New Key-Recovery Attacks on HMAC/NMAC-MD4 and NMAC-MD5

Lei Wang, Kazuo Ohta and Noboru Kunihiro*

The University of Electro-Communications

* The University of Tokyo at present.

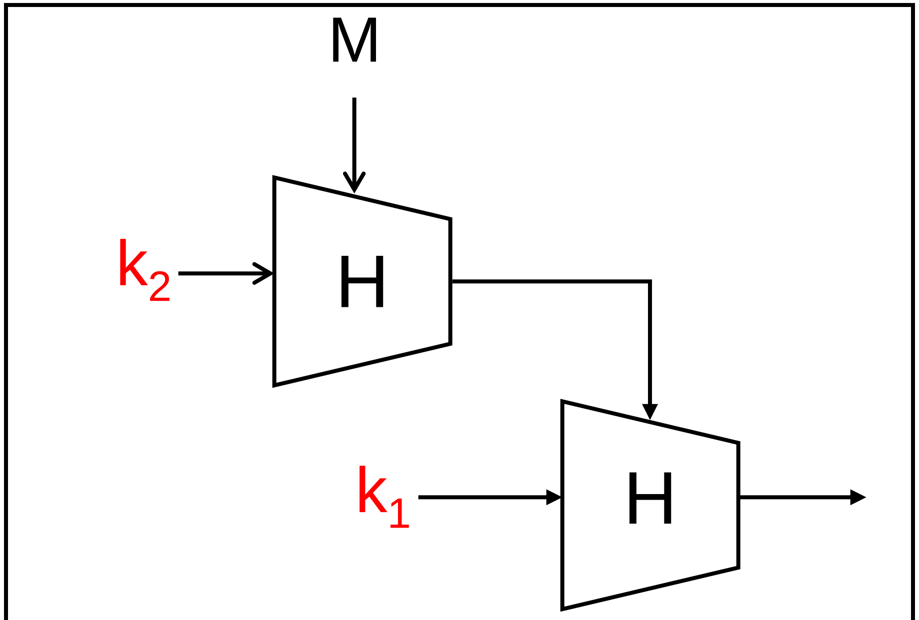
Motivation of This Research

- HMAC has been widely applied in many protocols including SSL, TLS, SSH, IPSec and so on.

- NMAC is theoretical foundation of HMAC: attacks on NMAC (without related-key setting) can be applied to HMAC.

In this presentation, we will pick NMAC as an example.

Structure of NMAC



M: massage;

k_1: the outer key;

k_2: the inner key;

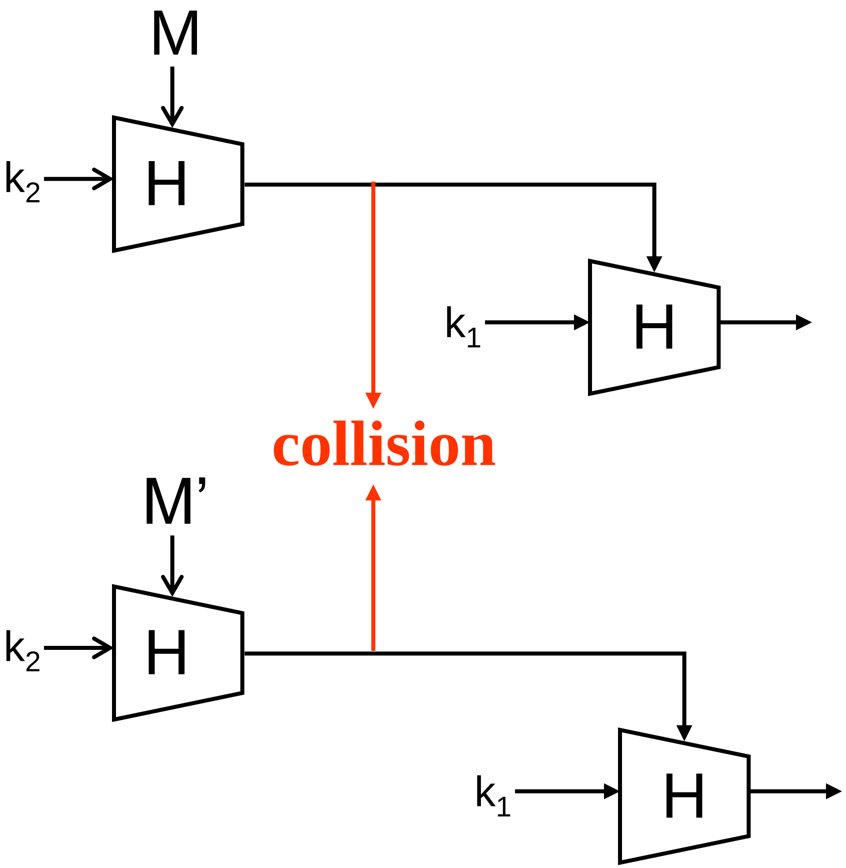
One structural weakness of NMAC based on iterating hash functions: k_1 and k_2 can be recovered separately.

Corresponding hash functions will be the inner and outer hash functions.

General Key-Recovery Attacks on NMAC

- Proposed by Preneel and van Oorschot in 1999:

Crucial idea: generate a collision in the inner hash function by the birthday attack.



