Proofs of security for the Unix password hashing algorithm

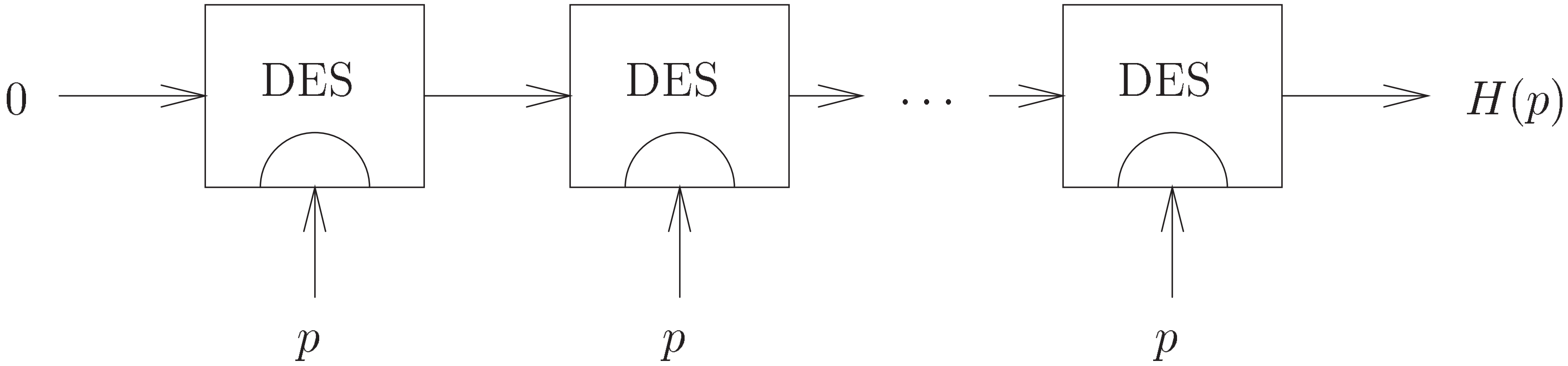
David Wagner and Ian Goldberg

University of California, Berkeley

{daw,iang}@cs.berkeley.edu

The Unix password hash algorithm

- Iterate DES 25 times, using the password p as key:



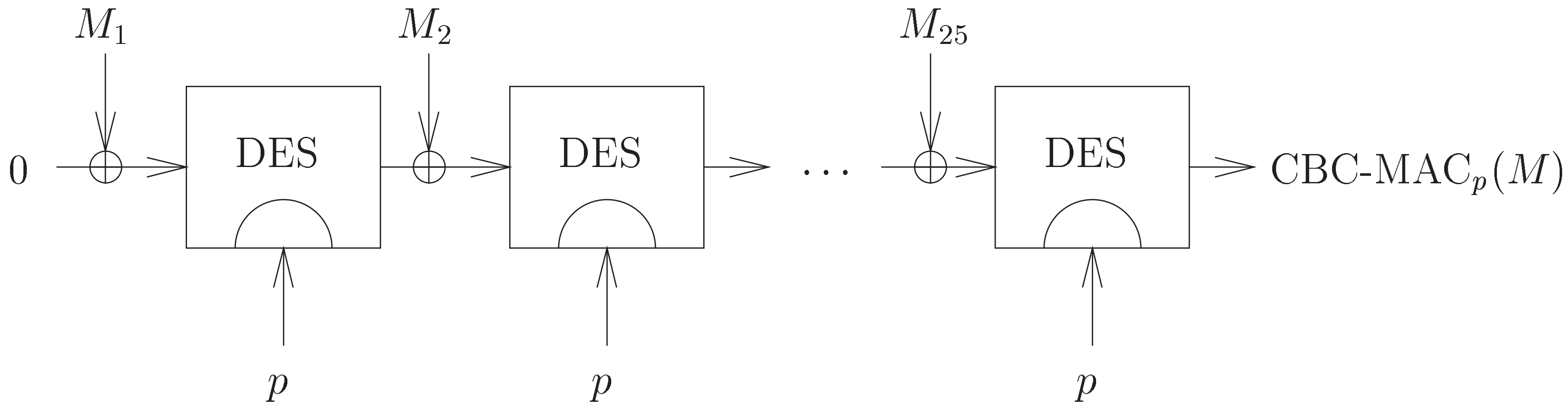
Question. *Is this a secure way to hash passwords?*

