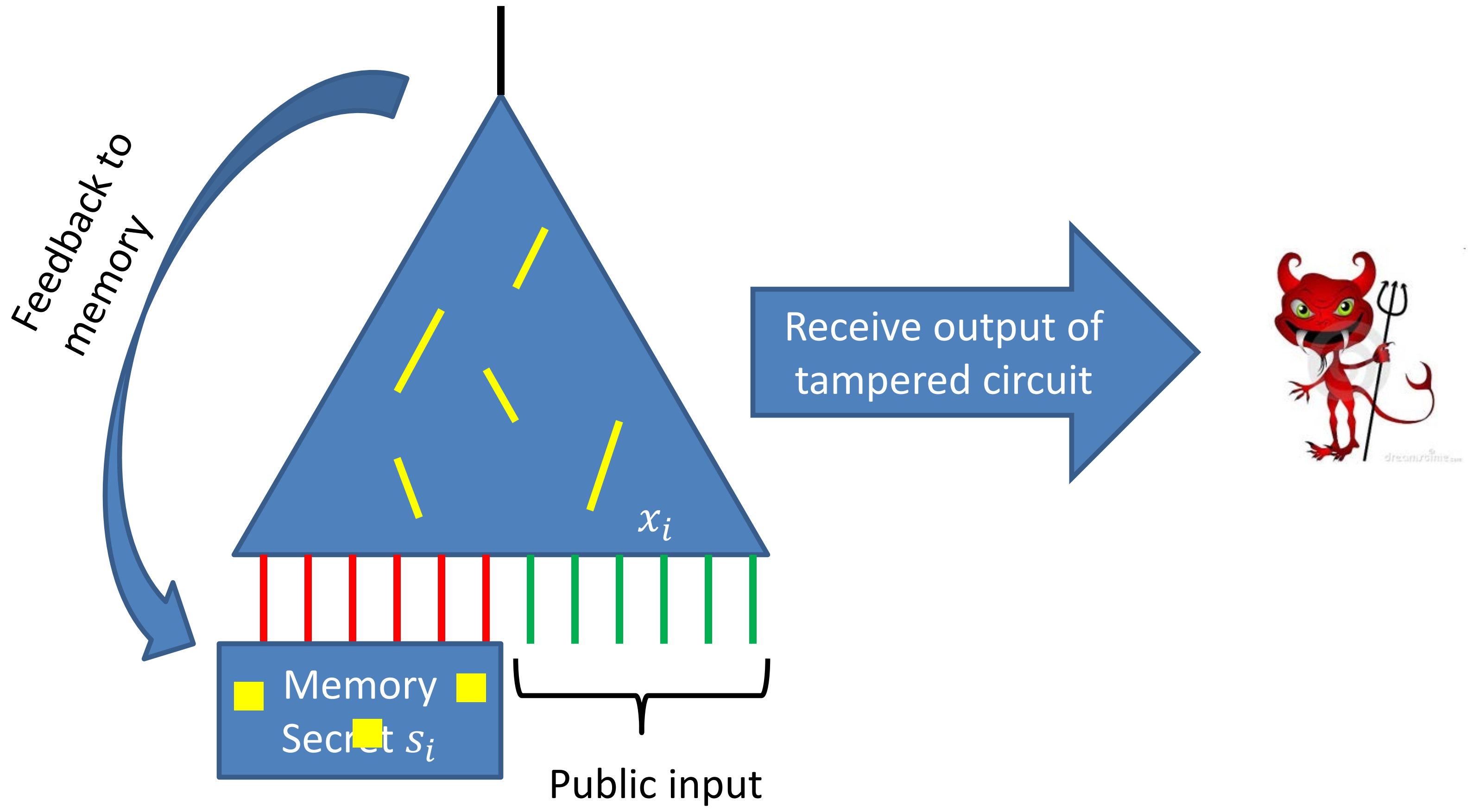
Our Model: Private Circuits

- Introduced by Ishai, Prabhakaran, Sahai, Wagner 2006
- Attack Model: i-th run of circuit C_s