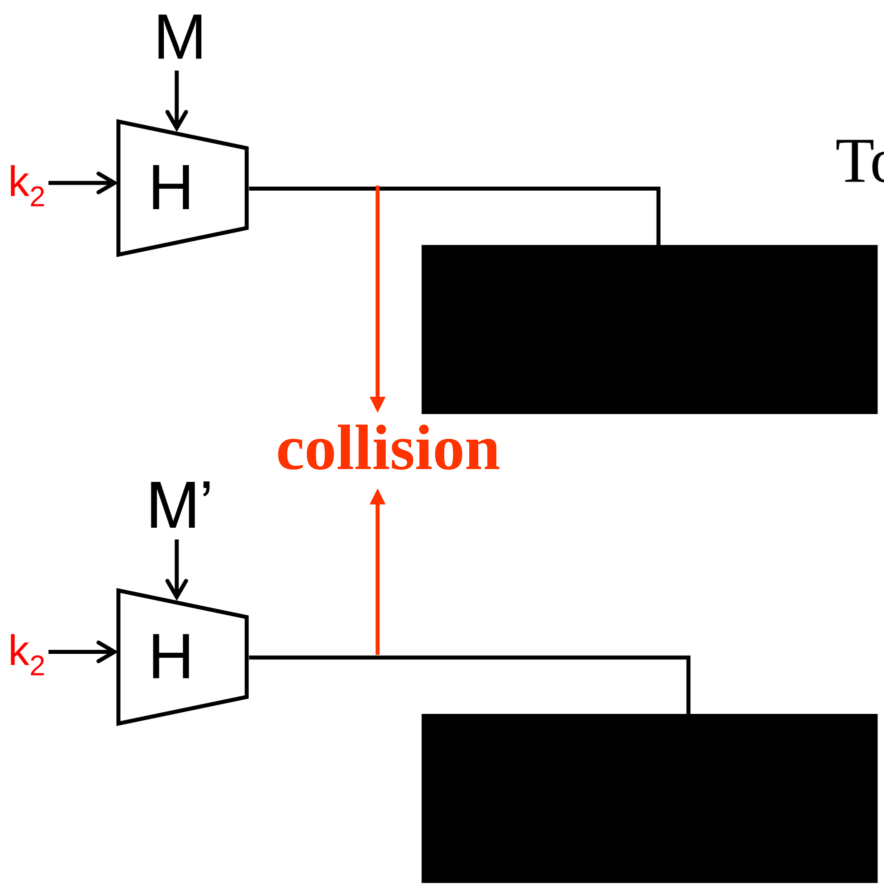
1. Obtain one pair messages (M, M’) cause collision of NMAC.

2. Randomly generate r, and check whether (M||r, M’||r) collide. If collision does not happen, repeat steps 1 and 2.

General Key-Recovery Attacks on NMAC

- Proposed by Preneel and van Oorschot in 1999:

  Crucial idea: generate a collision in the inner hash function by the birthday attack.



To recover k_2:

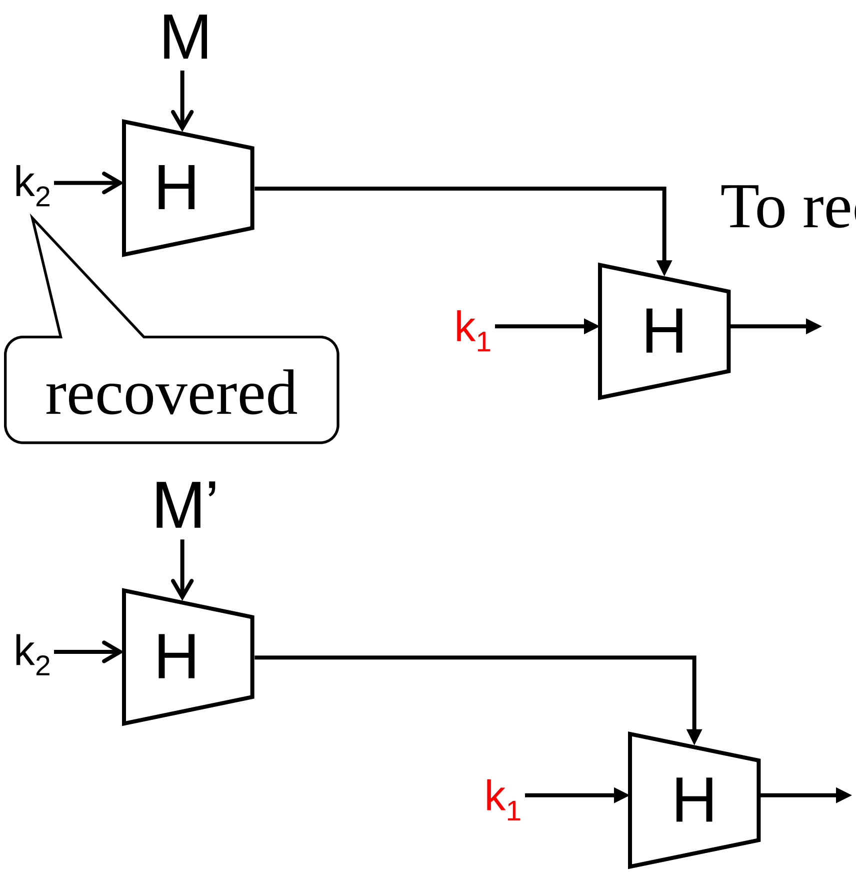
1. guess the value of k_2.

2. check whether the guessed k_2 can satisfy that (M, M') cause the inner collision.

General Key-Recovery Attacks on NMAC

- Proposed by Preneel and van Oorschot in 1999:

  Crucial idea: generate a collision in the inner hash function by the birthday attack.

