Efficient Collision-Resistant Hashing from Fixed-Length Random Oracles

Tom Shrimpton, Martijn Stam

ECRYPT Hash Workshop 2007

May 25th, 2007
Merkle-Damgård Transform
Hash Function Domain Extension

$H$ Collision resistant $\Rightarrow \mathcal{H}$ Collision resistant
Davies-Meyer Construction

Blockcipher based Hash Function

\[ H(V, M) = \oplus \]

In ideal cipher model: collision resistance of \(2^{n/2}\).
1. Hash Functions
   - Goal
   - Model

2. Related Work
   - Bitwise Construction
   - Rate-1 Impossibility Result

3. Our Construction
   - Description
   - Collision Resistance
   - Other Properties

4. Conclusion
1. **Hash Functions**
   - Goal
   - Model

2. **Related Work**
   - Bitwise Construction
   - Rate-1 Impossibility Result

3. **Our Construction**
   - Description
   - Collision Resistance
   - Other Properties

4. **Conclusion**
Build a Compression Function
Based on Non-Compressing Primitives

\[ M \xrightarrow{n} f \]

\[ V \xrightarrow{n} f \]

\[ H^f(V, M) \]
Build a Compression Function
Based on Non-Compressing Primitives

\[ M \xrightarrow{n} V \xrightarrow{n} H^{f_1, f_2}(V, M) \]
Build a Compression Function
Based on Non-Compressing Primitives

Find a construction $H^{f_1,f_2} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ that:

- has collision resistance close to $2^{n/2}$
  (for random $f_1, f_2 : \{0,1\}^n \to \{0,1\}^n$);
- uses as few calls to $f_1, f_2$
  (thus has a high rate: blocks processed/call).
Two Types of Random Function $f$

**Two-Way Permutation**
- Instantiating $f$ is easy
  (fixed key blockcipher)
- Constructing $H$ looks harder
  (adversary is stronger)

**One-Way Function**
- Constructing $H$ is probably easier
- Instantiating $f$ might be harder
Two Types of Random Function $f$

Two-Way Permutation

- Instantiating $f$ is easy (fixed key blockcipher)
- Constructing $H$ looks harder (adversary is stronger)

One-Way Function

- Constructing $H$ is probably easier
- Instantiating $f$ might be harder
Two Types of Random Function $f$

**Two-Way Permutation**
- Instantiating $f$ is easy (fixed key blockcipher)
- Constructing $H$ looks harder (adversary is stronger)

**One-Way Function**
- Constructing $H$ is probably easier
- Instantiating $f$ might be harder

$$X \xrightarrow{E} \oplus^n \xrightarrow{f(X)}$$
Two Types of Random Function $f$

### Two-Way Permutation
- Instantiating $f$ is easy (fixed key blockcipher)
- Constructing $H$ looks harder (adversary is stronger)

### One-Way Function
- Constructing $H$ is probably easier
- Instantiating $f$ might be harder
Two Complexity Models

Query Complexity
The adversary is computationally unbounded. Count only the queries.

Full Complexity
The adversary is computationally constrained. Count the full cost.
Two Complexity Models

Query Complexity
The adversary is computationally unbounded. Count only the queries.

\[ \Theta(2^{n/4}) \]

Full Complexity
The adversary is computationally constrained. Count the full cost.

\[ \tilde{O}(2^{n/3}) \]
Two Complexity Models

Query Complexity
The adversary is computationally unbounded. Count only the queries.

\[ \Theta(2^{n/4}) \]

Full Complexity
The adversary is computationally constrained. Count the full cost.

\[ \tilde{O}(2^{n/3}) \]

Shrimpton, Stam
Efficient Collision-Resistant Hashing from Fixed-Length Random Oracles
Outline

1. Hash Functions
   - Goal
   - Model

2. Related Work
   - Bitwise Construction
   - Rate-1 Impossibility Result

3. Our Construction
   - Description
   - Collision Resistance
   - Other Properties

4. Conclusion
MD-Iteration
Bitwise Processing

\[ M = (m_1, m_2, \ldots, m_n) \]

\[ V \xrightarrow{n} f \xrightarrow{f} f \xrightarrow{f} f \xrightarrow{n} H^f(V, M) \]

Given \( f_1, f_2 : \{0, 1\}^n \to \{0, 1\}^n \) define

\[ f : \{0, 1\}^{n+1} \to \{0, 1\}^n : \quad f(b||x) = f_{b+1}(x) \]

Not very efficient: rate \( 1/n \).
Rate-1 Impossibility
Black, Cochran, Shrimpton (Eurocrypt’05)