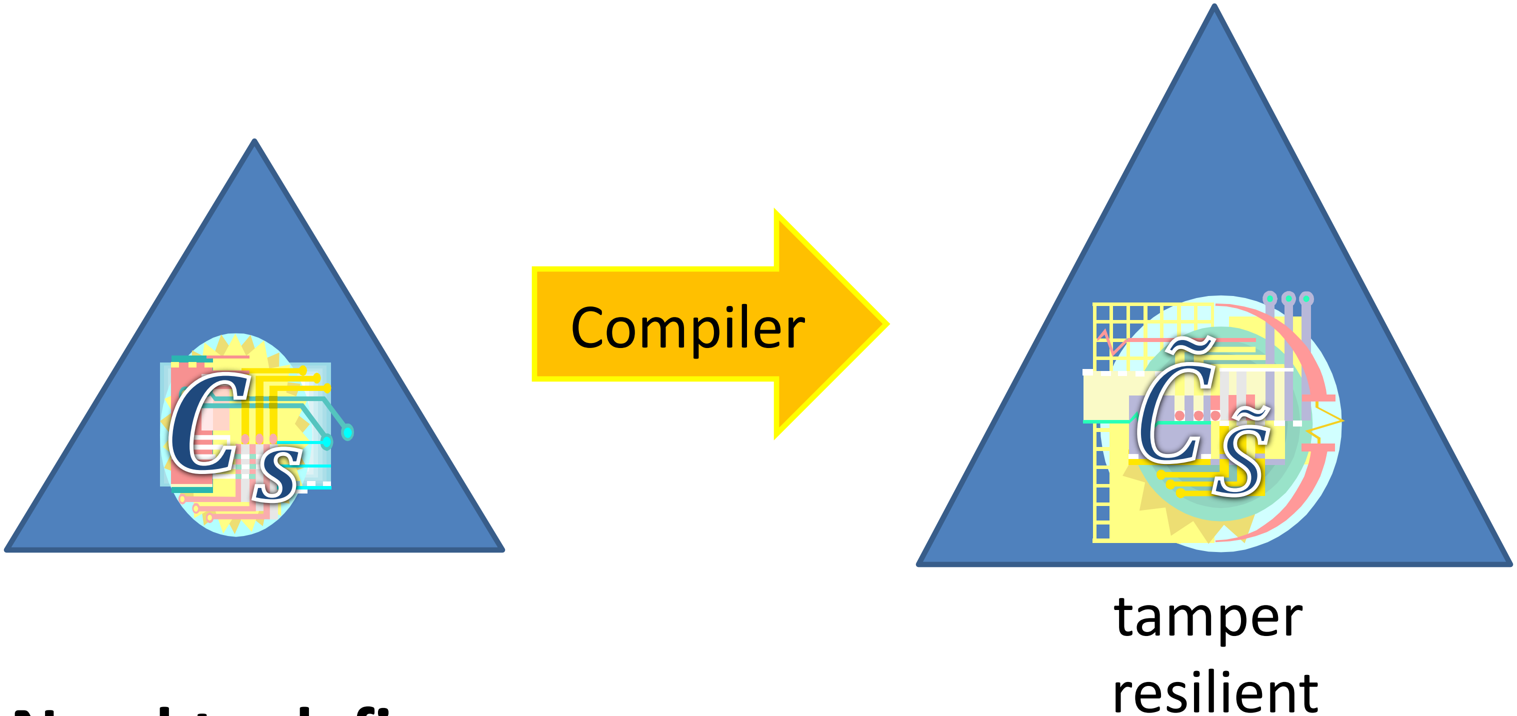


Our Model: Private Circuits

- Introduced by Ishai, Prabhakaran, Sahai, Wagner 2006
- Attack Model: i-th run of circuit C_s



Our Results



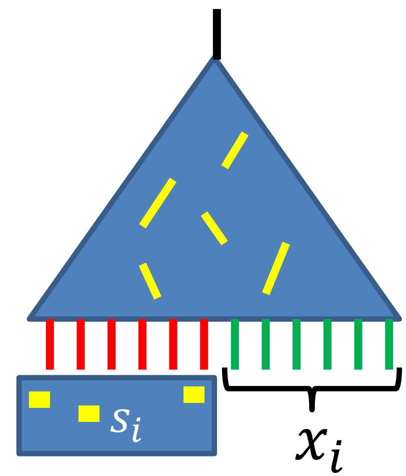
Need to define:

1. Tampering model
2. Security guarantee

Secu e

Tampering with a constant fraction of wires and memory gates.