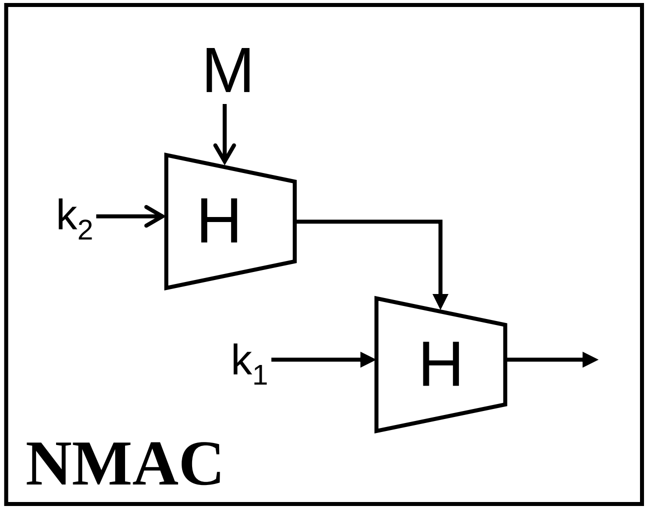


To recover k_1 after k_2 has been recovered:

1. guess the value of k_1.

2. check whether the guessed k_1 can satisfy that NMAC(M) using guessed k_1 is the same with orginal value.

Security Boundary of NMAC



Suppose bit-length of hash value and secret keys is n:

Whatever H is,

One collisi

a high prol

If underlying hash function is weak, more powerful key-recovery attack is possible.

Both secret keys of NMAC can be recovered with $2^{n/2}$ online queries and 2^{n+1} offline computations.

Key-Recovery Attacks on NMAC

Wang et al. revealed weakness of several hash functions from MD4 family, which leaded to key-recovery attacks on NMAC based on specific weak hash functions:

- At Asiacrypt 2006, Contini and Yin proposed inner-key recovery attacks on NMAC instantiated with MD4, MD5, SHA-1.

- At Crypto 2007, Fouque, Leurent and Nguyen proposed full-key recovery attacks on NMAC-MD4 and NMAC-MD5.

- At Financial Crypt 2007, Rechberger and Rijmen proposed full-key recovery attacks on NMAC-MD5 and NMAC-SHA-1.

Key-Recovery Attacks on NMAC

Wang et al. revealed weakness of several hash functions from MD4 family, which leaded to key-recovery attacks on NMAC based on specific weak hash functions:

- At Asiacrypt 2006, Contini and Yin proposed inner-key recovery attacks on NMAC instantiated with MD4, MD5, SHA-1.

- At Crypto 2007, Fouque, Leurent and Nguyen proposed full-key recovery attacks on NMAC-MD4 and NMAC-MD5.

- At Financial Crypt 2007, Rechberger and Rijmen proposed full-key recovery attacks on NMAC-MD5 and NMAC-SHA-1.

Related to our research.

Framework of Key-Recovery Attacks

1. Online work.

Secret key will be partially recovered by online queries.

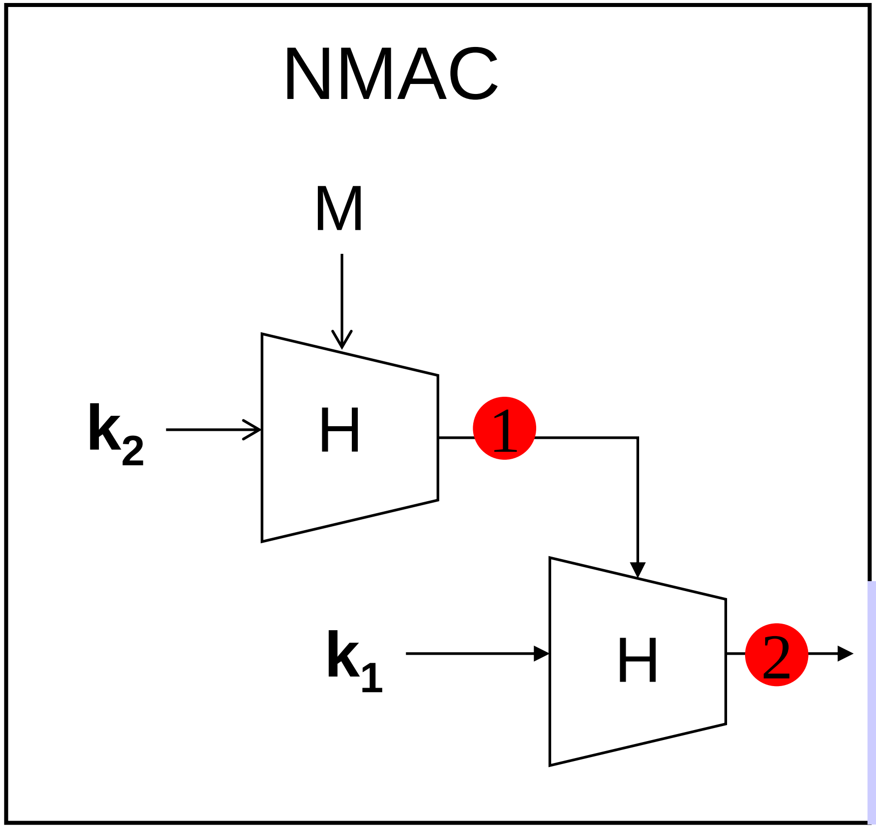
Theoretically interesting! In this presentation, we only focus on online work.

2. Offli

The remaining part of secret key will be recovered by the exhaustive search.

Previous Outer-Key Recovery Attack

Previous outer-key recovery attacks on HMAC/NMAC-MD4 and NMAC-MD5:



1: the value is known based on the knowledge of k_2.

2: detect whether collision happens.

