Security Analysis of the Mode of JH Hash Function

Rishiraj Bhattacharyya Avradip Mandal Mridul Nandi

ISI Kolkata

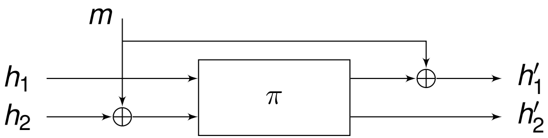
University of Luxembourg

NIST, USA, The George Washington University, USA

JH Hash Function

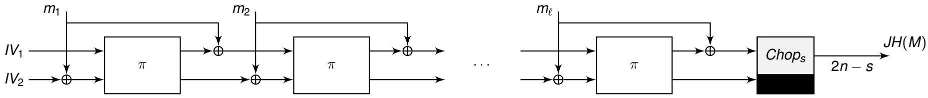
- One of the 14 second round candidates of NIST SHA3 competition.
- Designed by Hongjun Wu.
- Novel design of compression function based on fixed permutation.
- No security proof is known.

JH Compression Function



- $\pi : \{0,1\}^{2n} \rightarrow \{0,1\}^{2n}$ is a fixed permutation.
- $h_1 || h_2$ is $2n$-bit chaining value with each h_i of n bits.
- m is the n-bit message block.
- In SHA3 proposal, $n = 512$.

JH Mode of Operation



- $\textsc{Pad}(M) = m_1 \| m_2 \| \cdots \| m_\ell$.
- Each m_i is of n bits.
- $Chop_s$ chops the last s bits.

JH Padding Rule for $n = 512$

- Length of the message M is $\ell(M)$.
- Append 1 to M.

JH Padding Rule for $n = 512$

- Length of the message M is $\ell(M)$.
- Append 1 to M.
- Append $383 + (-\ell(M) \mathrm{mod} 512)$ bits of 0.

JH Padding Rule for $n = 512$

- Length of the message M is $\ell(M)$.
- Append 1 to M.
- Append $383 + (-\ell(M) \bmod 512)$ bits of 0.
- Append $\ell(M)$ in 128 bits.

JH Padding Rule for $n = 512$

- Length of the message M is $\ell(M)$.
- Append 1 to M.
- Append $383 + (-\ell(M) \bmod 512)$ bits of 0.
- Append $\ell(M)$ in 128 bits.
- Ensures one extra block for padding with 383 bits of Zeros followed by length.

Security of Mode of Operation

Classical Approach Mode of operation should maintain the security of compression function

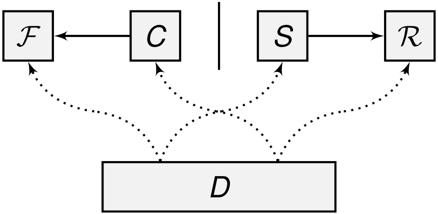
- If compression function is collision resistance then so is the hash function.
- If compression function is PRF then so is the hash function.
- ...

Is this enough??

Security of Mode of Operation

- Random Oracles are popular for proving security of cryptographic protocols.
- In practice, Random Oracles are instantiated by hash functions.
- The (public) domain extension algorithm should maintain the RO-property.
- Indistinguishability is not the applicable as mode of operation is public function.

Indifferentiability of Hash Functions



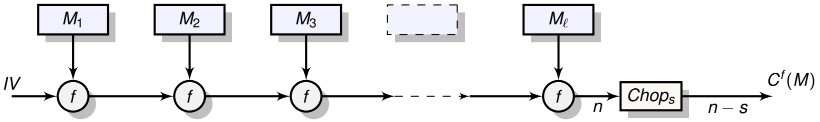
Definition

A Domain Extension algorithm C with oracle access to an ideal primitive $\mathcal{F}$ is said to be $(t, q_C, q_{\mathcal{F}}, \varepsilon)$ indifferentiable from a Random Oracle $\mathcal{R}$ if there exists a simulator S with an oracle access to $\mathcal{R}$ and running time at most t, such that for any distinguisher D, it holds that

$$|\Pr[D^{C^{\mathcal{F}},\mathcal{F}} = 1] - \Pr[D^{\mathcal{R},S^{\mathcal{R}}} = 1]| < \varepsilon$$

The distinguisher makes at most q_C queries to C or $\mathcal{R}$ and at most $q_{\mathcal{F}}$ queries to $\mathcal{F}$ or S.

Previous Works



Theorem (Coron et.al. 2005)

The chop-MD construction (without prefix free padding) based on a Fixed Input Length Random Oracle is $(t_D, t_S, q, \varepsilon)$ *indifferentiable from a Variable Input Length Random Oracle for any* t_D, $t_S = \mathcal{O}(\ell \cdot q^2)$ *and* $\varepsilon = \mathcal{O}(q^2\ell^2/2^n)$ *where* ℓ *is the maximum length of a query made by the distinguisher D.*

Previous Works

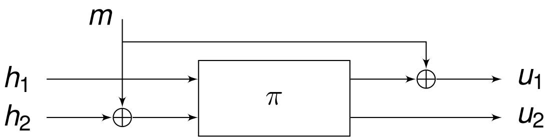
- Chang et.al. in 2006 proved indifferentiability of Double length hash function based on FIL-RO.
- Bellare et. al. in 2006 proved indifferentiability of Envelope Merkle-Damgård construction based on FIL-RO.
- Bertoni et. al. in 2008 proved indifferentiability of Sponge Construction based on random permutation.
- Dodis et. al. in 2009 proved indifferentiability of Tree mode of operations based on Random Permutation.(MD6)

Limits of extending previous results to JH

- JH uses chopMD; Compression function is based on fixed permutation.
- Need to prove indifferentiability of compression function in order to apply Coron et. al. result.
- The compression function is Differentiable even if π is random.