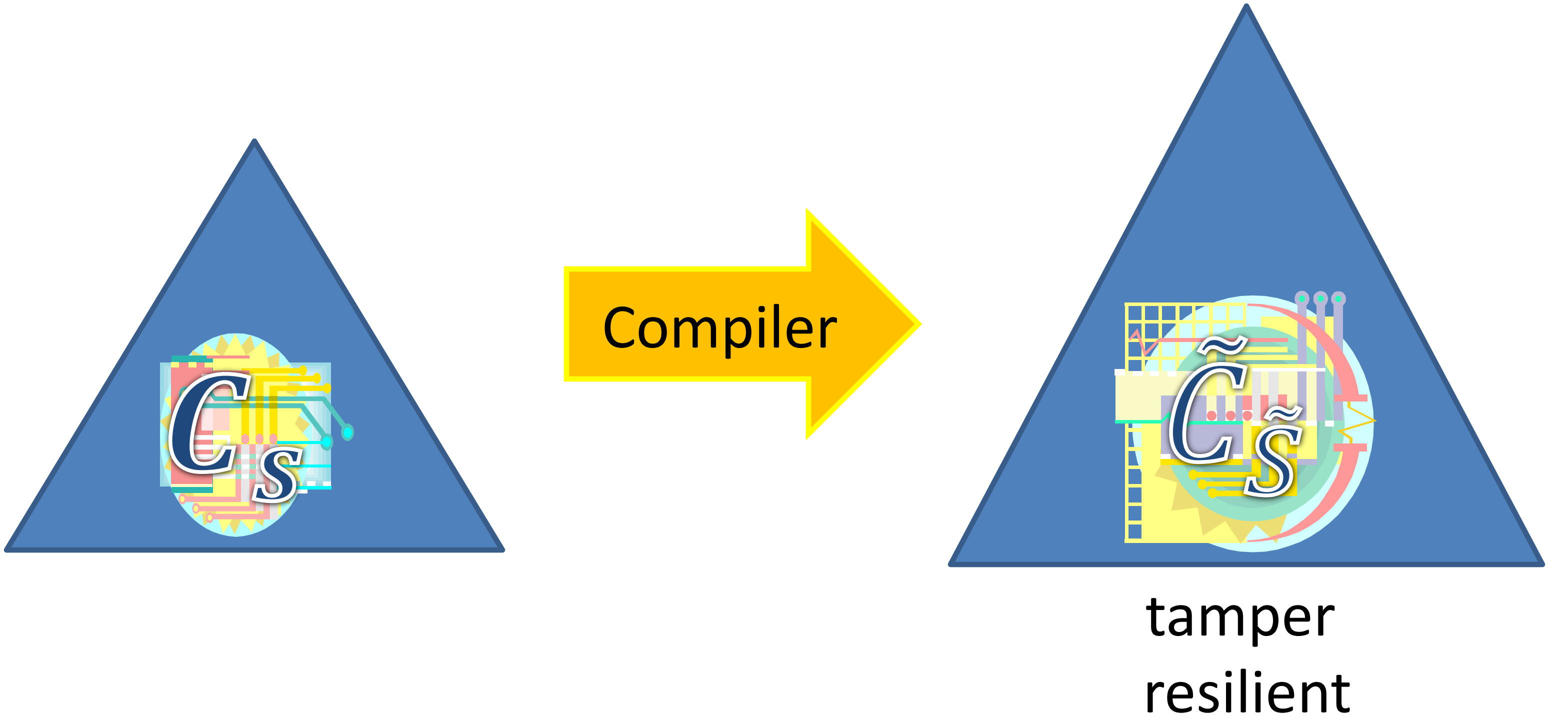
For every  present a simulator Sim s.t.

$$Sim^{C, L(s)} \approx$$  

Only **log** bits of leakage

- log bits of leakage?
- Previous work of [IPSW06]: No leakage, but tampering rate of $1/|C|$.