Regardless of $C_{IN}$ and $C_{OUT}$:
Polynomially many queries to find collision on MD-iterated function.
Outline

1. Hash Functions
   - Goal
   - Model

2. Related Work
   - Bitwise Construction
   - Rate-1 Impossibility Result

3. Our Construction
   - Description
   - Collision Resistance
   - Other Properties

4. Conclusion
Rate-1/3 Compression Function

\[ H(V, M) = f_1(M) \oplus f_3(f_1(M) \oplus f_2(V)) \]
Rate-1/3 Compression Function

\[ H(V, M) = f_1(M) \oplus f_3(f_1(M) \oplus f_2(V)) \]

Collision Resistance (in ROM) \( \approx \Theta(2^{n/2}/n) \).
Introducing the Yield

Adversary has $q$ queries to each oracle.
For how many pairs $(V, M)$ can he compute $H(V, M)$? Yield!

\[(M_1, \ldots, M_q) \rightarrow f_1 \]
\[(V_1, \ldots, V_q) \rightarrow f_2 \rightarrow f_3 \rightarrow H_1, \ldots, H_q\]

\[q\] queries

Yield
Introducing the Yield

\[(M_1, \ldots, M_q) \xrightarrow{f_1} H_1, \ldots, H_q \]

\[(V_1, \ldots, V_q) \xrightarrow{f_2} \sum q \xrightarrow{f_3} H_1, \ldots, H_q \]

Adversary has \(q\) queries to each oracle.
For how many pairs \((V, M)\) can he compute \(H(V, M)\)? Yield!
Introducing the Yield

Let \((a_1, \ldots, a_q)\) and \((b_1, \ldots, b_q)\) be queries to two different oracles. 

\[
\begin{align*}
(a_1, \ldots, a_q) & \xrightarrow{f_3} \{H_1, \ldots, H_q\} \\
\text{yield} & \quad \text{q queries}
\end{align*}
\]

Adversary has \(q\) queries to each oracle.
For how many pairs \((V, M)\) can he compute \(H(V, M)\)? Yield!
Introducing the Yield

Adversary has $q$ queries to each oracle. For how many pairs $(V, M)$ can he compute $H(V, M)$? Yield!

Claim: maximizing yield is best adversary can do.
Maximizing the Yield

Query $f_3$ on collisions in $a_i \oplus b_j$.

$k$-way collision $\Rightarrow k$ evaluations of $H$. 
Maximizing the Yield

Query $f_3$ on collisions in $a_i \oplus b_j$.

$k$-way collision $\Rightarrow k$ evaluations of $H$.

$\text{yield}(q) = \Theta(2^{n/2})$ for $q \approx \Theta(2^{n/2}/n)$
Other Properties

Preimage Resistance

\[ \text{yield}(q) \approx 2^n \Rightarrow \text{Effective Preimage Attack} \]

Disappointing: \( q = O(2^{2n/3}) \) suffices for this.

Multicollision Resistance

A single collision in \( f_1 \) leads to collisions for all \( V \).

Thus also to arbitrary multicollisions when MD-iterated.

Find \( k \)-way collision in time \( O(2^{n/2}) \)
(compared to \( O(2^{n/2 \log k}) \) by Joux for generic MD)
Other Properties

Preimage Resistance

\( \text{yield}(q) \approx 2^n \Rightarrow \text{Effective Preimage Attack} \)

Disappointing: \( q = O(2^{2n/3}) \) suffices for this.

Multicollision Resistance

A single collision in \( f_1 \) leads to collisions for all \( V \).
Thus also to arbitrary multicollisions when MD-iterated.

Find \( k \)-way collision in time \( O(2^{n/2}) \)
(compared to \( O(2^{n/2} \log k) \) by Joux for generic MD)
Outline

1. Hash Functions
   - Goal
   - Model

2. Related Work
   - Bitwise Construction
   - Rate-1 Impossibility Result

3. Our Construction
   - Description
   - Collision Resistance
   - Other Properties

4. Conclusion
Conclusion

\[ M \xrightarrow{n} f_1 \]
\[ V \xrightarrow{n} f_2 \xrightarrow{\oplus} f_3 \xrightarrow{n} H(V, M) \]

- \( \approx \Theta(2^{n/2}/n) \) CR hashing from non-compressing primitive.
- Reasonably efficient: 3 function calls per \( n \) bits.
- However, some other properties are suboptimal.

Shrimpton, Stam
Efficient Collision-Resistant Hashing from Fixed-Length Random Oracles