

## 2: Iterated Cryptographic Hash Functions

- ▶ we want *hash function*  $H : (\{0, 1\}^n)^* \rightarrow \{0, 1\}^n$  of *potentially infinite* input size
- ▶ instead we have *compression function*  
 $F : \{0, 1\}^m \times \{0, 1\}^n \rightarrow \{0, 1\}^n$  and define  $H$  via many calls to  $F$
- ▶ One iterated construction (Merkle-Damgård):
  1. fix an “initial value”  $H_0 \in \{0, 1\}^n$
  2. write input  $M$  as a sequence of  $m$ -bit blocks:  $M = (M_1, \dots, M_L)$ ,  
 $M_i \in \{0, 1\}^m$ .
  3. For  $i$  in  $\{1, \dots, L\}$  compute  $H_i := F(H_{i-1}, M_i)$
  4. Output  $H(M) = H_L$ .
- ▶ Variant (“Merkle-Damgård strengthening”): change to
  - 2b. write input  $M$  as a sequence of  $m$ -bit blocks:  $M = (M_1, \dots, M_L)$ ,  
 $M_i \in \{0, 1\}^m$ , where  $M_L$  encodes the length  $L$(typical for 1st generation hash functions)

# The Benefits of Merkle-Damgård

Compression function secure  $\Rightarrow$  hash function secure

## Theorem 3

*Let  $H$  be a MD hash function with compression function  $F$ . Let a collision for  $H$  be given. Then one has either a collision for  $F$ , or a preimage of  $H_0$  for  $F$  (Fixed Point).*

## Theorem 4

*Let  $H$  be a MD hash function with compression function  $F$  and with MD-strengthening. Let a collision for  $H$  be given. Then one has found a collision for  $F$ .*

# Some Disadvantages of Merkle-Damgård

- ▶ “Length-Extension” (→ blackboard)
- ▶ and related structural weaknesses (soon to follow)

# MD4 (“Message Digest 4”)

- ▶ Rivest, 1990
- ▶ compression function with three internal rounds, each round with 16 internal steps, 128-bit output
- ▶ since 1993 not much used in practice (successor: MD5)
  
- ▶ collisions
  - ▶ for MD4 reduced to 2 rounds (Merkle, 1990, Bosselaers, den Boer, 1991)
  - ▶ all 3 rounds (Dobbertin, 1996), *2<sup>20</sup> Ops*
  - ▶ all 3 rounds (Wang et al., 2004) *by hand*
- ▶ (2nd) Preimages for 2 rounds (Dobbertin, 1997)

# MD5 (“Message Digest 5”)

- ▶ Rivest, 1991
- ▶ compression function with four internal rounds, each round with 16 internal steps, 128-bit output
  
- ▶ collisions
  - ▶ for compression function but not for hash function (Dobbertin, 1996)
  - ▶ for hash function ( $2^{39}$  Ops, Wang et al., 2004)
  - ▶ for hash function (a few seconds on a PC)

## Development of Security:

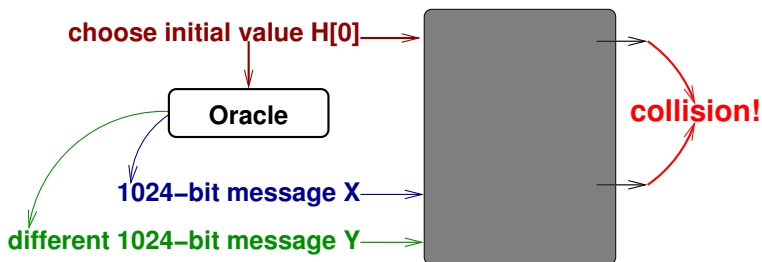
- ▶ 1992: some experts from the european RIPE-project advise against MD5
- ▶ 1996: all experts advise against MD5; this advice has been greatly ignored by the industry
- ▶ 2005: with great hectic, people move away from MD5, still not completely abandoned

# SHA-1 (“Secure Hash Algorithm One”)

- ▶ 1993: SHA (later “SHA-0”), developed by the NIST and the NSA
- ▶ 1995: SHA-1 (minor change), also NIST/NSA
- ▶ 5 internal rounds, 160 bits of output
  
- ▶ 2004: first attacks against SHA-0 (Joux et al.,  $2^{51}$  Ops.)
- ▶ 2005: improved attacks against SHA-0 (Wang et al.,  $2^{39}$  Ops.)
  