Our Results



1. Resilient to constant tampering rate
2. Information theoretic

Overview of our Construction

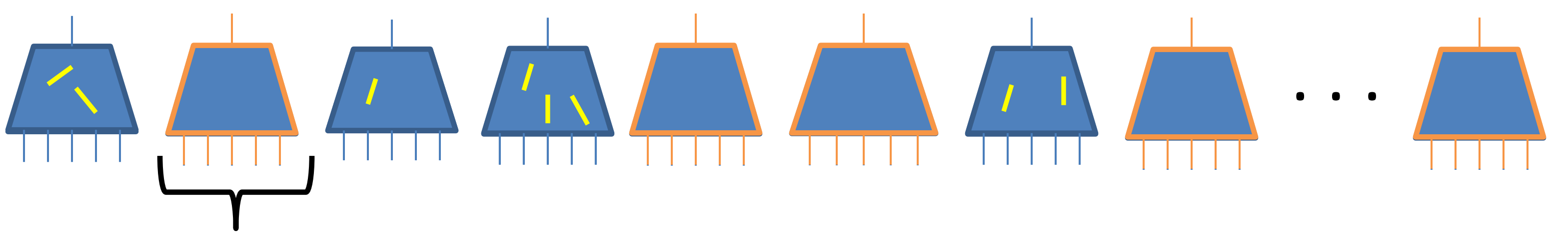
Starting point [IPSW06]:

tamper-resilient

Add tamper-detection component that erases memory if tampering is detected.

We show:

