

Random Oracle Reducibility

CRYPTO 2011

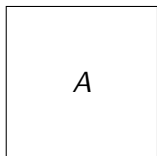
Paul Baecher, Marc Fischlin

Darmstadt University of Technology,
supported by DFG Heisenberg and
Emmy Noether Programmes

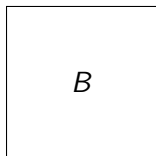


Introduction

Two Cryptographic Schemes. . .



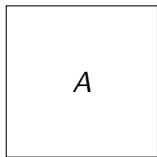
Secure under
assumptions \mathbb{A}



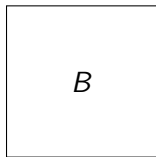
Secure under
assumptions \mathbb{B}

- Possible comparison criteria
 - which scheme is more efficient?
 - how do \mathbb{A} and \mathbb{B} relate?
 - purpose-specific properties (e.g. ciphertext size)?
- rather easy to compare in the standard model

Two Cryptographic Schemes #2

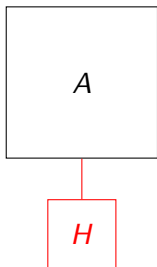


Secure under \mathbb{A}

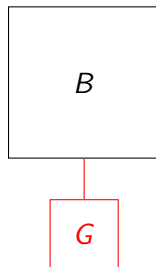


Secure under \mathbb{B}

Two Cryptographic Schemes #2



Secure under \mathbb{A}
in the ROM

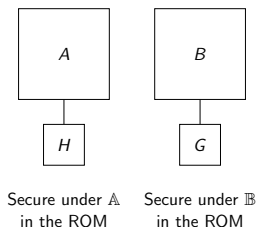


Secure under \mathbb{B}
in the ROM

- Comparison “biased” by random oracle dependency

Comparing The Schemes

- Comparison “biased” by random oracle dependency
- e.g. $\mathbb{A} \subsetneq \mathbb{B}$, but H more demanding than G
 - RO G : provide randomness
 - RO H : POWHF, CR, ...
- perhaps H even uninstantiable!



The Reduction Approach

- Formalizing exact requirements is tedious
- instead, use the cryptographer's approach: reduction
 - A^H secure $\Rightarrow B^{T^H}$ secure
 - any hash function which makes A secure also makes B secure
 - uninstantiability of B implies uninstantiability of A

The Reduction Approach

- Formalizing exact requirements is tedious
- instead, use the cryptographer's approach: reduction
 - A^H secure $\Rightarrow B^{T^H}$ secure
 - any hash function which makes A secure also makes B secure
 - uninstantiability of B implies uninstantiability of A
- may require a non-trivial transformation T (stateless, deterministic, efficient)
 - guarantee “structural compatibility”
- i.e., relative security amongst two schemes

Random Oracle Reducibility

Semi-formal Definition

Scheme A {strictly,strongly,weakly} reduces to scheme B if for every H there exists a transformation T such that

- strictly:

A is G_A^H -secure under $\mathbb{A} \Rightarrow B$ is $G_B^{T^H}$ -secure under \mathbb{B}

where G_S^O defines a security game (think IND-CCA for example) for scheme S

Semi-formal Definition

Scheme A {strictly, strongly, weakly} reduces to scheme B if for every H there exists a transformation T such that

- strictly:

A is G_A^H -secure under $\mathbb{A} \Rightarrow B$ is $G_B^{T^H}$ -secure under \mathbb{B}

- weakly:

A is G_A^H -secure under $\mathbb{A} \Rightarrow B$ is $G_B^{T^H}$ -secure under $\mathbb{A} \cup \mathbb{B}$

where G_S^O defines a security game (think IND-CCA for example) for scheme S

Semi-formal Definition

Scheme A {strictly, strongly, weakly} reduces to scheme B if for every H there exists a transformation T such that

- strictly:

A is G_A^H -secure under $\mathbb{A} \Rightarrow B$ is $G_B^{T^H}$ -secure under \mathbb{B}

- strongly:

A is G_A^H -secure under $\mathbb{A} \Rightarrow \left\{ \begin{array}{l} B \text{ is } G_B^{T^H} \text{-secure under } \mathbb{A} \cup \mathbb{B} \text{ and} \\ B \text{ is } G_B^{T^{H'}} \text{-secure under } \mathbb{B} \text{ for some } H' \\ \text{relying on } \mathbb{H}' \end{array} \right.$

- weakly:

A is G_A^H -secure under $\mathbb{A} \Rightarrow B$ is $G_B^{T^H}$ -secure under $\mathbb{A} \cup \mathbb{B}$

where $G_S^{\mathcal{O}}$ defines a security game (think IND-CCA for example) for scheme S

Example

Example: Hashed ElGamal

- Twin hashed ElGamal (THEG) encryption scheme [CKS09]
- extends hashed ElGamal (HEG) encryption scheme, but milder assumption
 - DH assumption as opposed to strong DH assumption
 - IND-CCA secure given an IND-CCA symmetric scheme
- hence superior at first glance

Example: Hashed ElGamal

- Twin hashed ElGamal (THEG) encryption scheme [CKS09]
- extends hashed ElGamal (HEG) encryption scheme, but milder assumption
 - DH assumption as opposed to strong DH assumption
 - IND-CCA secure given an IND-CCA symmetric scheme
- hence superior at first glance
- our result: THEG* is strongly reducible to HEG

