Algorithmically Random Oracles

Margus Niitsoo

UNIVERSITY OF TARTU
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

CYBERNETICA

Outline

- Random Oracle Model
  - What is it?
  - What is it good for?
- Algorithmically Random Oracles
  - Kolmogorov Complexity
  - Deterministic one-way functions
- Isomorphism result

ROM: What is it good for?

- You have an ID scheme:
  - Alice has a secret sk

sk

ROM: What is it good for?

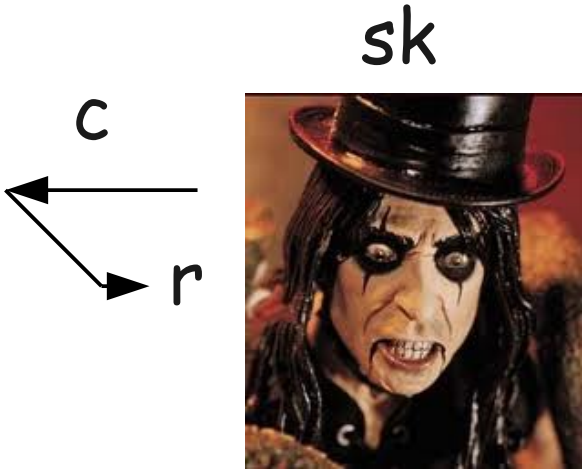
- You have an ID scheme:
  - Alice has a secret sk
  - Alice sends c to Bob

c

sk

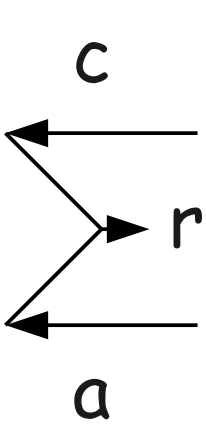
ROM: What is it good for?

- You have an ID scheme:
  - Alice has a secret sk
  - Alice sends c to Bob
  - Bob sends r to Alice



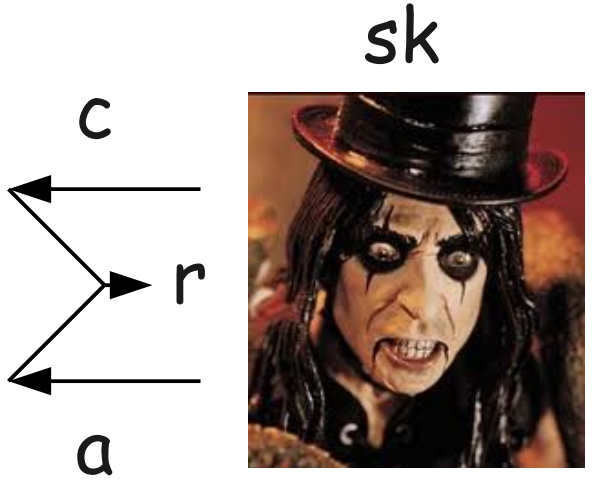
ROM: What is it good for?

- You have an ID scheme:
  - Alice has a secret sk
  - Alice sends c to Bob
  - Bob sends r to Alice
  - Alice sends a to Bob



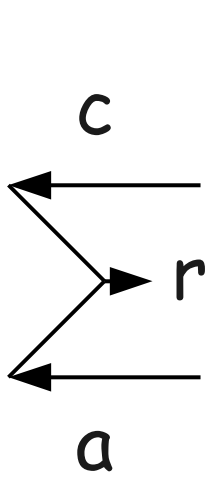
ROM: What is it good for?

- You have an ID scheme:
  - Alice has a secret sk
  - Alice sends c to Bob
  - Bob sends r to Alice
  - Alice sends a to Bob
  - Bob verifies f(pk,c,r,a) = 1



ROM: What is it good for?