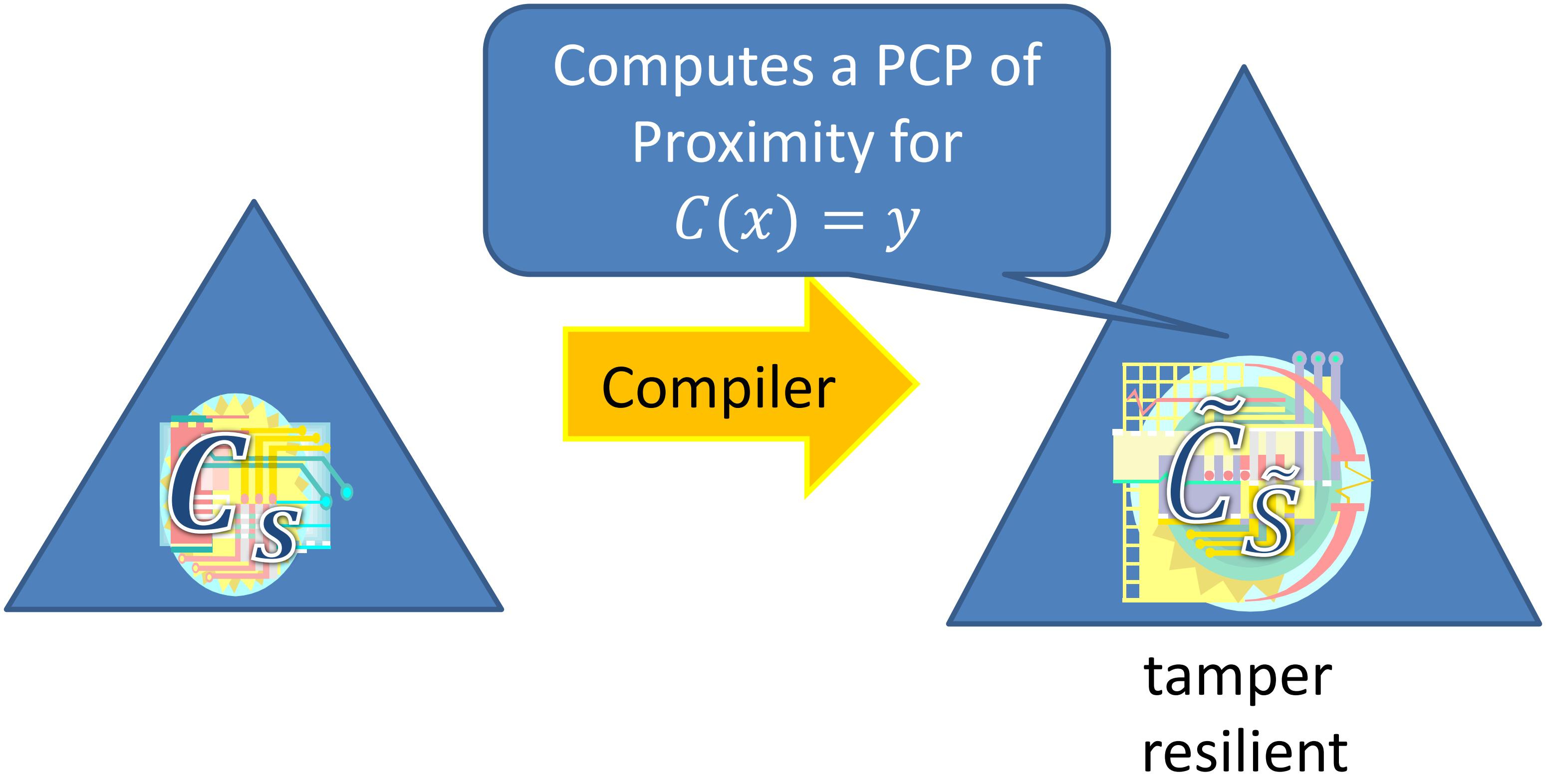
Tamper-detection component in NC^0



circuit of constant size

Tamper-Detection Component

