Better Security for Deterministic Public-Key Encryption: The Auxiliary-Input Setting

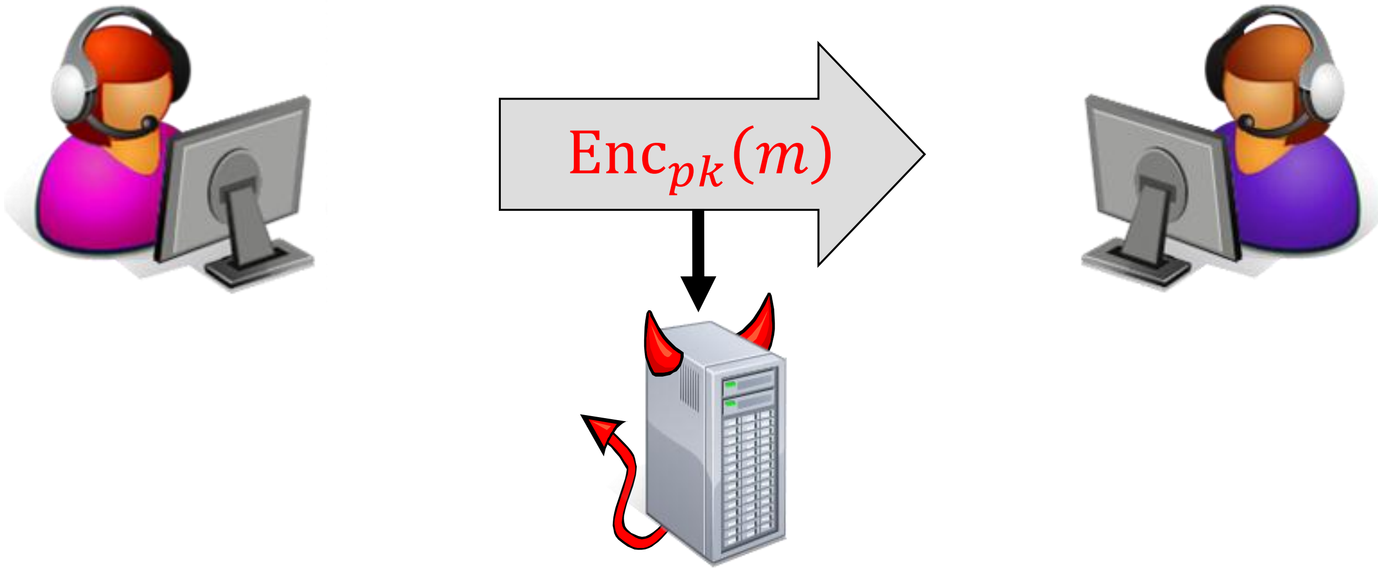
Zvika Brakerski

Weizmann Institute

Gil Segev

Microsoft Research
Silicon Valley

Probabilistic Encryption



Semantic Security [GM82]:

No adversary can learn any meaningful information on m

Encryption algorithm
must be randomized

Deterministic Encryption

Efficiency: short ciphertexts

- Each pk may even define a permutation

Functionality: searchable encryption

- Each pk defines a one-to-one mapping
- Easy to check whether c encrypts m relative to pk

What About Security?

Inherent limitation:

