Limits on the Power of Indistinguishability Obfuscation and Functional Encryption

Gilad Asharov
Gil Segev

Hebrew University
This Talk

A framework for proving impossibility results for commonly-used non-black-box techniques

- Limits on the Power of Indistinguishability Obfuscation
- Limits on the Power of Functional Encryption
Obfuscation

- Makes a program “unintelligible” while preserving its functionality

```javascript
for (i=0; i < M.length; i++) {
  // Adjust position of clock hands
  var ML=(ns)?document.layers['nsMinutes'+i]:ieMinutes[i].style;
  ML.top=y[i]+HandY+(i*HandHeight)*Math.sin(min)+scroll;
  ML.left=x[i]+HandX+(i*HandWidth)*Math.cos(min);
}
```

```javascript
for (O79=0; O79<l6x.length; O79++) { var O63=(l70)?document.layers
  ["nsM\151\156u\164\145s"+O79]:ieMinutes[O79].style;
  O63.top=161[O79]+O76+(O79*O75)*Math.sin(O51)+173;
  O63.left=175[O79]+177+(O79*176)*Math.cos(O51);}
```
Obfuscation

- [Barak Goldreich Impagliazzo Rudich Sahai Vadhan Yang01] :
  - **Virtual black-box obfuscation (VBB)**
    Obfuscated program reveals no more than a black box implementing the program **impossible**
  - **Indistinguishability obfuscation (iO)**
    Obfuscations of any two functionally-equivalent programs be computationally indistinguishable may be **possible**

- [Garg Gentry Halevi Raykova Sahai Waters12] :
  A candidate **indistinguishability obfuscator** (iO)
The Power of Indistinguishability Obfuscation
The Power of Indistinguishability Obfuscation

• Public-key encryption, short “hash-and-sign” signatures, CCA-secure public-key encryption, non-interactive zero-knowledge proofs, Injective trapdoor functions, oblivious transfer [SW14]
• Deniable encryption scheme [SW14]
• One-way functions [KMN+14]
• Trapdoor permutations [BPW15]
• Multiparty key exchange [BZ14]
• Efficient traitor tracing [BZ14]
• Full-domain hash without random oracles [HSW14]
• Multi-input functional encryption [GGG+14, AJ15]

• Functional encryption for randomized functionalities [GJK+15]
• Adaptively-secure multiparty computation [GGH+14a, CGP15, DKR15, GP15]
• Communication-efficient secure computation [HW15]
• Adaptively-secure functional encryption [Wat14]
• Polynomially-many hardcore bits for any one-way function [BST14]
• ZAPs and non-interactive witness-indistinguishable proofs [BP15]
• Constant-round zero-knowledge proofs [CLP14]
• Fully-homomorphic encryption [CLT+15]
• Cryptographic hardness for the complexity class PPAD [BPR14]

(Last update: April 2015)
Is there a natural task that cannot be solved using indistinguishability obfuscation?
Black-Box Separations

- The main technique for proving lower bound in cryptography: **Black Box Separations**

- The vast majority of constructions in cryptography are “black box”

  “Building a primitive $X$ from 
  *any implementation* of a primitive $Y$”

- The construction and security proof rely only on the input-output behavior of $Y$ and of $X$'s adversary
- The construction ignores the internal structure of $Y$

- **Examples:**
  - PRF from PRG [GGM86], PRG from OWFs [HILL93,99]
Black-Box Separations

• Typically, show impossibility of “X $\Rightarrow Y$” by:

  “There exists an oracle relative to which Y exists but X does not exist”

• Examples:
  • No key agreement from OWFs [IR89]
  • No CRHF from OWFs [Sim98]
Our Challenge: Non-Black-Box Constructions

- Constructions that are based on iO or FE, almost always have some *non-black-box* ingredient

- Typical example
  From private-key to public-key encryption [SW14] *(simplified)*
  - Private-key scheme: \( \text{Enc}(K, m) = (r, \text{PRF}(K, r) \oplus m) \)
  - Public-key scheme: \( \text{SK} = K, \text{PK} = iO(\text{Enc}(K, \cdot)) \)

**Non-black-box ingredient:**

Need the specific evaluation circuit of the PRF

- How can one reason about such non-black-box techniques?
Overcome this challenge by considering $iO$ for a richer class of circuits: **oracle-aided circuits** (circuits with oracle gates)

Our Solution

Possible gates:
Our Solution

• Transform almost all iO-based constructions from non-black-box to black-box

\[ iO(r, \text{PRF}(K, r) \oplus m) \]

(possible due to [GGM86]+[HILL89])

• Constructing iO for oracle-aided circuits is clearly harder than constructing iO for standard circuits

• Limits on the power of iO for oracle-aided circuits clearly implies limits on the power of iO for standard circuits
iO + TDP ≡ CRHF
Theorem:
There is no black-box construction of a collision-resistant hash function family from
- a trapdoor permutation \( f \) and
- an indistinguishability obfuscator for all oracle-aided circuits \( C^f \)

Unless with an exponential security loss (rules out sub-exponential hardness as well!)