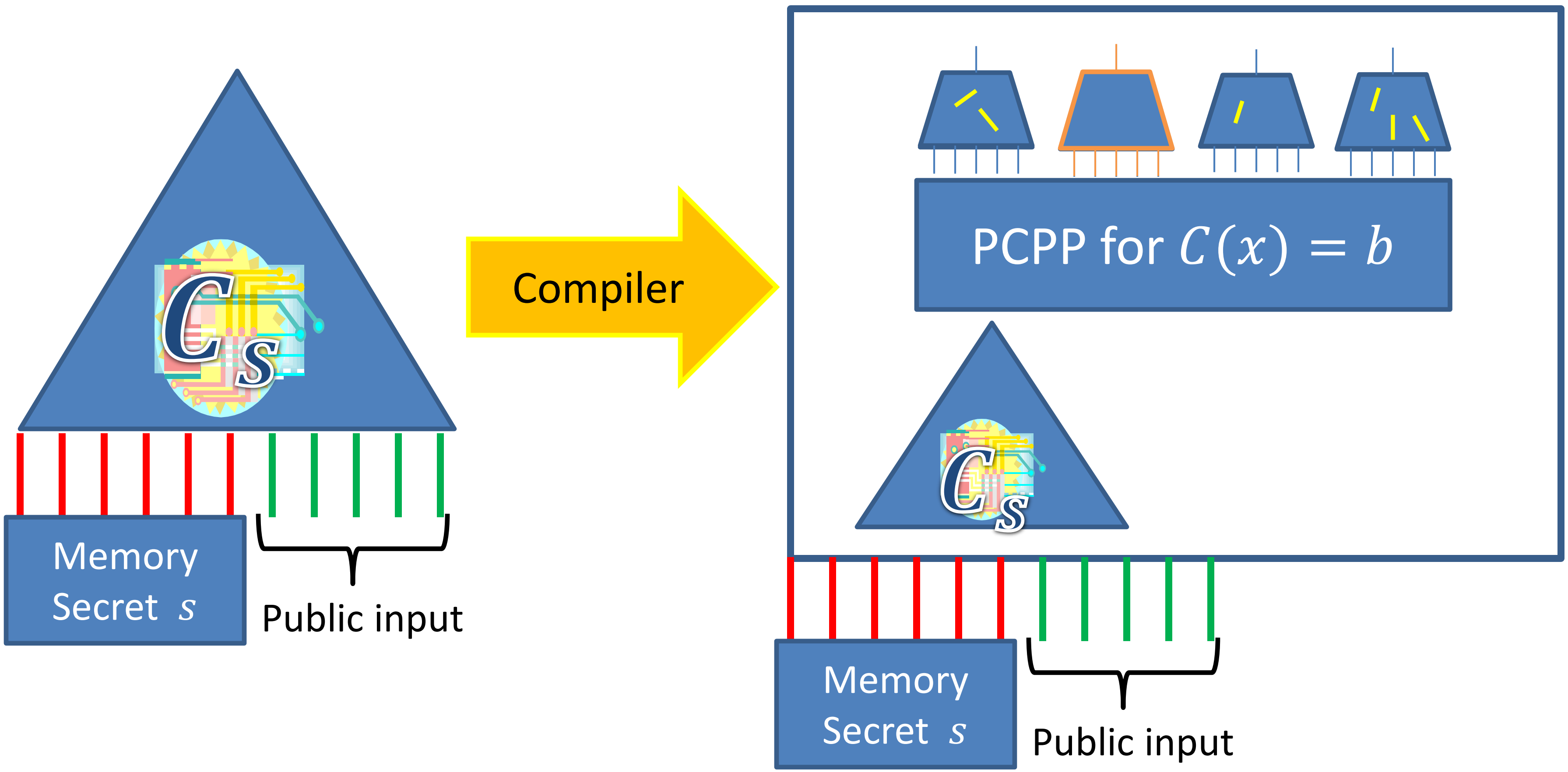
Tool: PCP of Proximity—proof of correctness with special properties [Ben-Sasson, Goldreich, Harsha, Sudan, Vadhan, 06]

