On the Design of Hash Functions

Lars R. Knudsen

May 8, 2007

Introduction

Iterated hash functions

Based on number-theoretic problems

Block cipher constructions

Definition - hash function

Message

Padding

$x_1, x_2, \ldots, x_{t-1}, x_t$

$150763210262$

$H$

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

Iterated hash functions

Damgård and Merkle (1989)

Build $H : \{0,1\}^* \rightarrow \{0,1\}^n$ from $h : \{0,1\}^m \rightarrow \{0,1\}^n$, $m > n$

1. apply padding such that $x = x_1 | \ldots | x_{t-1}$ and $x_{t-1}$ full block
2. append to $x$ integer $t - 1$ as a string, $x = x_1 | \ldots | x_{t-1} | x_t$
3. define $h_0 = IV$ and $h_i = h(h_{i-1} | x_i)$ for $1 \leq i \leq t$
4. define $H(x) = h_t$

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

Generic attacks

For $H : \{0,1\}^* \rightarrow \{0,1\}^n$ and $h : \{0,1\}^m \rightarrow \{0,1\}^n$, $m > n$

\[
\begin{array}{|c|c|}
\hline
\text{attack} & \text{rough complexity} \\
\hline
\text{collisions} & 2^{n/2} \\
\text{2nd preimages} & 2^n \\
\text{preimage} & 2^n \\
\hline
\end{array}
\]

Goal: generic attacks are best (known) attacks
Number-theoretic, difficult problems

- Factoring: given $N = pq$, find $p$ and $q$, where $p, q$ big, (odd) prime numbers, $p \neq q$
- Discrete logarithm: given $\beta = \alpha^a \mod p$, find $a$, where $p$ prime, $a$ chosen random from $\mathbb{Z}_p - 1$, $\alpha \in \mathbb{Z}_p^*$ primitive
- Note that not all instances of these problems are hard

Based on number-theoretic problems

- $N = pq$, $p \neq q$, large odd primes, $\alpha$ fixed, large order mod $N$
- Public: $N, \alpha$

$$H : \{0,1\}^* \rightarrow \mathbb{Z}_N$$

$$H(x) = \alpha^x \mod N$$

- Collision: $H(x) = H(x') \Rightarrow x - x' = k\phi(N)$.

Based on number-theoretic problems (2)

- Pfitzmann, Van Heijst
- Public primes: $p, q = \frac{p-1}{2}$, s.t. DLP($p$) is hard
- Public primitive elements of $\mathbb{Z}_p$: $\alpha, \beta$ (randomly chosen)

$$h : \mathbb{Z}_q \times \mathbb{Z}_q \rightarrow \mathbb{Z}_p^*$$

$$h(x, y) = \alpha^x \beta^y \mod p$$

- Find a collision for $h \Rightarrow$ compute $\log_{\alpha}(\beta)$

Based on number-theoretic problems (3)

- Goldwasser, Micali, Rivest
- $N = pq$, $p \neq q$, large primes, $a_0, a_1$ random squares modulo $N$

$$h : \mathbb{Z}_N^* \rightarrow \mathbb{Z}_N^*$$

$$h(b, y) = y^2 a_0^b a_1^{1-b} \mod N$$

- Collision gives $x, x'$ such that $x^2 = x'^2 \mod N \rightarrow$ factoring

- More efficient variants with more squares $a_0, \ldots, a_k$, Damgård

Based on number-theoretic problems (4)

- $N = pq$, $p \neq q$, large primes
- MASH-1 (Modular Arithmetic Secure Hash)

$$h_i = ((m_i \oplus h_{i-1}) \lor a)^2 \mod N \oplus h_{i-1}$$

- $a$: 4 most significant bits in every byte are redundant: equal to 1111 (last byte 1010), $a = 0xf00\ldots00$
- MASH-2: replace exponent 2 by $2^8 + 1$

- Claims: preimages $\sqrt{N} = N^{1/2}$, collisions $\sqrt[4]{N} = N^{1/4}$


Number-theoretic hash functions

- Most schemes slow, e.g., no real speed-up for use in digital signature schemes
- Some schemes have unfortunate algebraic properties (may interact badly with other public-key algorithms)
- Open problem to devise efficient “provably” secure hash function
Newer constructions

- **VSH - Very Smooth Hash**
  - Contini, Lenstra, Steinfield, 2005
  - collision ⇒ nontrivial modular square roots of very smooth numbers modulo \( N \) (composite)
  - efficient collision finder implies fast factoring algorithm
- **LASH - A Lattice Based Hash Function**
  - Bentahar, Page, Saarinen, Silverman, Smart 2006
  - based on the problem of finding small vectors in lattices

VSH - iterated hash function

Let \( N = pq \) be a public RSA modulus (\( p \neq q \), both secret)

Let \( p_1, \ldots, p_k \) be public primes such that \( \prod_{i=1}^{k} p_i < N \)

- Let \( m = m_1, m_2, \ldots, m_\ell \) be message, \( m \in \{0,1\} \)
- \( x_0 = 1 \)
- \( x_1 = x_0^2 (p_1^{m_1} p_2^{m_2} \cdots p_k^{m_k}) \mod N \)
- \( x_{j+1} = x_j^2 \prod_{i=1}^{k} p_i^{m_{j+1,i}} \mod N \)
- Hash(m) = \( x_\ell \)

Hash function using a block cipher

Why build on a block cipher?