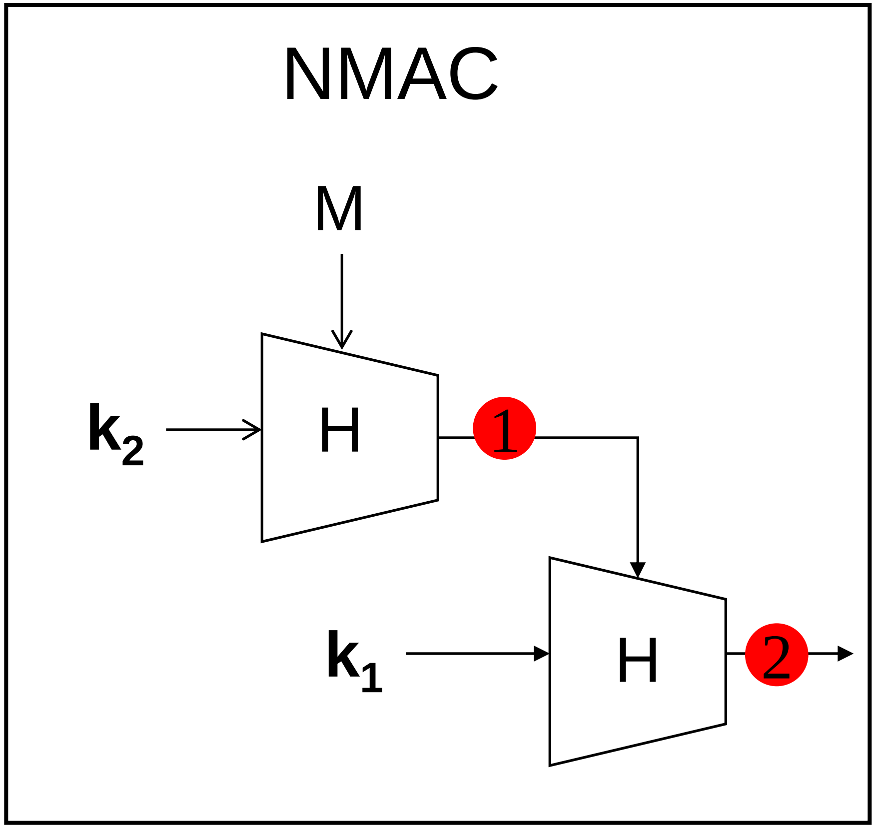
MD4: set conditions on k_1 for collision attack.

If collision happens with expected number of pair queries, k_1 can satisfy the conditions;

Otherwise, k_1 can not satisfy the conditions.

Previous Outer-Key Recovery Attack

Previous outer-key recovery attacks on HMAC/NMAC-MD4 and NMAC-MD5:



(1): the value is known based on the knowledge of k_2.

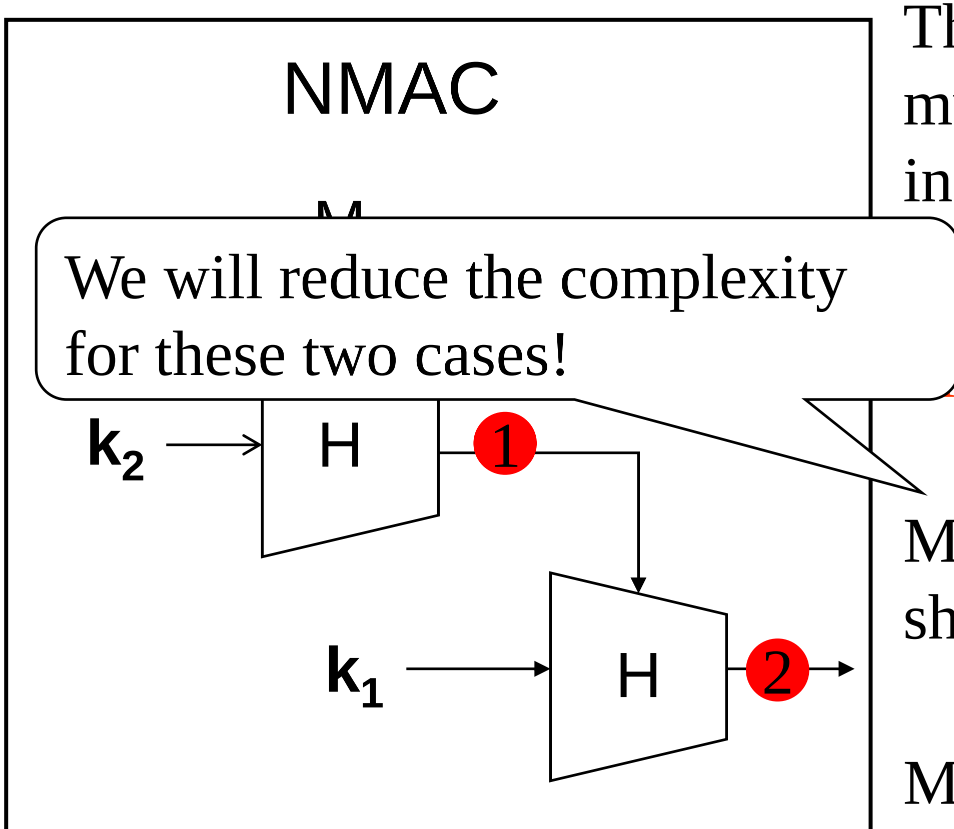
(2): detect whether collision happens.

MD5: recover internal states in outer MD5, and then inverse calculate k_1.

modifying the value at point (1) to set conditions on internal states.

If collision happens with expected number of queries, internal states satisfy conditions.

Analysis of previous work



The outer-key recovery attack is much more expensive than the inner key recovery attack.

lain reason: control ability and eedom lost at point ❶.

MD4: pre-determined pair difference should be generated.

MD5: partially pre-fixed pair values should be generated during modifiying inner hash values.

Advantages of Our Attack (MD4)

HMAC/NMAC-MD4:

Previous work:

Point 1 : generate pre-determined pair difference.

Differential attack: real collision.

Our work:

Point 1 : the same with previous work.

Differential attack: near collision attack, which reduces the complexity, since generating one near-collision needs less pair queries. Moreover it can recover more bit- values.