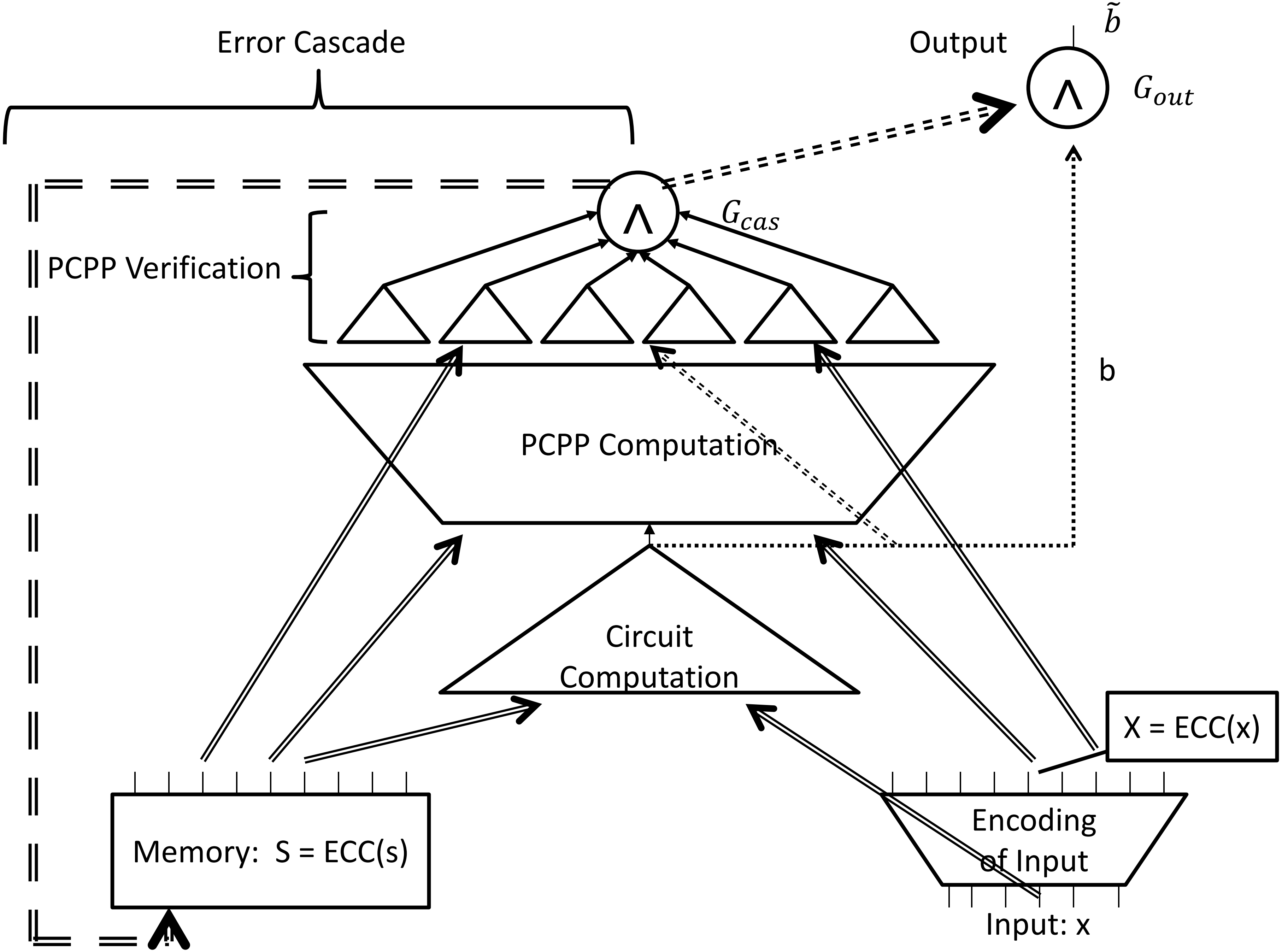


Tamper-Detection Component

Tool: PCP of Proximity

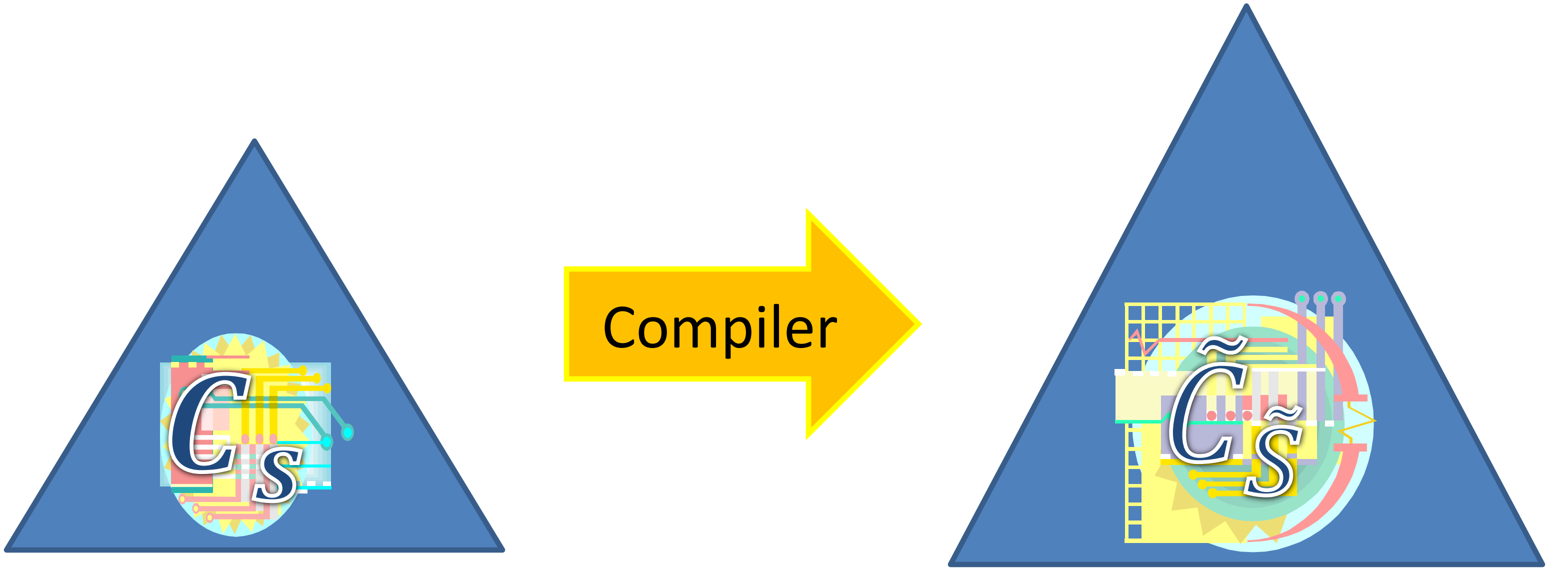
[Ben-Sasson, Goldreich, Harsha, Sudan, Vadhan 06]





Error Cascade
Output
$\tilde{b}$
G_{out}
$\wedge$
G_{cas}
PCPP Verification
PCPP Computation
b
Circuit Computation
X = ECC(x)
Memory: S = ECC(s)
Encoding of Input
Input: x

Summary



- Resilient to constant tampering rate.
- Information theoretic
- Extend to leakage + tampering (in the paper)