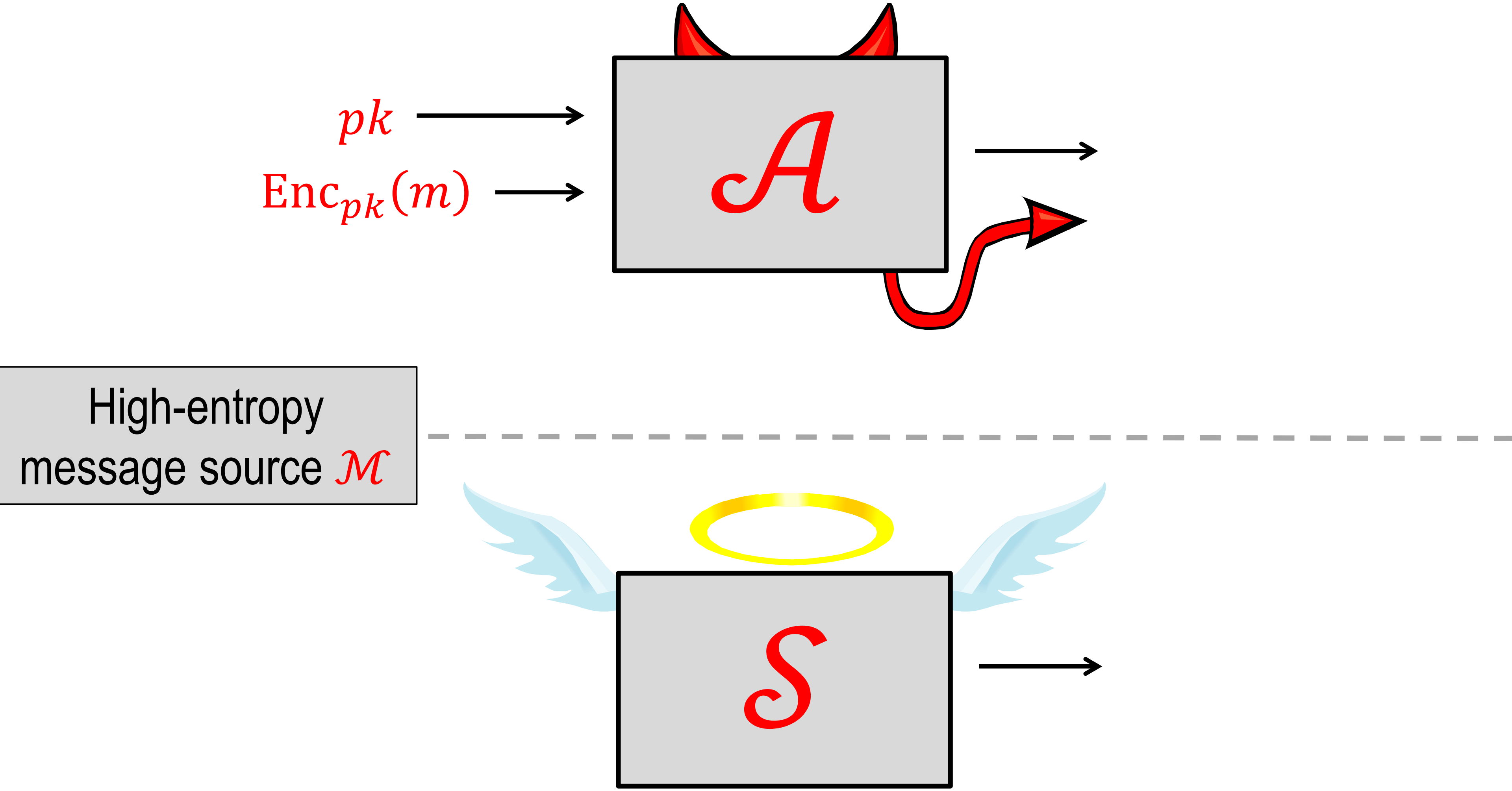
- Each pk defines a one-to-one mapping
- Easy to check whether c encrypts m relative to pk

Security for high-entropy messages [BBO07]

- Inspired by [RW02, DS05] in the symmetric-key setting
- Exciting line of research [BFO08, BFOR08, BBNRSSY09, O'N10,…]
- Meaningful for various applications (e.g., key encapsulation)

$$\left(\mathrm{Enc}_{pk}(key), \mathrm{AES}_{key}(0), \mathrm{AES}_{key}(1), \ldots\right)$$

Notion of Security ([BBO07] simplified)



The Auxiliary-Input Setting

$$\left(\mathrm{Enc}_{pk}(key), \mathrm{AES}_{key}(0), \mathrm{AES}_{key}(1), \ldots\right)$$

Encryption as a building block of a larger system

- Additional information is available
- Does key have any entropy given $(\mathrm{AES}_{key}(0), \mathrm{AES}_{key}(1), \ldots)$?
- No security guarantees from current models and schemes (noticed already by [DS05, BBO07])

This Talk: Better Security

Model

- Deterministic encryption in the auxiliary-input setting
- Hard-to-invert auxiliary inputs
  - Generalizes the high-entropy setting

Constructions

- Security w.r.t all auxiliary inputs that are sub-exponentially hard
- Based on standard hardness assumptions
  - d-Linear for any $d \geq 1$ (Decisional Diffie-Hellman,…)
  - Subgroup indistinguishability [BG10] (Quadratic Residuosity, Composite Residuosity,…)

Outline

- **Hard-to-invert auxiliary inputs**
- **Security in the auxiliary-input setting**
- **Construction based on d-Linear**