Advantages of Our Attack (MD4)

	[FLN 07]	Our Work
Online complexity	2^{88}	2^{72}
#bits by online	22	51
Offline comlexity	2^{95}	2^{77}
Total complexity	2^{95}	2^{77}

HMAC/NMAC-MD4: both online and offline complexities have been improved.

Advantages of Our Attack (MD5)

the number of pre-fixed values will increase with the number of recovered bits.

NMAC-MD5:

Previous work:

Point ❶ : generate partially pre-fixed values.

Differential attack: real collision (FLN work), near-collision (RR work).

Our work:

Point ❶ : not necessary (online work). k_1 can be recovered partially without the knowledge of k_2 at all.

Differential attack: near-collision.

Advantages of Our Attack (MD5)

NMAC-MD5:

Complexity becomes higher than the exhaustive search to recover remaining bits after 28 bits recovered.

Previous work:

Point 1 : generate partially pre-fixed values.

Differential attack: real collision (FLN work), near-collision (RR work).

Up to 53 bits can be recovered.

Our work:

Point 1 : not necessary (online work). k_1 can be recovered partially without the knowledge of k_2.

Differential attack: near collision attack.

Usage of Near-collision attacks

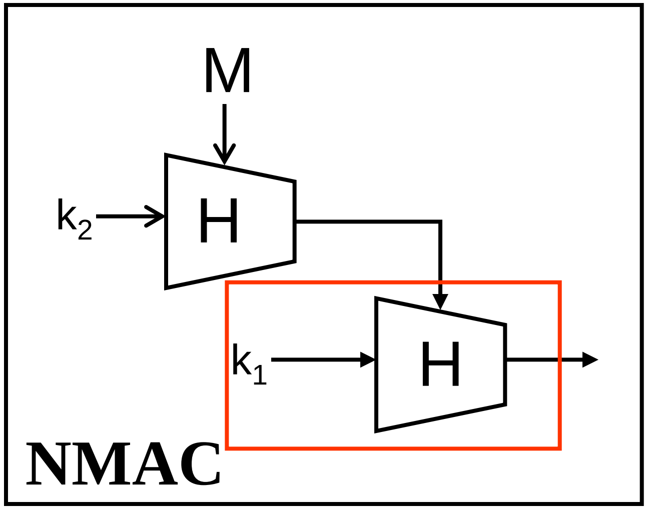
In Financial Cryptography 2007, Rechberger and Rijmen utilized near-collisions on MD5 to recover the outer key of NMAC-MD5, which might be **the first usage** of near-collision to attack HMAC and NMAC.

Advantages of Our Attack (MD5)

	[FLN 07] [RR 07]	Our Work
Online complexity	2^{51}	2^{75}
#bits by online	28	53
Offline comlexity	2^{100}	2^{75}
Total complexity	2^{100}	2^{76}

NMAC-MD5: more bit-values can be recovered by online work. The outer key can be partially recovered without the knowledge of the inner key.

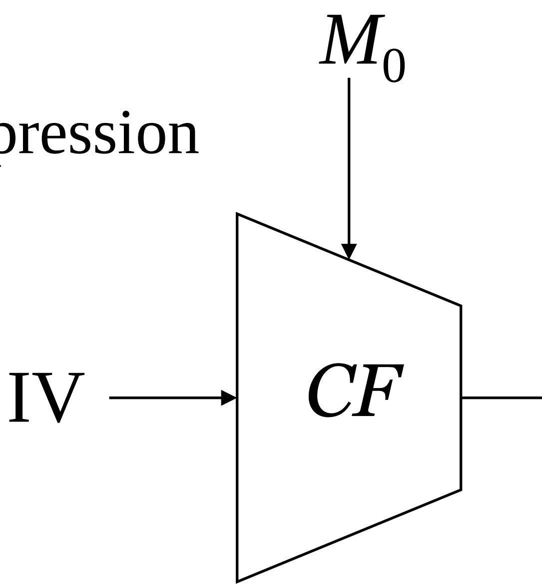
One Novelty of Our Attack



A new approach of key-recovery technique: utilizing feed-forward operation.

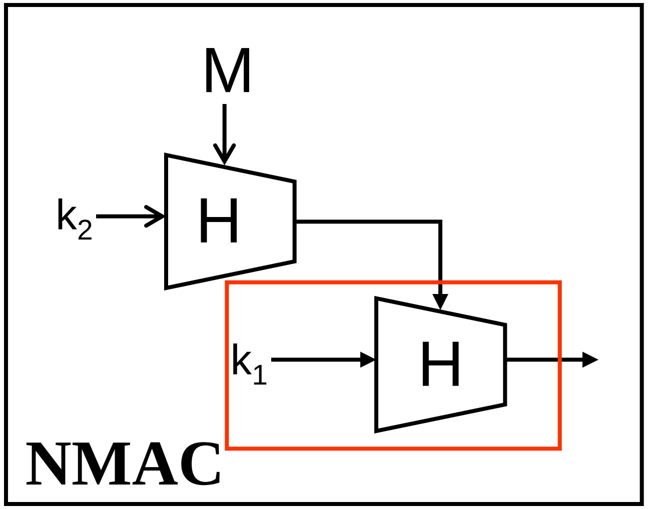
The inner hash value after padding is only one block:

CF: compression function.



