Towards a new US government hash function standard

Lars R. Knudsen

May 5, 2009

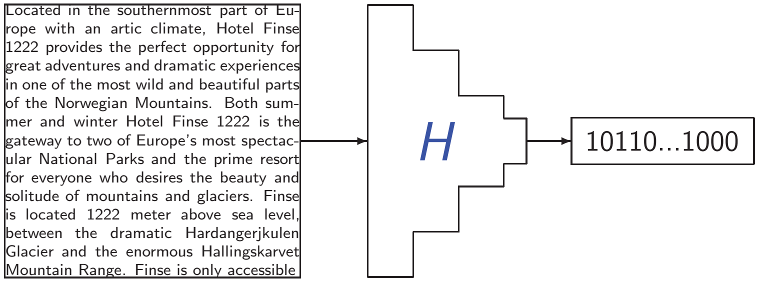
Hash Functions at Finse 1222

1. Lars R. Knudsen, Professor
2. 2001- Technical University of Denmark
3. 2002-2007 University of Bergen, Norway, Professor II
4. 1997-2001 University of Bergen, Norway
5. 1995-1997 Katholieke Universiteit, Leuven, Belgium
6. 1994-1995 Ecole Normal Superieure, Paris, France
7. 1992-1994 PhD at Aarhus University of Denmark

Hash Functions at Finse 1222

1. Introduction
2. Block-cipher based hashing
3. MD4-family, SHA-1, SHA-2, SHA-3
4. Permutation-based hashing and Grøstl

Definition - hash function



$H : \{0,1\}^* \longrightarrow \{0,1\}^n$, for fixed value of n

Cryptographic hash functions

- play a crucial role in cryptography
- many applications
- a hash function
  - is a many-to-one function
  - should appear to be one-to-one in practice

Definition - more

- $H : \{0,1\}^* \longrightarrow \{0,1\}^n$, for fixed value of n
- no secret parameters
- given x, easy to compute $H(x)$
- Often in practice:
  $H : \{0,1\}^M \longrightarrow \{0,1\}^n$, for fixed value of n, big M

Hash Function Applications

- Message authentication codes, HMAC
- Password protection, Unix
- Digital signatures
- As random oracle in various protocols, RSA-OAEP, RSA-PSS, PKCS #1, v2.1
- Pseudo-random generator (key-derivation), DSS
- ...

Hash function properties

$H : \{0,1\}^* \rightarrow \{0,1\}^n$, for fixed value of n

Definitions

- **preimage**, given $H(x)$, find x' s.t. $H(x) = H(x')$
- **2nd preimage**, given x, find $x' \neq x$ s.t. $H(x) = H(x')$
- **collision**, $x \neq x'$, s.t. $H(x) = H(x')$

HMAC - Hash Message Authentication Code

MAC of message x is:

$$\mathrm{MAC}_K(x) = H(K_2 \mid H(K_1 \mid x))$$

With H=SHA-1, K_1 and K_2,derived from 512-bit K:

$$\begin{aligned} K_1 &= K \oplus 363636....36 \\ K_2 &= K \oplus 5C5C5C....5C, \end{aligned}$$

HMAC secure if

- H is collision resistant for secret initial value, **and**
- H is a secure MAC for one-block messages.

Password protection

User id	H(password)
...	...
La, Shangri	09283409283977
Lan, Magel	01265743912917
Lang, Serge	02973477712981
Lange, Tanja	92837540921835
Langer, Bernhard	98240254444422
...	...

- Problem: Parallel attack?!

Password protection, cont.

User id	Salt	H(password, salt)
...	...	...
La, Shangri	68678927431	09283409283977
Lan, Magel	00000000001	01265743912917
Lang, Serge	23092839482	02973477712981
Lange, Tanja	30092341218	92837540921835
Langer, Bernhard	86769872349	98240254444422
...	...	...

It should be “hard” to find preimage of H

Digital signatures (no hashing)

Alice		Bob
m message	$\xrightarrow{sig(m),\, m}$	check signatur

Digital signatures with hashing

Sign $H(m)$ instead of m

Alice		Bob
m message		
compute $H(m)$	$\xrightarrow{sig(H(m)),\, m}$	compute $H(m)$ check signatur on $H(m)$

Attack scenarios - inversion

Given m and $H(m)$, **Eve** finds m' such that $H(m) = H(m')$

Alice		Eve		Bob
$H(m)$	$\xrightarrow{sig(H(m)),\, m}$		$\xrightarrow{sig(H(m)),\, m'}$	compute $H(m')$ check sign. on $H(m')$

It must be "hard" to find 2nd preimages for H.

Attack scenarios - inversion

If **Eve** can forge signature on random-looking message $\tilde{m}$ she may find m such that $H(m) = \tilde{m}$

Alice	Eve		Bob
		$\xrightarrow{sig(H(m)),\, m}$	compute $H(m)$ check sign. on $H(m)$

It must be "hard" to find preimage of H.