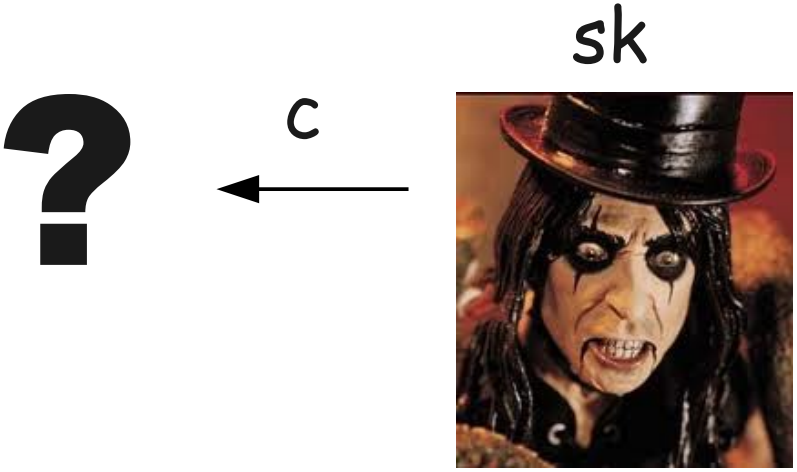
- You have an ID scheme:
  - Alice has a secret sk
  - Alice sends c to Bob
  - Bob sends r to Alice
  - Alice sends a to Bob
  - Bob verifies f(pk,c,r,a) = 1



- We want a Signature scheme!

ROM: What is it good for?

- Signature Scheme
  - Alice has a secret sk
  - Alice generates c

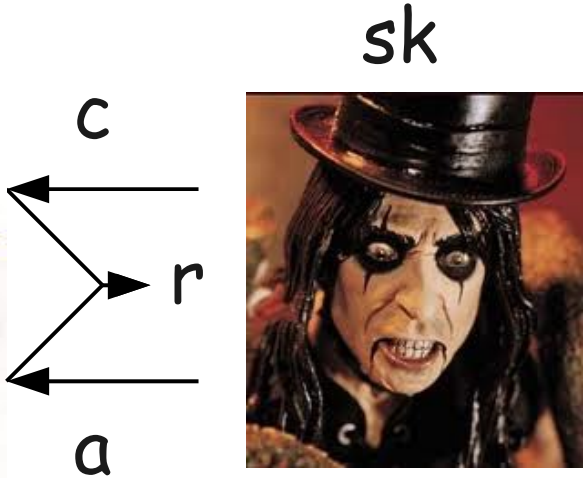


ROM: What is it good for?

- Signature Scheme
  - Alice has a secret sk
  - Alice generates c
  - Computes r=h(d,c)

ROM: What is it good for?

- Signature Scheme
  - Alice has a secret sk
  - Alice generates c
  - Computes r=h(d,c)
  - Alice signs d: (a,r,c)



ROM: What is it good for?

- Signature Scheme
  - Alice has a secret sk
  - Alice generates c
  - Computes r=h(d,c)
  - Alice signs d: (a,r,c)
  - Bob verifies f(pk,c,h(d,c),a) = 1

c

r

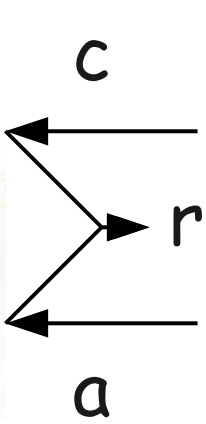
a

sk

(d,a,r,c)

ROM: What is it good for?