[redacted]

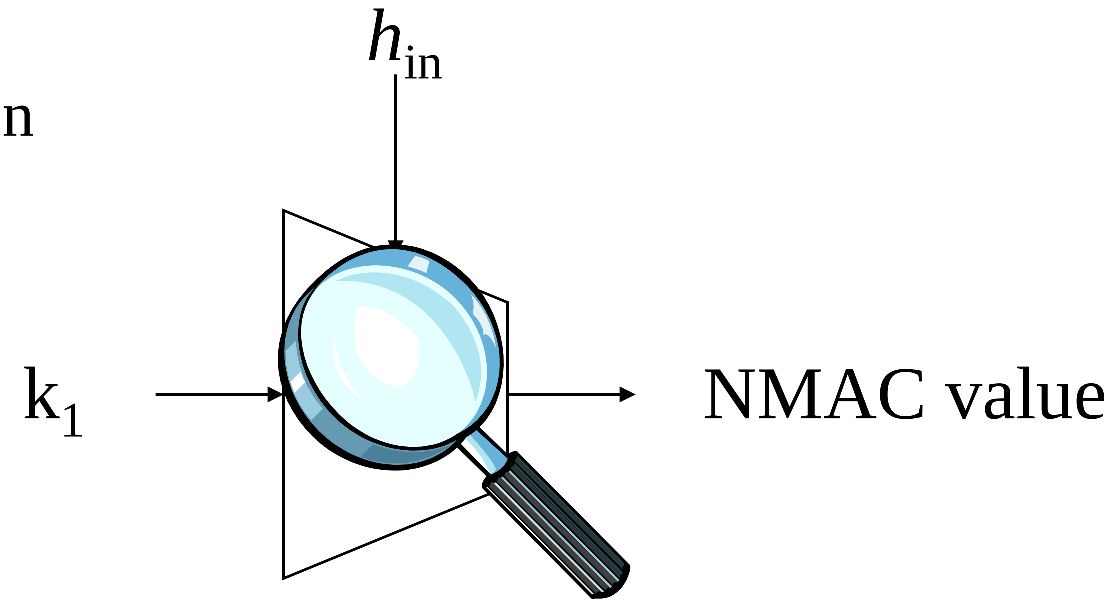
One Novelty of Our Attack



A new approach of key-recovery technique: utilizing feed-forward operation.

The inner hash value after padding is only one block:

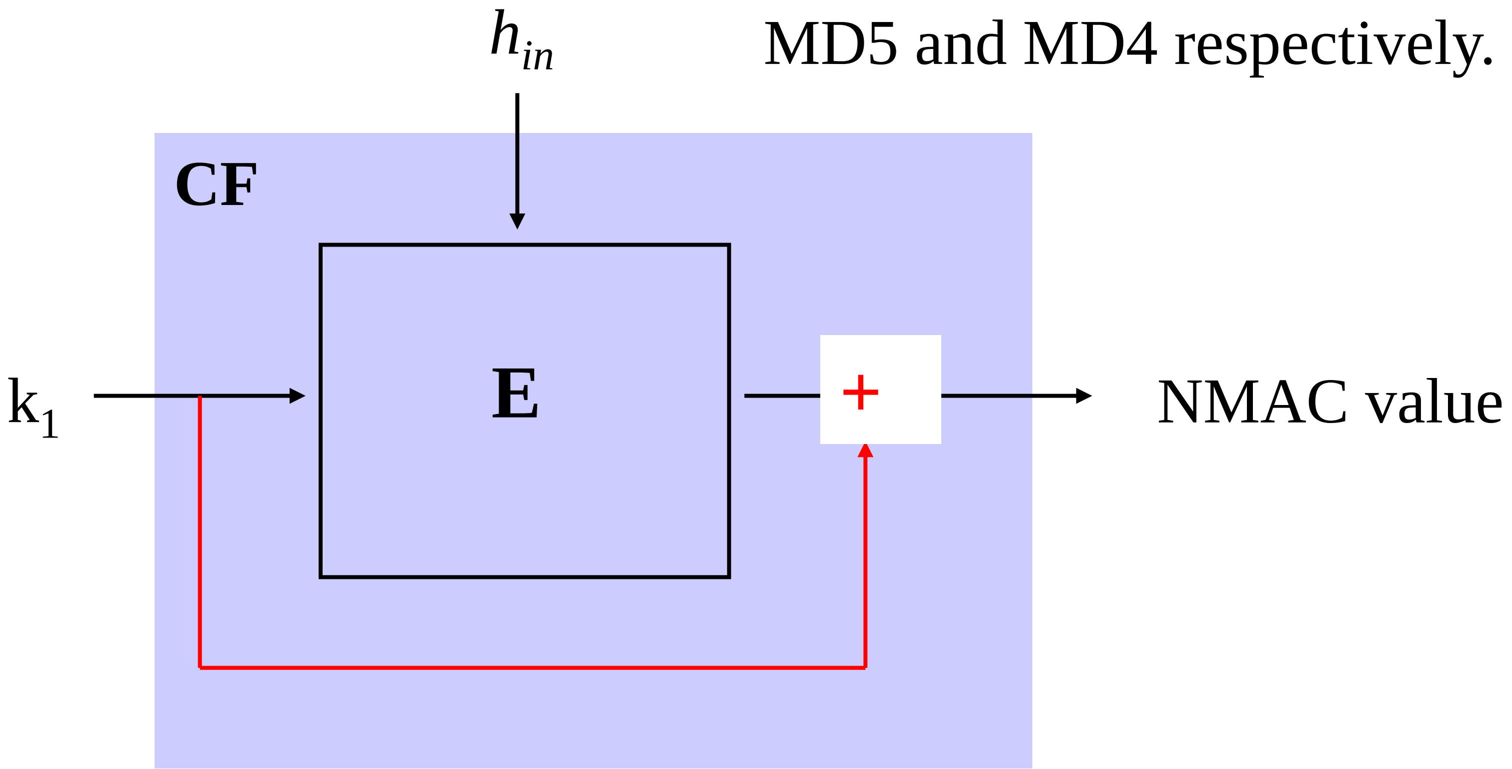
CF: compression function.