(assuming that DES is secure)

Outline of this talk

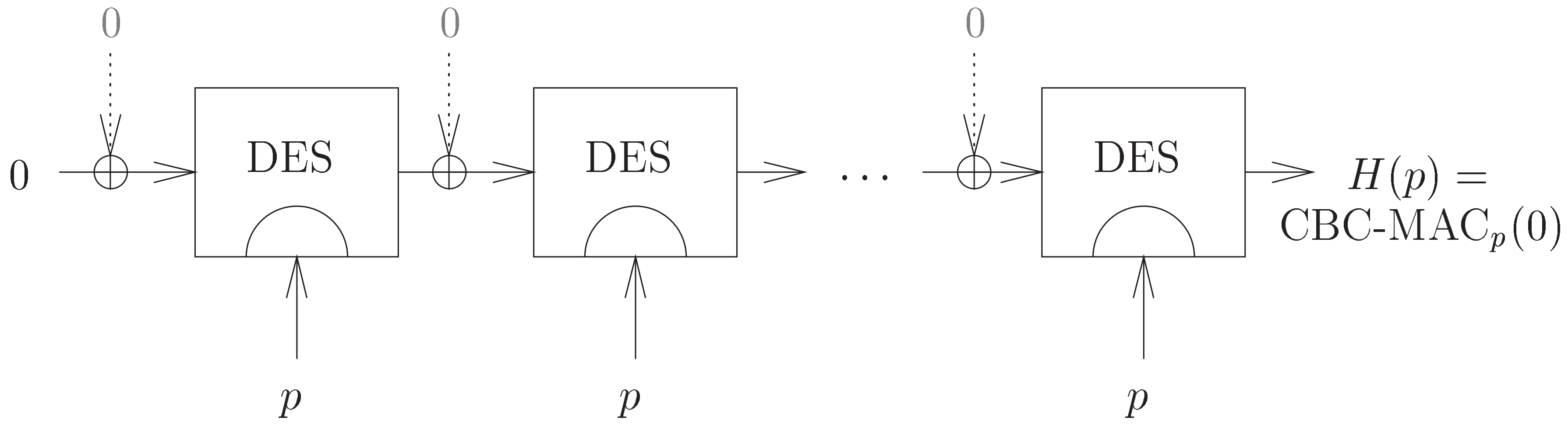
- Theory
  1. Iterated encryption forms a pseudorandom function.
  2. A pseudorandom function yields a pseudorandom generator.
  3. A pseudorandom generator is a one-way function.
- Interpretation
  1. The Unix hash is secure if DES is.
  2. A few concluding remarks.

Previous work: The DES-CBC-MAC



Theorem [BKR]. *DES-CBC-MAC is a secure pseudorandom function if DES is a secure block cipher.*

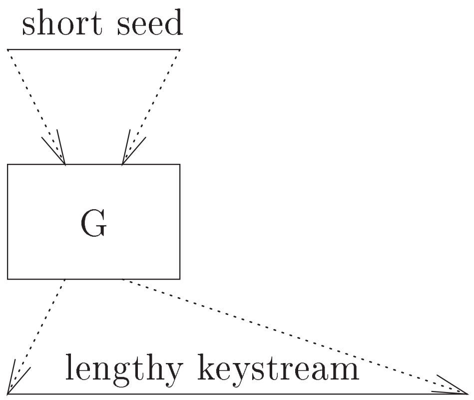
The Unix hash is pseudorandom



Insight (new). The Unix hash may be viewed as a special case of DES-CBC-MAC where the message is fixed at $M = \langle 0, 0, \ldots, 0 \rangle$.

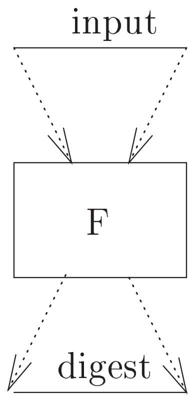
Pseudorandom generators are one-way

A pseudorandom generator



If k is random, $G(k)$ appears random.