- ▶ 2005: first attacks against SHA-1 (Wang et al., claim  $2^{69}$  Ops.)
- ▶ so far, attacks have not been implemented / demonstrated

# Structural Weaknesses of M-D

- ▶ Example: the Wang/Yu attack against MD5 ...



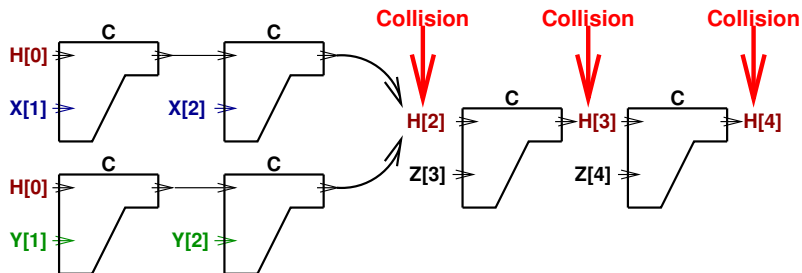
... provides „random“ **Collisions** ( $X[1], X[2]$ ) and ( $Y[1], Y[2]$ ).

**Random collisions should not be too bad**

*“2nd Preimages, yes, these would be damaging!”*

# But ...

Colliding messages can be extended to form **new collisions**:



When we start with a random collision, can we do something to turn it into a “useful” one?



# The exploit (Daum, Lucks, 2005)

0. Alice: internship in Rome.
1. letter of recommendation (LOR);  
on paper
2. sends a file with the LOR and  
asks for a digital signature
3. Caesar compares his LOR to the  
file and then signs it
4. When Caesar is gone to Gaol,  
Alice presents a digitally signed  
order which grants her access to  
secret documents . . .



# What did Alice do?

## Finding a 2nd preimage

letter

... I highly recommend hiring her.

order

... full access to all ... secret information ...

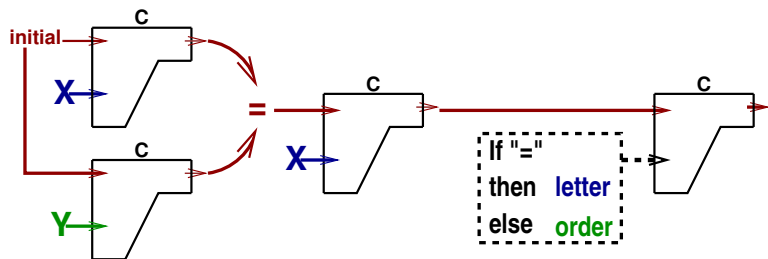
MD5

```
graph TD; A[letter: ... I highly recommend hiring her.] --> MD5[MD5]; B[order: ... full access to all ... secret information ...] --> MD5; MD5 --> C[5421a523481fdc6a2a1c832e72c7b8a5]
```

5421a523481fdc6a2a1c832e72c7b8a5

# No 2nd preimage, only “random” collisions:

- ▶ random  $X \neq Y$  with  $H(X)=H(Y)$  (Wang/Yu).
- ▶ Postscript-Code:  
put  $X$  or  $Y$  onto stack
- ▶ put  $X$  onto stack
- ▶ if two entries on stack are **equal**, then print **letter** else print **order**.



# Two Postscript Documents

This shows the principle, not the postscript syntax

$$\begin{array}{l} \text{preamble; put(X); } \underbrace{\text{put(X); if(=) then } T_1 \text{ else } T_2;}_= \\ \text{preamble; put(Y); } \underbrace{\text{put(X); if(=) then } T_1 \text{ else } T_2;} \end{array}$$

# Similar attacks

- ▶ for documents *without* if-then-else (PDF, pictures, . . .) (Gebhard, Illies, Schindler)
- ▶ for executables (Kaminski, Mikle)
- ▶ for RSA-keys and certificates (Lenstra, Stevens, Wang, de Weger)

# Other exploits of the Merkle-Damgård structure

Once finding any collision has become feasible

- ▶ Joux (2004): multicollisions and attacks against hash cascades



- ▶ The nostradamus attack (Kelsey, Kohno, 2006)