Also rules out: homomorphic encryption, homomorphic commitment, two-message PIR [IKO05]
Techniques We Don’t Capture

- Constructions that use NIZK proofs for languages that are defined relative to a computational primitive

**NIZK proof** \( L = \{(d,r) \mid \exists r \text{ s.t. } d = Enc(i;r)\} \)

- Uses Cook-Levin reduction to SAT
- Makes use of the circuit for deciding L by representing its computation state as boolean formula - *non-black-box*

- [BKSY11] seems as a promising approach for extending our framework to capture such constructions

- Other (less common) techniques (so far not used with iO)
Proof Sketch

• Builds upon and generalizes [Sim98, HHRS07]

• We define an oracle $\Gamma$ such that relative to it:

  1. There exists a **one-way permutation** $f$
     (for this talk - OWP and not TDP...)

  2. There exists an **indistinguishability obfuscator** for all oracle-aided circuits $C_f$

  3. There does not exist a **collision-resistant hash function**
The one-way permutation $f$

$f = \{f_n\}_n$, where each $f_n$ is a uniformly chosen permutation over $\{0,1\}^n$

$O$ and Eval

$O = \{O_n\}_{n \in \mathbb{N}}$, where each $O_n$ is a uniformly chosen permutation over $\{0,1\}^{2n}$

$Eval(\bar{C},a)$ with $|\bar{C}| \leq |a| = n$

Looks for the unique pair $(C,r) \in \{0,1\}^{2^n}$ such that $O_n(C,r) = \bar{C}$

Returns $C^f(a)$

ColFinder

1) On input $C$, ColFinder chooses a uniform $w$, evaluates $C(w)$
2) Samples a uniform $w'$ such that $C(w') = C(w)$
3) Returns $(w,w')$

• **We implement iO as follows:** $\hat{C}(\cdot) = iO(C)$
  • On input oracle-aided circuit $C$ (with $|C|=n$), choose a random $r$
  • Outputs $\tilde{C} = O_n(C,r)$
We Need to Prove

1. $f$ is a **one-way permutation** relative to $\Gamma$
2. $iO$ is an **indistinguishability obfuscator** relative to $\Gamma$
3. There is no CRHF relative to $\Gamma$ (easy)

**Main difficulty:**
Both Eval and ColFinder may carry out an exponential amount of “work”
- Need to show that it does not help the adversary in inverting $f$ or in breaking $iO$
- In [Sim98, HHRS07] there was only ColFinder; here we also have Eval - we have to deal with two “exp-time” oracles and their interaction
- Details: see the paper
Follow-up Work

- **A**, Gil Segev, “On Constructing One-Way Permutations from Indistinguishability Obfuscation”. In TCC-2016-A, ePrint 2015/752

- **Theorem:** There are no fully black-box constructions of a domain-invariant one-way permutation family (the domain is independent of the underlying primitives - f and iO) from
  - a one-way function f and
  - an indistinguishability obfuscator for all oracle-aided circuits C^f

- Matching positive result: There exists a construction of a non-domain-invariant TDP from iO+OWF (Bitansky-Paneth-Wichs, TCC-2016-A)
This Talk

A framework for proving impossibility results for commonly-used non-black-box techniques

• Limits on the Power of Indistinguishability Obfuscation

• Limits on the Power of Functional Encryption
Private-Key FE ⇔ Public-Key Crypto

• **Theorem:**
  There is no black-box construction of a key-agreement protocol with perfect completeness from
  • a one-way permutation f and
  • a **private-key functional encryption** for the class of oracle-aided circuits \( \mathcal{C} = \{ C^f \} \)

• Captures the known constructions [BS15, KSY15, BKS15]
Conclusions

- Limits on the Power of Indistinguishability Obfuscation
  - \( \text{iO} \not\equiv \text{CRHF} \)

- Limits on the Power of Private-Key Functional Encryption
  - Private-Key FE \( \not\equiv \) Key Agreement

Thank You!