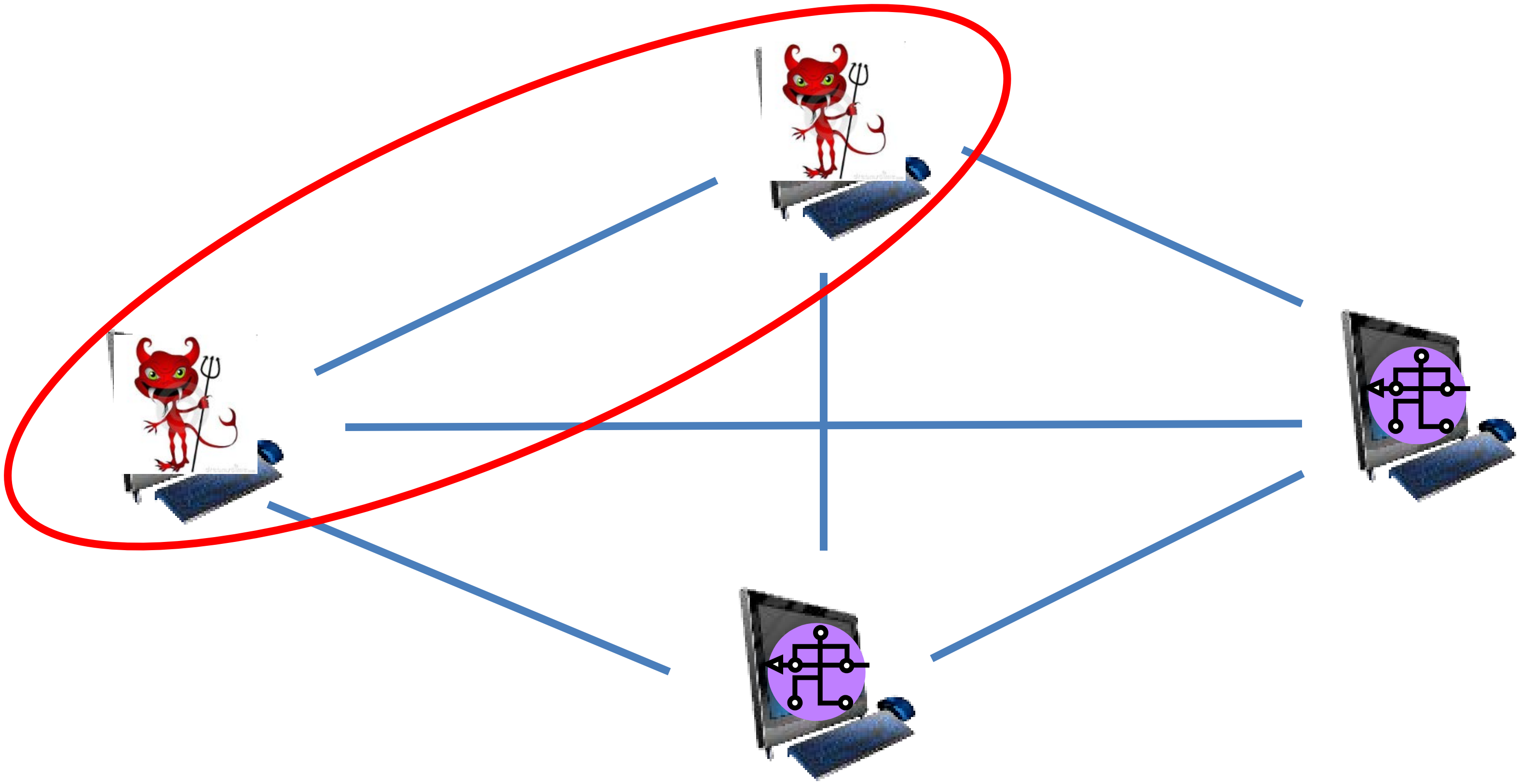
Roadmap

- Foundational Questions
  - Limits of Provable Security: Minimal Assumptions
  - OWF vs. Optimally Fair Coin-Tossing
  - New directions
- Towards More Realistic Models
  - Cryptography against Physical Attacks
  - Tamper Resilient Circuits
  - New directions

New Directions—Better Models

- Better theoretical models for leakage and tampering that capture actual attacks.
- Can we relax security requirements so that blowup in computational resources is reduced?
- Requires better knowledge of EE and actual chip design.

New Directions: Physical Attacks in MPC Setting



Can we give meaningful security guarantees in this setting?

Privacy of inputs, correctness of computation, etc.

[BGJK12, BCH12]

Thank you!