- **Advantages:**
  - use existing technology
  - transfer security (trust?!) to hash construction
- **Disadvantages:**
  - if “keys” change often, schemes slow (due to key-schedules)
  - weaknesses of block cipher not relevant for encryption

---

**DES & AES**

**DES = Data Encryption Standard**

**AES = Advanced Encryption Standard**

<table>
<thead>
<tr>
<th>System</th>
<th>Year</th>
<th>Block Size</th>
<th>Key Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>DES</td>
<td>1977</td>
<td>64</td>
<td>56</td>
</tr>
<tr>
<td>AES</td>
<td>2001</td>
<td>128</td>
<td>128, 192 or 256</td>
</tr>
</tbody>
</table>
Introduction Iterated hash functions Based on number-theoretic problems Block cipher constructions

Hash rate

Given hash function built from block cipher
\[ e : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n \]

Rate is defined as
\[ \text{rate} = \frac{\kappa}{n} \]

- one-way: no, given \( h_{i-1} \) easy to find \((m_i, h_{i-1})\)
- attacker has full control over block cipher key

Single block hash (Rabin)

\[ e : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n \]

\[ h_{i-1} \quad \text{e} \quad h_i \]

- rate = \( \kappa/n \)
- one-way: no, given \( h_i \) easy to find \((m_i, h_{i-1})\)

Single block hash, case: \( \kappa > n \) (Merkle)

\[ e : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n \]

\[ h_{i-1} \quad \text{e} \quad h_i \]

- \( x_0 \) fixed block
- rate = \( (\kappa - n)/n \)
- one-wayness: given \( h_i \) hard to find \((m_i, h_{i-1})\)
- collision resistance ??

Many hash functions have Davies-Meyer form

- Examples: MD4, MD5, SHAs
- Pros and cons of Davies-Meyer
  - Fixed points easy:
    \[ h_i = e_{m_i}(h_{i-1}) \oplus h_{i-1} \]
    Choose arbitrary \( m_i \), set \( h_{i-1} := d_{\infty}(0) \). Then
    \[ h_i = h_{i-1} \]
    Not possible in Matyas-Meyer-Oseas and Preneel-Miyaguchi
  - Hash rates for Davies-Meyer can be (arbitrarily) high

Double block hash

- Based on \( e : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n \)
- Length of hash, \( 2n \) bits
- Aim: \( 2^n \) security level for collisions
  - PBGV, QG, LOKI-DBH, ...  
  - Parallel-DM, 1993
  - Nandi, Hirose, 2005
A large class of rate 1 hash functions

Consider the double block hash constructions

\[ h_1^j = e_{\mathcal{A}}(B) \oplus C \]
\[ h_2^j = e_{\mathcal{D}}(E) \oplus F \]

where \( A, B, C \) linear combinations of \( m_1^j, m_2^j, h_1^j, h_1^{j-1}, \) and \( h_2^{j-1} \).

\( D, E, F \) are linear combinations of \( h_1^j, m_1^j, m_2^j, h_1^{j-1}, \) and \( h_2^{j-1} \).

- Knudsen-Lai (1993): preimages for all schemes in \( 2^n \)
- Knudsen-Lai-Preneel (1994-5): collisions \( 2^{3n/2} \) or \( 2^{3n/4} \)
- Ideal security not obtained by any schemes of above form

Abreast-DM & Tandem-DM - Lai, Massey 1990

\[ e : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n, \kappa > n \]
\[ f(x, y) = e_{\kappa}(y) \oplus y \]

Abreast-DM scheme:

\[ h_1^j = f(h_1^{j-1} || m_j, h_2^{j-1}) \]
\[ h_2^j = f(m_j || h_1^{j-1} \oplus \overline{h_2^{j-1}}) \]

where \( \overline{h} \) is bitwise complement of \( h \).