Hard-to-Invert Auxiliary Inputs

Definition

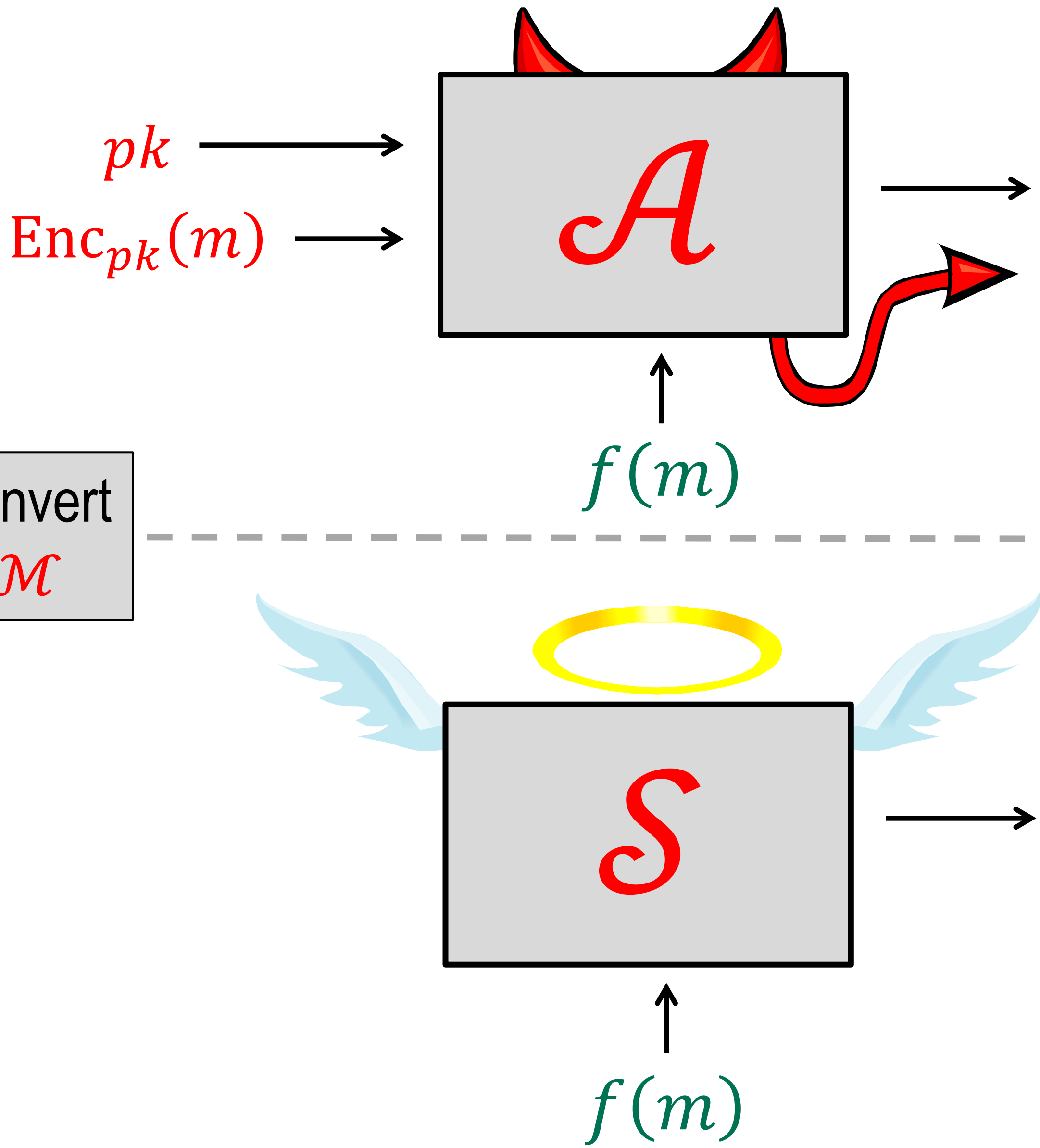
A function f is ϵ-hard-to-invert relative to $\mathcal{X}$ if for any efficient algorithm A it holds that

$$\Pr_{x \leftarrow \mathcal{X}}\left[A\big(f(x)\big) = x\right] \leq \epsilon$$

$$f(key) = \big(\mathrm{AES}_{key}(0), \mathrm{AES}_{key}(1), \dots\big)$$

- A is required to output the exact same x (and not any $x' \in f^{-1}\big(f(x)\big)$ as with one-wayness)
- The source of hardness may be any combination of:
  - Information-theoretic hardness (f has many collisions)
  - Computational hardness (f is injective)

Our Notion of Security (simplified)



f is hard-to-invert relative to $\mathcal{M}$

Construction Based on d-Linear

- Based on the lossy trapdoor function of [FGKRS10]
- $\mathbb{G}$ - group of order p generated by g

$(g^A)_{ij} = (g^{a_{ij}})$

Key generation

- Sample $A \leftarrow \mathbb{Z}_p^{n\times n}$
- Output $sk = A^{-1}$ and $pk = g^A \in \mathbb{G}^{n\times n}$