Attack scenarios - collisions

Given H, **Bob** finds m and m' such that $H(m) = H(m')$, tricks **Alice** into signing m

Alice		**Bob**
$H(m)$	$\xrightarrow{sig(H(m)), m}$	compute $H(m)$ check signatur on $H(m)$
	..later..	
		you signed m' not m

It must be "hard" to find collisions for H.

Introduction Definition Applications Properties Iterated hash functions

Random Oracle Model

Let $H : \{0,1\}^* \longrightarrow \{0,1\}^n$ be a hash function.
Random Oracle Model:

- the values $H(x)$ are "random": for any x and $y \in \{0,1\}^n$

$$\Pr(H(x) = y) = 2^{-n}$$

- let $\mathcal{X} = \{x_1, \ldots, x_t\}$,
  if $H(x_1), H(x_2), \ldots, H(x_t)$ known by attacker,
  for any $x \notin \mathcal{X}$ and $y \in \{0,1\}^n$

$$\Pr(H(x) = y) = 2^{-n}$$

Introduction Definition Applications Properties Iterated hash functions

Trivial (brute-force) attacks

Preimage attack for $H : \{0,1\}^* \rightarrow \{0,1\}^n$

- given $y = H(x)$
- let $\mathcal{X} = \{x_1, \ldots, x_q\}$
- **for** $x' \in \mathcal{X}$ **if** $H(x') = y$ **then success**

Probability of success:

$$1 - (1 - 2^{-n})^q$$

With $q = 2^n$ probability of success $1 - (1 - 2^{-n})^{2^n} \approx 0.63$

Introduction Definition Applications Properties Iterated hash functions

Trivial (brute-force) attacks

n	$(1 - 2^{-n})^{2^n}$
5	0.6379
10	0.6323
15	0.6321
20	0.6321

q	$1 - (1 - 2^{-n})^q$
2^{n-1}	0.3935
2^n	0.6321
2^{n+1}	0.8647
2^{n+2}	0.9817

Trivial (brute-force) attacks

2nd preimage attack for $H : \{0,1\}^* \to \{0,1\}^n$

- given x and $y = H(x)$
- let $\mathcal{X} = \{x_1, \ldots, x_q\}$, s.t., $x \notin \mathcal{X}$
- **for** $x' \in \mathcal{X}$ **if** $H(x') = y$ **then success**

Probability of success:

$$1 - (1 - 2^{-n})^q$$

With $q = 2^n$ probability of success $1 - (1 - 2^{-n})^{2^n} \approx 0.63$

Trivial (brute-force) attacks

collision attack for $H : \{0,1\}^* \to \{0,1\}^n$

- let $\mathcal{X} = \{x_1, \ldots, x_q\}$,
- let $\mathcal{Y} = \{y_1, \ldots, y_q\}$, where $y_i = H(x_i)$
- **if** $y_i = y_j$ for some $i \neq j$ **then success**

Probability of success:

$$1 - e^{\frac{q(q-1)}{2 \cdot 2^n}}$$

With $q = \sqrt{2} \cdot 2^{n/2}$ one gets probability of success of $1 - e^{-1} \approx 0.63$

Birthday paradox

Choose q elements at random (with replacements) from set of S random elements, where $q \ll S$
Let p be probability of at least one collision

$$\begin{aligned} 1 - p &= 1 \cdot \frac{S-1}{S} \cdot \frac{S-2}{S} \cdots \frac{S-(q-1)}{S} \\ &= \prod_{k=1}^{q-1} \left(1 - \frac{k}{S}\right) \\ &\approx \prod_{k=1}^{q-1} \exp(-\frac{k}{S}) = \exp\left(-\frac{q(q-1)}{2S}\right) \end{aligned}$$

NB. $e^{-x} = 1 - x + \frac{x^2}{2!} - \frac{x^3}{3!} + \frac{x^4}{4!} - \ldots$

Birthday paradox (2)

$$p \approx 1 - \exp\left(-\frac{q(q-1)}{2S}\right) = 1 - e^{-\frac{q(q-1)}{2S}}$$

q	$\approx p$
$1.17\sqrt{S}$	50%
$1.41\sqrt{S}$	63%
$2\sqrt{S}$	86%
$4\sqrt{S}$	99.99%

birthday paradox: $(S, q) = (365, 23)$, $p \approx 1/2$

Birthday paradox used on hash functions

Hash function $H : \{0,1\}^* \to \{0,1\}^n$

1. choose $q = 2^{(n+1)/2} = \sqrt{2} \cdot 2^{n/2}$ randomly chosen inputs each of at least $(n+1)/2$ bits
2. compute hash values for all k inputs

Prob(at least one collision) =

$$p \approx 1 - \exp\left(-\frac{q(q-1)}{2 \cdot 2^n}\right) \approx 1 - e^{-1} \simeq 0.63$$

Cryptographic hash functions - generic attacks

$H : \{0,1\}^* \to \{0,1\}^n$, fixed value of n

attack	rough complexity
collision	$\sqrt{2^n} = 2^{n/2}$
2nd preimage	2^n
preimage	2^n

Today: $n \geq 160$ is recommended
Aim: no better attacks than generic attacks

NB. Given 2^k hashed messages, effort to find 2nd preimage of ≥ 1 of them is 2^{n-k} (Merkle)