CFs of MD4 and MD5

CFs of MD5 and MD4 :

E denotes *n*-step updating functions: *n* is 64 and 48 for MD5 and MD4 respectively.



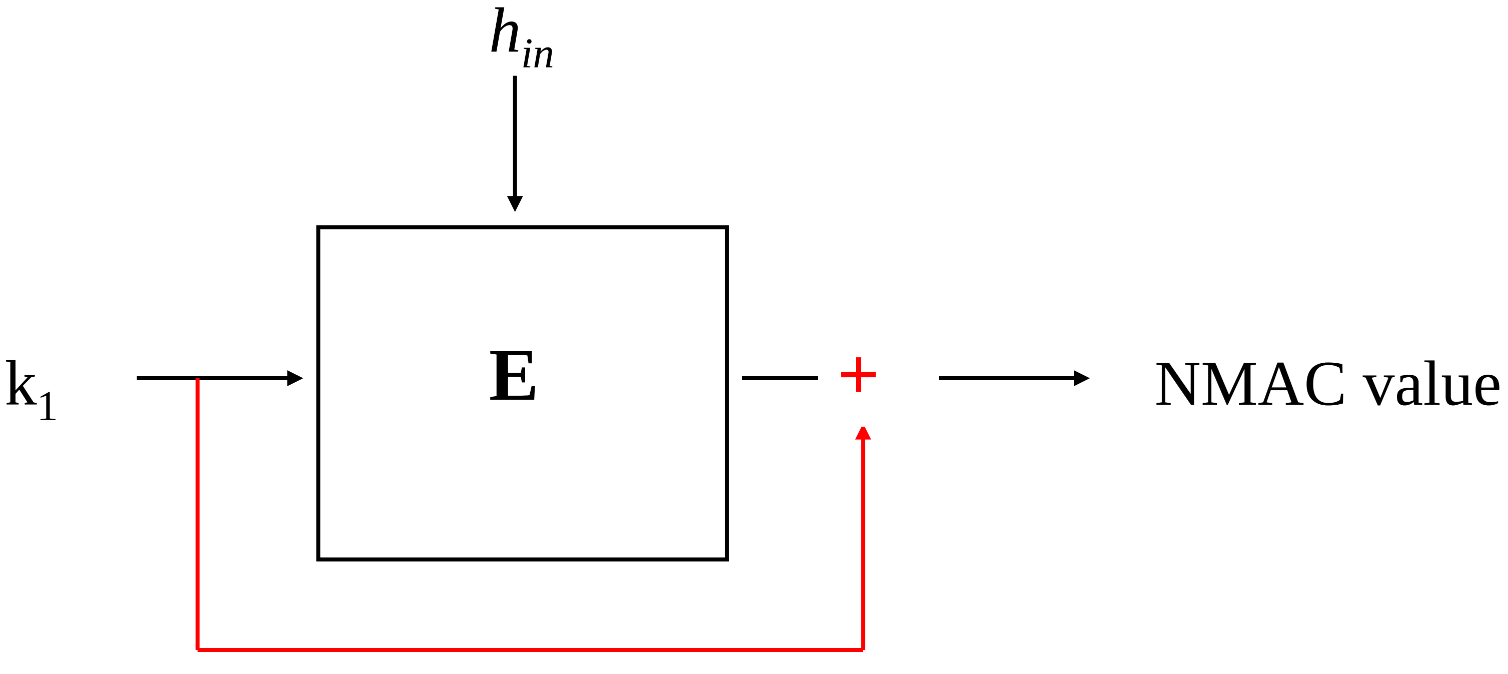
We will obtain output of E, then recover k_1.

Our outer key-recovery attacks on HMAC/NMAC-MD4

We will omit description of NMAC-MD5 case because of limited time.

Procedure of Our Attack

1. Obtain output of E in the outer MD4.



Obtain Output of E for MD4 Case

1. Determine message difference and differential path for near-collision attack:

Model of near-collision attack:

- Local collisions.
- The other differences only exist in last several steps.

Our Near-Collision on MD4

- Message differences:

$$\Delta m_3 = 2^i$$

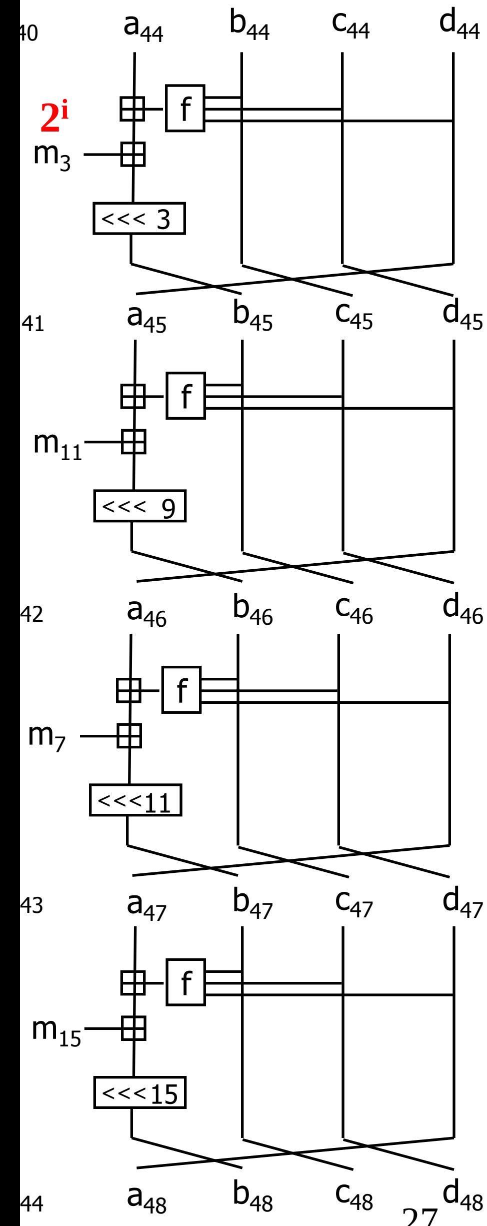
- Differential path:

The local Collision from step 1 until step 29;

The other differences only exist in the last 4 steps in third round.