Encryption

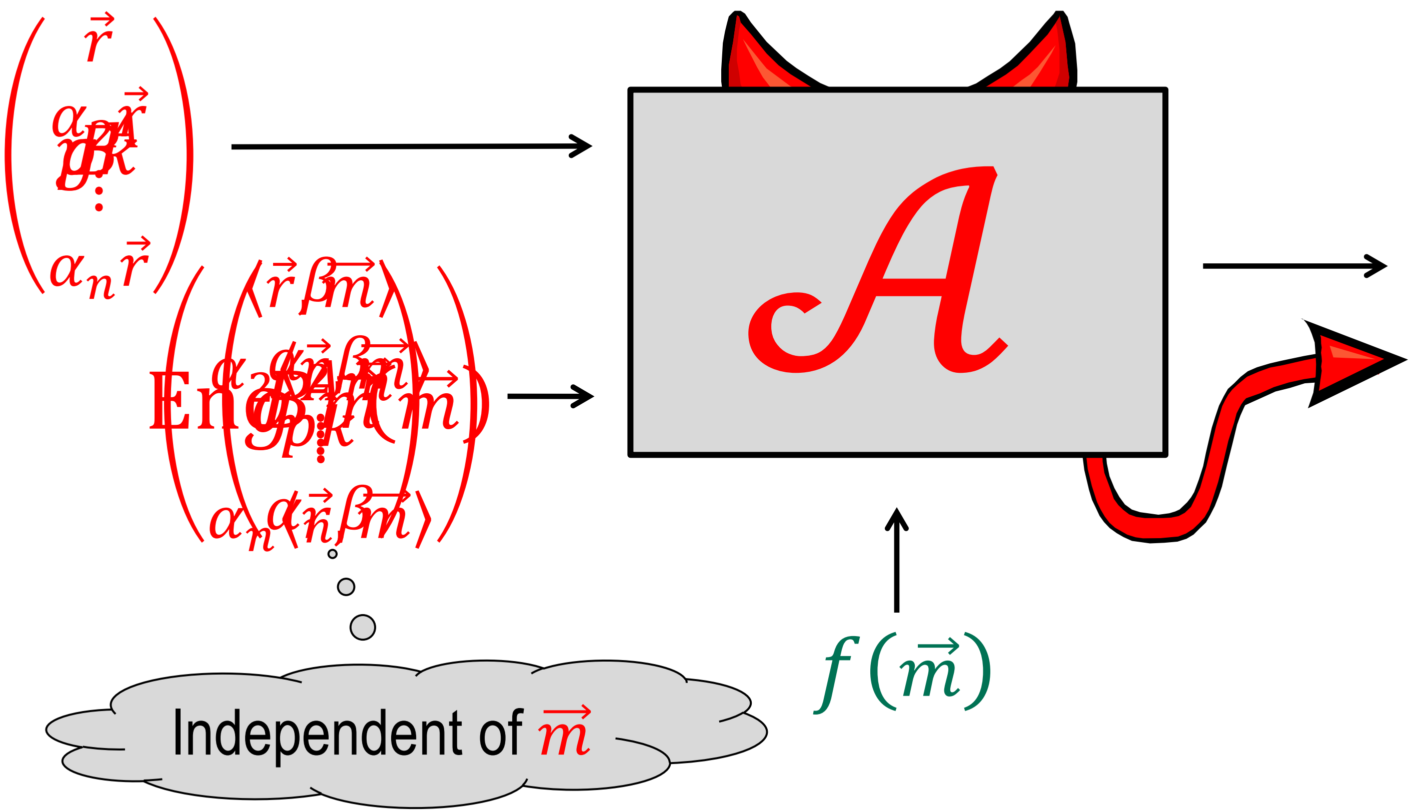
- Given $\vec{m} \in \{0,1\}^n$ output $g^{A\vec{m}} \in \mathbb{G}^n$

$(g^{A\vec{m}})_i = g^{\sum_j a_{ij} m_j} = \prod_j (g^A)_{ij}^{m_j}$

Decryption

- Output $m \in \{0,1\}^n$

Proof of Security



- [BHHO08,NS09]: d-Linear $\Rightarrow$ $g^A \approx_c g^B$ where $rank(B) = d$
- [GL89,DGKPV10]: f is ϵ-hard-to-invert relative to $\mathcal{M}$

  $\Rightarrow (\vec{r}, \langle \vec{r}, \vec{m} \rangle)$ is pseudorandom

Additional Features of Our Schemes

Security for multiple users & related messages

- Any number of users, linearly-related messages
- Without requiring sub-exponential hardness

$$\left(\mathrm{Enc}_{pk_1}(m_1), \dots, \mathrm{Enc}_{pk_n}(m_n)\right)$$

Homomorphic properties

- Additions and one multiplication

$$g^{Am_1} \cdot g^{Am_2} = g^{A(m_1+m_2)}$$

$$e\left(g^{Am_1}, g^{(Am_2)^T}\right) = e(g,g)^{Am_1 m_2^T A^T}$$

Conclusions and Open Problems

- Deterministic encryption in the auxiliary-input setting
- Meaningful security for hard-to-invert auxiliary inputs

Open problems

- Eliminating sub-exponential hardness requirement
- Security beyond linearly-related messages
- Dealing with pk-dependent messages and auxiliary inputs

Thank you!