Reductions

$H : \{0,1\}^* \to \{0,1\}^n$, for fixed value of n

In random oracle model:

- 2nd preimage attack for $H \Rightarrow$ collision attack for H
- preimage attack for $H \Rightarrow$ collision attack for H

which leads to

- collisions hard $\Rightarrow$ 2nd preimages and preimages hard

Iterated hash functions

Compression function

$$h : \{0,1\}^N \to \{0,1\}^n, N > n$$

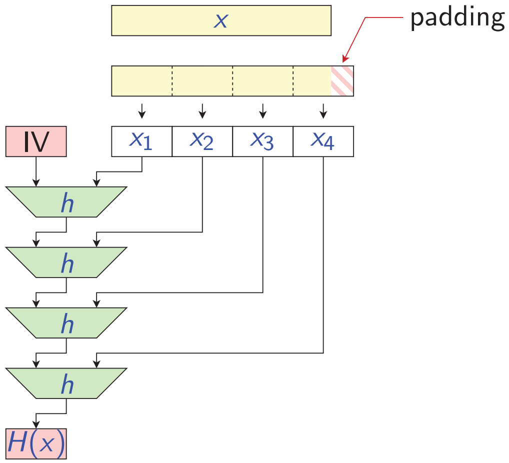
Construct

$$H : \{0,1\}^M \to \{0,1\}^n,$$

where $M >> N$ from h

Iterated hash function

slide by S.S.Thomsen



Extending hash functions - Merkle-Damgård

Padding-rule: $x \neq y \;\Rightarrow\; \text{pad}(x) \neq \text{pad}(y)$
Construct H from h:

1. let $x \in \{0,1\}^v$ be message
2. apply padding rule such that

$$x = x_1 \mid x_2 \mid \ldots \mid x_{t-1} \mid x_t$$

   where x_t full block which contains the integer v as a string
3. define $h_0 = IV$ and $h_i = h(x_i, h_{i-1})$ for $1 \leq i \leq t$
4. define $H(x) = h_t$

Theorem: collision for $H \Rightarrow$ collision for h

Properties of iterated hash functions

- Given hashed messages with 2^k message blocks, effort to find 2nd preimage is $\simeq k2^{n/2} + 2^{n-k}$ (Dean, Kelsey-Schneier)

attack	rough complexity
collision	$\sqrt{2^n} = 2^{n/2}$
2nd preimage	$k2^{n/2} + 2^{n-k}$
preimage	2^n

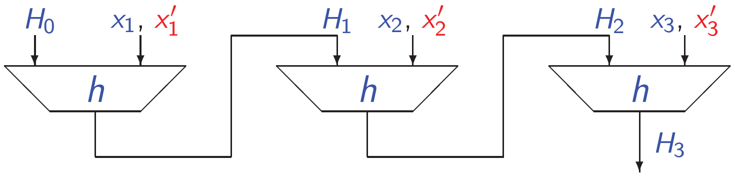
Properties of iterated hash functions (2)

- **Extension attack.** Given $x \neq x'$ such that $H(x) = H(x')$ then $H(x \mid y) = H(x' \mid y)$
- **Multi-collisions.** $x_1, \ldots, x_{2^t}$ s.t.

$$H(x_1) = H(x_2) = \ldots = H(x_{2^t}),$$

  - In general: time $(2^t!2^{n(2^t-1)})^{1/2^t}$
  - For iterated hash functions: time $t2^{n/2}$

Iterated hash functions - multi-collisions



- assume x_1 and x'_1 is a collision for h using H_0
- assume x_2 and x'_2 is a collision for h using H_1
- assume x_3 and x'_3 is a collision for h using H_2
- leads to eight collisions for H
- with hash size $n = 256$, complexity is $\approx 2^{130}$ compared to 2^{226}
- Coppersmith 85

Concatenated hash function - collision

- $F(m) = G(m) \mid H(m)$, where
  - G hash function of n bits
  - H iterated hash function of n bits
- Find $2^{n/2}$-collision on H in multi-collision attack.
- One of these gives collision also for $G \Rightarrow$
  Collision for F with effort $(n/2)2^{n/2}$.
- Joux 2004

Theoretical results on Merkle-Damgård iterated hashing

1. h is collision-resistant $\Rightarrow$ H is collision-resistant (MD)
2. h is random oracle $\Rightarrow$ H is random oracle ?
3. Coron et al 05: construction satisfying 2.
4. Bellare-Ristenpart 06: construction satisfying both 1. and 2.
5. much recent work in this direction

Practical extensions to Merkle-Damgård

- Add output transformation. $\div$ extension attack
- Large internal state (such that $2^{n/2}$ is huge).
  $\div$ multi-collision attack
  $\div$ 2nd preimage attacks
- Add weak second chain (e.g. checksum in MD2)
  Does not protect against 2nd preimage attacks
  (Gauravaram, Kelsey, Knudsen, Thomsen, 2008)
- Lucks (2005)
  - wide-pipe: large internal state, plus compress in output trans
  - double pipe: two parallel chains, combined in output trans
- Counters? Salts?