Limits of extending previous results to JH

- JH uses chopMD; Compression function is based on fixed permutation.
- Need to prove indifferentiability of compression function in order to apply Coron et. al. result.
- The compression function is Differentiable even if π is random.

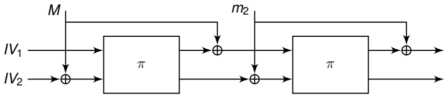


1. Pick random $h_1, h_2 \in \{0,1\}^n$.
2. Query $u_1 || u_2 = f(h_1, h_2 \oplus m)$.
3. Pick random $m \in \{0,1\}^n$.
4. Query $t_1 || t_2 = \pi(-, u_1 \oplus m, u_2)$.
5. If $h_1 = t_1$ and $h_2 \oplus m = t_2$; Return 1.

Our Results

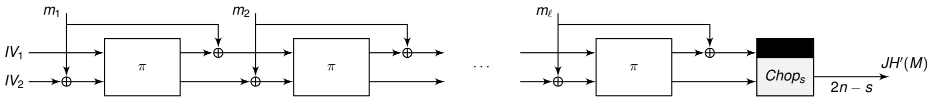
- Assuming π is a random permutation, JH mode of operation with padding is indifferentiable from a Random Oracle.
- Modified JH mode of operation (by chopping other half) is indifferentiable from a Random Oracle.
- Constant query Distinguisher for JH mode without length padding.
- Improved preimage attack of 2^{507} queries on JH mode.

JH mode without specified padding



1. Let $\mathcal{O}_1$ be $\mathcal{R}$ or JH$^\pi$ and $\mathcal{O}_2$ be the simulator or π.
2. $\mathcal{O}_2(+,.)$ denotes forward query and $\mathcal{O}_2(-,.)$ denotes inverse query.
3. $M \in_R \{0,1\}^n$.
4. $h = \mathcal{O}_1(M)$.
5. $t_1 \| t_2 = \mathcal{O}_2(+, h\|0^n)$.
6. $z_1 \| z_2 = \mathcal{O}_2(+, IV_1 \| IV_2 \oplus M)$.
7. $h_2 = \mathcal{O}_1(M \| z_2)$.
8. IF $t_1 \neq h_2 \oplus z_2$
   - return 1.
9. return 0.

JH′ mode of operation



- $\text{PAD}(M) = m_1 \| m_2 \| \cdots \| m_\ell$.
- $\text{PAD}(M)$ is any padding (not necessarily prefix free or length padding) to take care of messages of length not multiple of n.
- Each m_i is of n bits.
- $Chop_s$ chops the first s bits.

Overview of the simulator for JH′

- Maintain a partial permutation.
- If query is in the list, return same answer.
- Maintain list of computable messages from earlier responses and compute the $2n$ bit digest.
- Make sure there is no free computation; i.e. every computable message is queried to simulator.
- For forward query, maintain consistency using the partial permutation and computable messages.
- For inverse query, select the first half of response other than first half of digest of computable messages.
- Maintain permutation property.

Indifferentiability of JH$'$

Theorem

The JH$'$ mode of operation based on a random permutation is indifferentiable from Random Oracle with $Adv_{\mathcal{A}} \le \left(\frac{2\sigma^2}{2^{2n}} + \frac{q^2}{2^n} + \frac{q^2}{2^{\min(s,n)}} \right)$.

Overview of the simulator of JH with padding

- Maintain a partial permutation.
- If query is in the list, return same answer.
- Maintain list of computable messages from earlier responses and compute the $2n$ bit digest.
- Make sure there is no free computation; i.e. every computable message is queried to simulator.
- For forward query, maintain consistency using the partial permutation and computable messages.
- Make sure the first half of response is not equal to some first half of previous input of some length block.
- For inverse queries, the first half of the response is not equal to the first half of digest of computable message.
- Maintain permutation property.

Indifferentiability of JH with padding

Theorem

The JH mode of operation, with specific length padding, based on a random permutation is indifferentiable from Random Oracle with $Adv_{\mathcal{A}} \leq \left(\frac{\sigma^2}{2^{2n}} + \frac{q^3}{2^n} + \frac{q^2\sigma}{2^s}\right)$.

Preimage attack on JH-n with 2^{507} queries

- Let the target image be $h \in \{0,1\}^n$.
- Choose random $h' \in \{0,1\}^n$
- Choose an arbitrary padding block M_5, and compute $H_4 := h_4 \| h'_4 = f^{-1}(h \| h', M_5)$.
- Compute $Q(r)$ candidates for $H_3 = h_3 || h'_3 = f^{-1}(H_4, M_4)$ to obtain r-collision on the last half of H_3.
- Similarly we do it for forward computation of f for the first message block M_1 and get $Q(r)$ candidates for $H_1 = h_1 || h'_1$.
- Perform meet in the middle attack for h_3 and h'_1 by finding collisions between $Q(r)$ candidates of H_1 and H_3 and look fot collision at H_2.

Conclusions

- JH mode of operation with padding based on random permutation is indifferentiable from Random Oracle.
- The security bound for some cases ($s \geq 3n/2$) is beyond the birthday barrier; hence collision is not enough to differentiate
- Length Padding is essential for indifferentiability of JH.
- Chopping different bits give us a new secure mode.
- A new preimage attack on JH mode of complexity 2^{507}.

Thank You