Tandem-DM scheme:

\[ h_1^j = f(h_1^{j-1} || m_j, h_2^{j-1}) \]
\[ h_2^j = f(m_j || (h_1^{j-1} \oplus h_2^{j-1}), h_2^{j-1}) \]

Both hash rate 1/2, conjectured security level for collisions \( 2^n \)
Knudsen-Preneel 1996

- Compression function built from:
  - error-correcting codes
  - $t$ small secure compression functions $f_i$
- Split input into small blocks, expand using code
- Different arguments to at least $d$ of the $t$ subfunctions
- Size of hash larger than security level
- Needs output transformation

Knudsen-Preneel, example

$f_i(x, y) = e_i(y) \oplus y$

Compress: $(h^{t-1}_1, \ldots, h^{t-1}_0, m_i) \mapsto (h^t_1, \ldots, h^t_0)$

$\begin{align*}
  h^1_t &= f_1(h^{t-1}_1, h^{t-1}_2) \\
  h^2_t &= f_2(h^{t-1}_1, h^{t-1}_2) \\
  h^3_t &= f_3(h^{t-1}_1, m_i) \\
  h^4_t &= f_4(h^{t-1}_1 \oplus h^{t-1}_2 \oplus h^{t-1}_3, h^{t-1}_2 \oplus h^{t-1}_4 \oplus m_i) \\
  h^5_t &= f_5(h^{t-1}_1 \oplus h^{t-1}_2 \oplus h^{t-1}_4 \oplus m_i, h^{t-1}_1 \oplus h^{t-1}_2 \oplus h^{t-1}_4 \oplus m_i)
\end{align*}$

Constructed from $[5,3,3]$ Hamming code over GF($2^3$): rate $1/5$

Higher rates by using codes over larger fields

Ideal cipher model

- Let $B_{n,k}$ be all block ciphers with a $k$-bit key and $n$-bit blocks,
  $\{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$
- There are $2^n! \approx 2^{n^2}$ bijections on $n$ bits
- It holds that
  $|B_{n,k}| = \left(\frac{2^n!}{2^k}\right)$
- An ideal cipher is randomly selected from $B_{n,k}$

Ideal cipher model, cont.

- proofs in model give protection against generic attacks
- no real-life cipher is an ideal cipher
- "nearly ideal" cipher can be strong for encryption but very weak when used for hashing
- attacker in control of key, can invest time in finding key(s) with certain properties

DES, weak keys, semi-weak keys

- SHACAL-1:
  - block cipher built from SHA-1
  - 160-bit blocks, 512-bit keys
  - best known attacks today:
    - key-recovery attack on SHACAL-1 has complexity $\approx 2^{500}$
    - collision attack on SHA-1 has complexity $\approx 2^{50}$
Nandi et al, 2005

Variant based on block cipher with $\kappa = 2n$

$$e : \{0,1\}^{2n} \times \{0,1\}^{\kappa} \rightarrow \{0,1\}^{\kappa}$$

Yields compression function

$$h : \{0,1\}^{4n} \rightarrow \{0,1\}^{2n}$$

With $\kappa = 2n$, construction has rate 2/3

Knudsen-Muller, 2005
- collision in $2^{2n/3}$, preimages in time $2^n$
- truncation to 2s bits: collisions in $2^{2n/3}$, preimages in $2^s$

Hirose’s double block mode, 2006

$$e : \{0,1\}^\kappa \times \{0,1\}^n \rightarrow \{0,1\}^n$$

$h : \{0,1\}^\kappa \rightarrow \{0,1\}^n$

Hash rate is $(\kappa - n)/2n$
- Collision requires $2^n$ operations assuming $e(\cdot, \cdot)$ is ideal cipher
- With AES-256 (128-bit block, 256-bit key), one gets hash rate 1/2 and security level $2^{128}$ for collisions

Whirlpool - Barreto, Rijmen, 2003

Based on 512-bit, 10-round block cipher $W$ with a 512-bit key

Preneel-Miyaguchi scheme:

$$h_i = W_{h_{i-1}}(m_i) \oplus m_{i} \oplus h_{i-1}$$

$W$ built in AES-style, 8 by 8 byte-matrix state, diffusion layer from MDS code

ISO/IEC 10118-3:2004

Daemen-style hash constructions

- Iterated hash functions
- Compression function invertible or not hard to invert
- Invertible compression function $\sim$ meet-in-the-middle preimage attack with birthday attack complexity
- Radiogatun. Daemen, Peeters, Van Assche 2006
- Grindahl. Knudsen, Rechberger, Thomsen 2007
Concluding remarks

- 1980s: Hash functions based on block ciphers
- 1990s:
  - Dedicated, faster hash functions (Rivest-kickoff)
  - Many broken block cipher based hash function proposals
- 2000s:
  - Many dedicated schemes have been broken in later years
  - Many new constructions
- Future designs more conservative? (thereby slower?)
- Renaissance of block cipher based proposal?