Proof of Reducibility

- THEG* is strongly reducible to HEG
- Proof strategy
 1. show weak reducibility from THEG* to HEG
 2. prove THEG* secure on its own (in the ROM)
- strong reducibility then follows

Scheme Details

HEG (scheme A)

Enc $_A$ (m):

$$y \leftarrow \mathbb{Z}_q$$

$$k \leftarrow H(g^y, X^y)$$

$$c \leftarrow \mathbf{E}_k(m)$$

$$\text{return } (g^y, c)$$

THEG* (scheme B)

Enc $_B$ (m):

$$y \leftarrow \mathbb{Z}_q$$

$$k_0 || k_1 \leftarrow G(g^y, X_0^y, X_1^y)$$

$$c \leftarrow \mathbf{E}_{k_0}(m)$$

$$\text{return } (g^y, c, k_1)$$

Scheme Details

HEG (scheme A)

Enc $_A(m)$:

$$y \leftarrow \mathbb{Z}_q$$

$$k \leftarrow H(g^y, X^y)$$

$$c \leftarrow \mathbf{E}_k(m)$$

$$\text{return } (g^y, c)$$

THEG* (scheme B)

Enc $_B(m)$:

$$y \leftarrow \mathbb{Z}_q$$

$$k_0 || k_1 \leftarrow G(g^y, X_0^y, X_1^y)$$

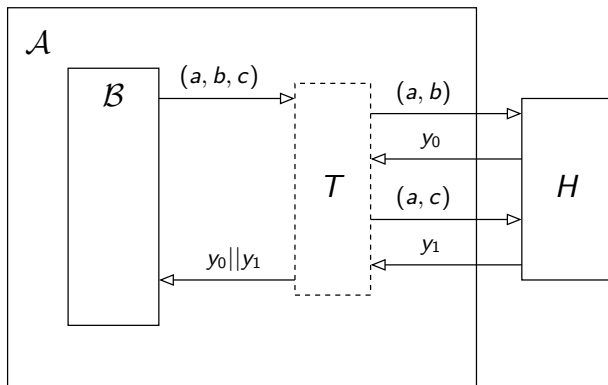
$$c \leftarrow \mathbf{E}_{k_0}(m)$$

$$\text{return } (g^y, c, k_1)$$

- Oracles H and G : need transformation function
- $T^H(a, b, c) = H(a, b) || H(a, c)$

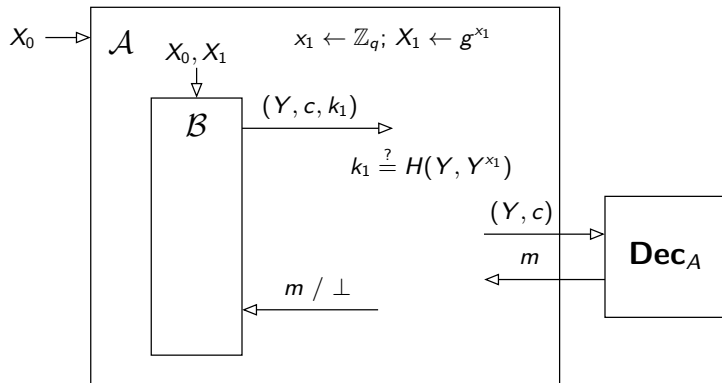
Proof Details

- Handling hash oracle queries
- alleged adversary \mathcal{B} against THEG^*
- algorithm \mathcal{A} performs $T^H(a, b, c) = H(a, b) || H(a, c)$



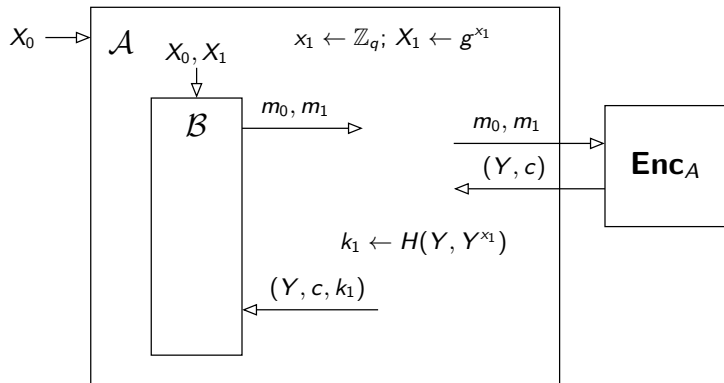
Proof Details

- Handling decryption queries
- algorithm \mathcal{A} simulates second key half



Proof Details

- Handling the encryption challenge query
- algorithm \mathcal{A} simulates second key half



Proof Details

- Algorithm \mathcal{A} outputs whatever \mathcal{B} outputs
- all queries are simulated perfectly
- thus, \mathcal{A} is successful whenever \mathcal{B} is

- THEG* is secure in the ROM (rather technical, see paper)
- hence strongly reducible

Further Results/Applications

Results on Signature Schemes

More examples of (strict) random oracle reductions

- probabilistic RSA FDH signatures reducible to Guillou-Quisquater signatures
- probabilistic RSA FDH signatures reducible to PSS signatures
- Schnorr signatures reducible to BLS signatures

recall: reducibility allows to argue about instantiability

The End

Thank you!

?

References



David Cash, Eike Kiltz, and Victor Shoup.

The twin DiffieHellman problem and applications.

Journal of Cryptology, 22(4):470–504, October 2009.