- Signature Scheme
  - Alice has a secret sk
  - Alice generates c
  - Computes r=h(d,c)
  - Alice signs d: (a,r,c)
  - Bob verifies f(pk,c,h(d,c),a) = 1



(d,a,r,c)

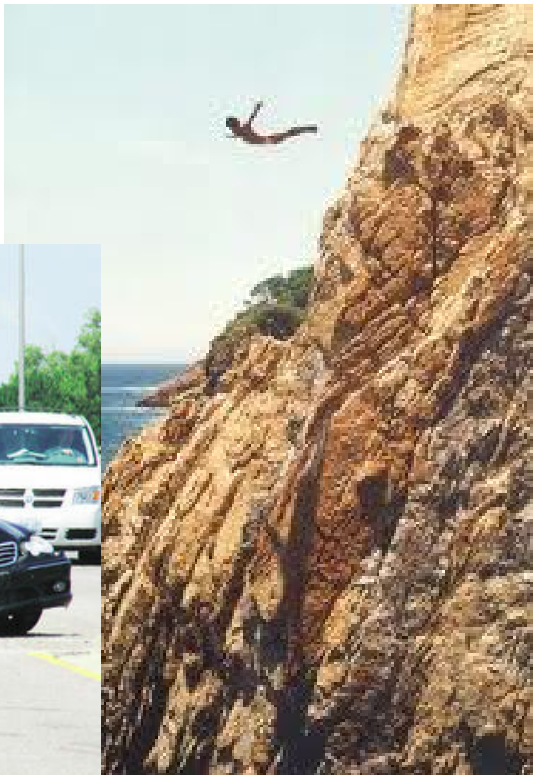
- How good Hash do we need?

ROM: What is it good for?

- We need hash to...
  - Work Fast - so we use SHA-2

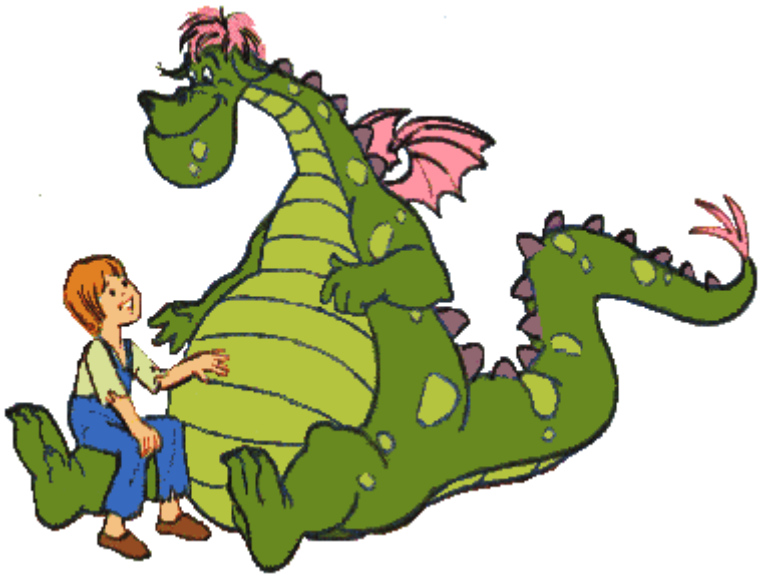
ROM: What is it good for?

- We need hash to...
  - Work Fast - so we use SHA-2
  - One-way? Collision-resistant?



ROM: What is it good for?

- We need hash to...
  - Work Fast - so we use SHA-2
  - One-way? Collision-resistant?
  - Magic (Dwork 03)



ROM: What is it good for?

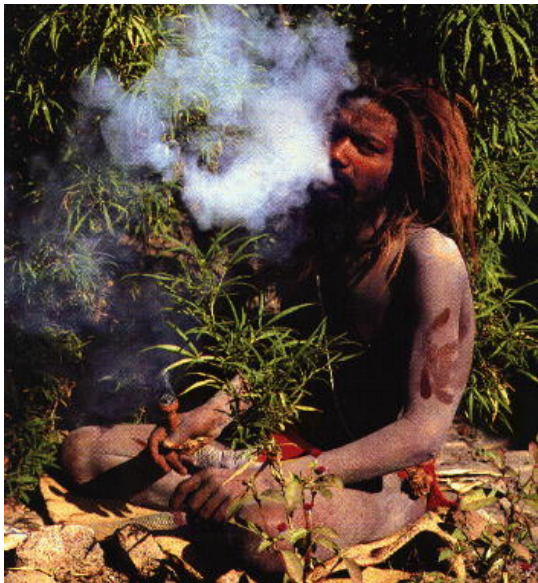
- We need hash to...
  - Work Fast - so we use SHA-2
  - One-way? Collision-resistant?
  - Magic (Dwork 03)
  - Ideally random
    - Answers to queries are independent and uniformly distributed

ROM: What is it?