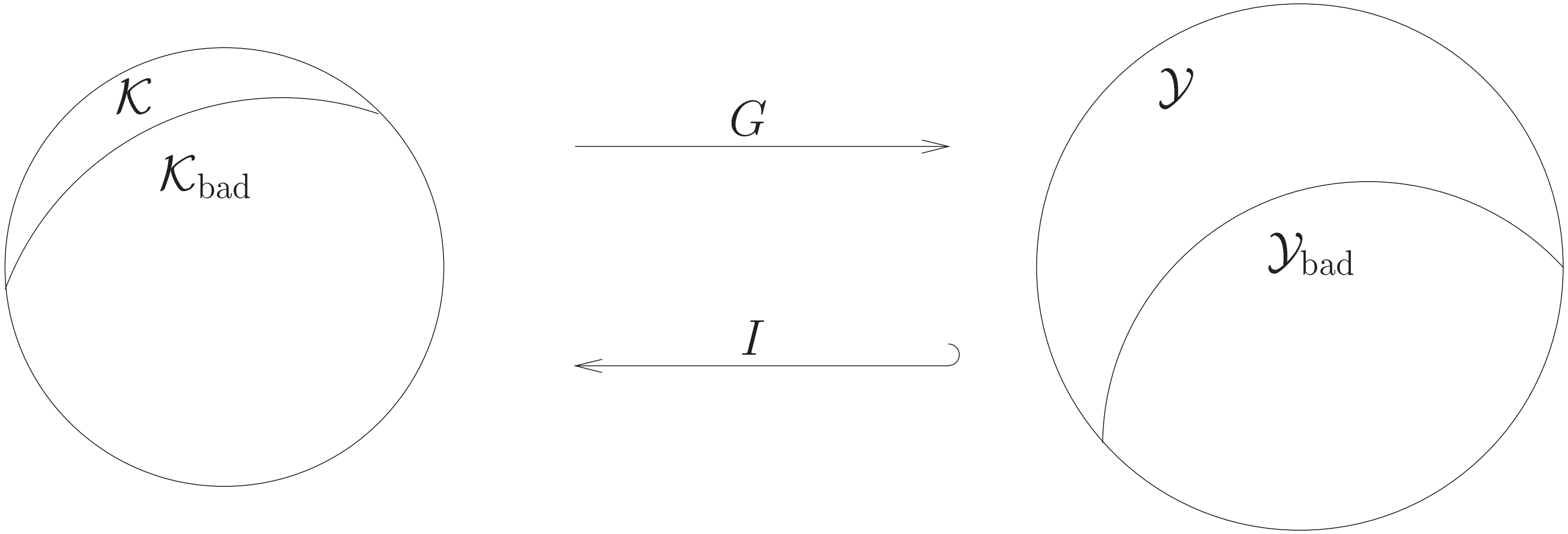
A one-way function



Given $F(x)$, can't find x' with $F(x') = F(x)$.

Theorem [LR]. *Let $G : \mathcal{K} \to \mathcal{Y}$ be a (t, e)-secure pseudorandom generator. Then G is a (t, p)-secure one-way function, for $p = e/(1 - |\mathcal{K}|/|\mathcal{Y}|)$.*

Proof: Every PRG is one-way



Proof. Suppose I finds pre-images for G. Build a recognizer for G-outputs:

$$A(y) = \begin{cases} \text{“non-random”} & \text{if } I \text{ inverts } y \text{ successfully, i.e., } y \in \mathcal{Y}_{\mathsf{bad}} \\ \text{“random”} & \text{otherwise.} \end{cases}$$

Note that $|\mathcal{K}_{\mathsf{bad}}| \geq |\mathcal{Y}_{\mathsf{bad}}|$, since $I : \mathcal{Y}_{\mathsf{bad}} \to \mathcal{K}_{\mathsf{bad}}$ is one-to-one. Also, $|\mathcal{K}| \leq |\mathcal{Y}|$.

Hence, if I inverts G with good probability, A distinguishes G from random.

Combining results

The Unix hash H is a special case of DES-CBC-MAC, and DES-CBC-MAC is a pseudorandom function.

$\Rightarrow H(p)$ is a pseudorandom function.

$\Rightarrow H$ is a pseudorandom generator.

$\Rightarrow H$ is a one-way function.

$\Rightarrow H$ is good for hashing passwords.

Summary of results

THE UNIX PASSWORD HASH

IS SECURE IF DES IS.

(. . . assuming passwords are well-chosen . . .)

Theorem. *If DES is a* $(t, 25, e)$*-secure block cipher, then the Unix hash is a* (t, p)*-secure one-way function, where* $p \approx 1.004 \cdot e$*.*

Contributions

- Justification for the design of the Unix password hash
- Techniques for analyzing iterated encryption

Open problem: Security for non-uniform passwords?