## 2.1: Indifferentiability

# Iterated Hash Functions “Without” Structural Weaknesses

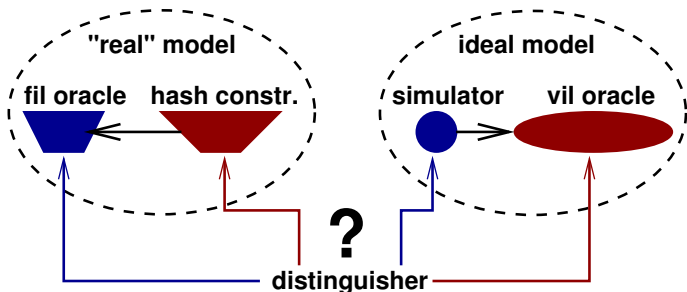
Indifferentiability from a Random Oracle (Coron, Dodis, Mailinaud, Puniya, 2005)

The adversary has a pair of oracles ( $\text{fil}$ ,  $\text{vil}$ ),

	“real”	ideal
$\text{fil}$ (fixed input length) – the compr. function	1. the $\text{fil}$ random oracle $\mathbf{F}$	1. some $\text{fil}$ simulator $\mathbf{S}$
$\text{vil}$ (variable input length) – the hash function	2. the $\text{vil}$ constr. $\mathbf{H}_{\mathbf{F}}$	2. the $\text{vil}$ random oracle $\mathbf{H}$

and tries to distinguish “real” from ideal.

The construction  $\mathbf{H}_{\mathbf{F}}$  is good (“indifferentiable from a random oracle”), if an efficient simulator  $\mathbf{S}$  exists, such that it is hard for all adversaries to distinguish “real” from ideal.

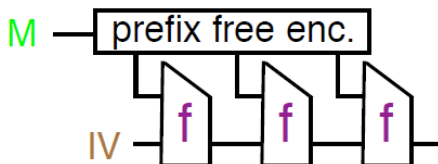


negative examples:

- ▶ plain Merkle-Damgård
- ▶ the two-level construction  $H_F(x) = F(h(x))$  with a collision-resistant  $h$  and a fil random oracle  $F : \{0, 1\}^n \rightarrow \{0, 1\}^n$  (we will revisit this construction some time later)



# Prefix-Free MD and its Structure



- ▶ encode input; no  $M$  is a prefix of another (longer)  $M'$  i.e., there is no  $X$  with  $M' = (M||X)$
- ▶ e.g., append 0 to each internal block and 1 to the last
- ▶ this foils extension attack because output of  $H_F$  does not reveal the entire internal state
- ▶ Is p-f MD indistinguishable from a RO?

## Theorem 5

*Prefix-Free MD is indiffereniable from a random oracle.*

Core ideas for the proof:

- ▶ simulator **S** distinguishes two query types
  - ▶ internal query  $(H_{i-1}, M_i)$ :
    - ▶ give random answer
    - ▶ keep track of chains  $H_0 \rightarrow H_1 \rightarrow \dots \rightarrow H_i$
  - ▶ final-block query  $(H_{i-1}, M_i)$ :
    - ▶ if chain  $H_0 \rightarrow \dots \rightarrow H_{i-1}$   
then **H** $(M_1, \dots, M_{i-1}, M_i)$  for nonrandom part of answer  
else give completely random answer
- ▶ as long as there is at most one single chain for a final query, **S** can give an answer consistent with **H**
- ▶ random responses prevent collisions between chains (for  $\ll 2^{n/2}$ )

# Is “Indifferentiable from ROM” = “No Structural Weakness”

- ▶ Not really!
- ▶ Examples:
  - ▶ Reconsider the two-level construction

$$H_F(x) = F(h(x)).$$

Is it really a bad idea? (Homework)

- ▶ another issue: prove of storage  $H(M||C)$  vs.  $H(C||M)$ 
  - ▶  $H$ : iterated hash function
  - ▶  $M$ : large data stored on server (external)
  - ▶  $C$ : challenge from client

(Homework)

- ▶ But at this point of time, cryptographic theory has no better indication for a hash function not suffering from structural weaknesses – finding a better notion is an open research problem.