- We need h to be...
  - Ideally random
    - Answers to queries are uniformly distributed and independent

ROM: What is it?

- We need h to be...
  - Ideally random
    - Answers to queries are uniformly distributed and independent
- Introducing a Random Oracle:

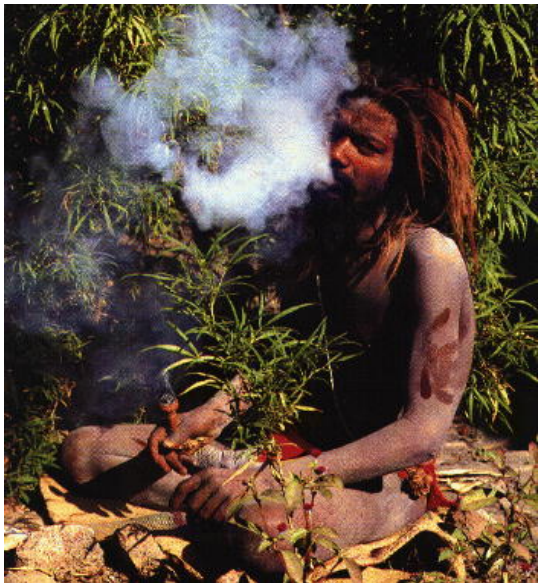


ROM: What is it?

- We need h to be...
  - Ideally random
    - Answers to queries are uniformly distributed and independent
- Introducing a Random Oracle:
  - h chosen uniformly from all h
  - Result holds in ROM

$$\Leftrightarrow$$

$$\Pr_{\{h\}}[\text{Result}] = 1$$



ROM: What is it good for?

- Fiat-Shamir heuristic
  - What we just described
- Simple proofs for strong security
  - Efficient CCA-2 cryptosystem (BR93)
- Also used in Oracle Separation
  - (Proving certain reductions impossible)

ROM: Flipside?

- Family of functions
  - No secure instances (CGH99)
  - Certain things impossible (Nielsen 02)
  - “Oracle Extraction Step” (BJN 09)

Getting rid of Randomness

- Literally “too good to be true”
  - Most functions non-computable!
    - With Probability 1, to be precise

Getting rid of Randomness

- Literally “too good to be true”
  - Most functions non-computable!
    - With Probability 1, to be precise
- Hypothesis:
  Non-computability of h is enough
  - This is almost the case

Kolmogorov Complexity

- Which is more complex?
  - 01001101011100100010
  - 11111111111111111111
- Kolmogorov Complexity C(**s**) :
  Length of the shortest
  program that outputs **s**

Algorithmic Randomness

- **s** is c-random if $C(\mathbf{s}) > |\mathbf{s}| - c$

- An infinite sequence **s** is random if $C(\mathbf{s}_k) > \mathbf{s}_k - c_s$ for all k
  - Such sequences occur with Prob. 1

Alg. Rand. Oracle

- Instead of choosing h randomly
- Fix alg. random sequence
  - Construct h based on that
- “Deterministic random functions”
  - Good one-way functions
    - Proof: If there exists an inverter, we can find a shorter description

AROM vs ROM

Theorem:

A system is secure in ROM

iff

it is secure relative to every AROM

Applications

- Philosophically interesting
  - Secure fixed instantiations do exist!
- Kolmogorov-complexity based security proofs
  - May be conceptually simpler
- Avoid “oracle extraction” step
  - Concise description of the Measure 1 set

Thank you for listening!

Questions? Comments?