3R of MD4

Local collision



Obtain Output of E for MD4 Case

1. Determine message difference and differential path for near-collision attack

2. Obtain output of E by detecting near-colliding shape.

One Weakness of Feed-Forward Operation

(k_a, k_b, k_c, k_d): 128-bit k_1 divided into four 32-bit values.

$(a_{48}, b_{48}, c_{48}, d_{48})$: output of E in the outer MD4 divied into four 32-bit values.

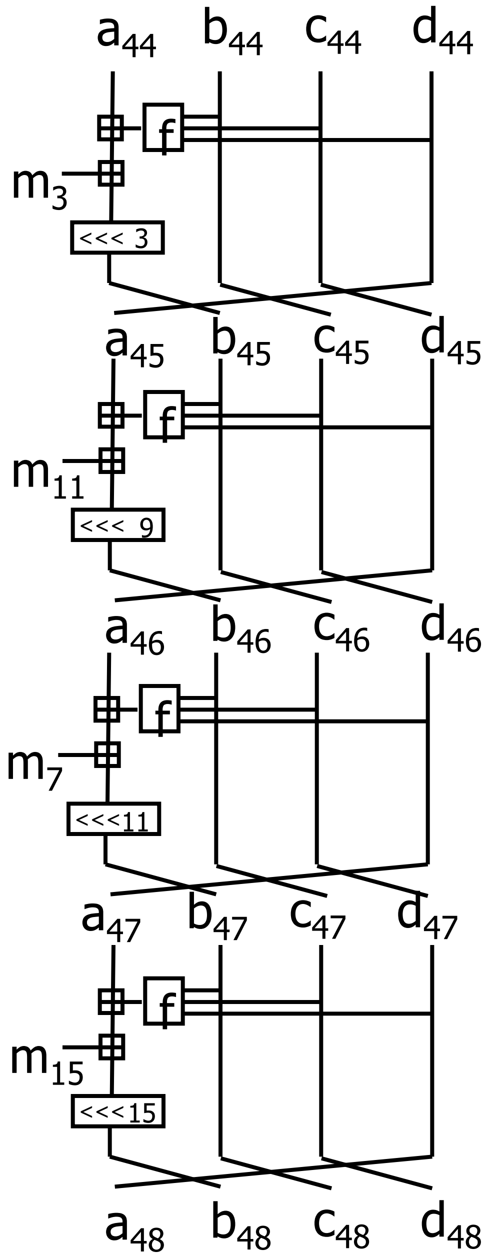
(h_a, h_b, h_c, h_d): final output of NMAC divided into four 32-bit values.

$(h_a, h_b,$

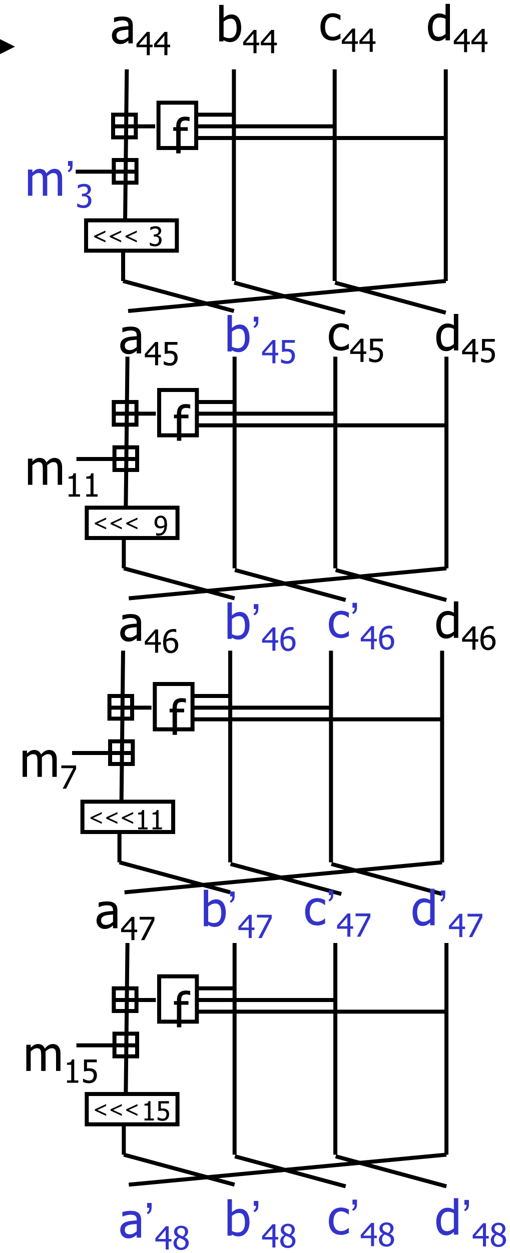
Consequently, we can obtain output of E by detecting difference propagation in last 4 steps.

$\Delta h_a = \Delta a_{48}$ $\Delta h_b = \Delta b_{48}$ $\Delta h_c = \Delta c_{48}$ $\Delta h_d = \Delta d_{48}$

One Toy Example



collision



$$m_3' - m_3 = 2^{i+3}$$

near-collision shape:

$$\Delta h_a = 2^{i+3};$$

$$\Delta h_c = 2^{i+14}+2^{i+15}+2^{i+23};$$

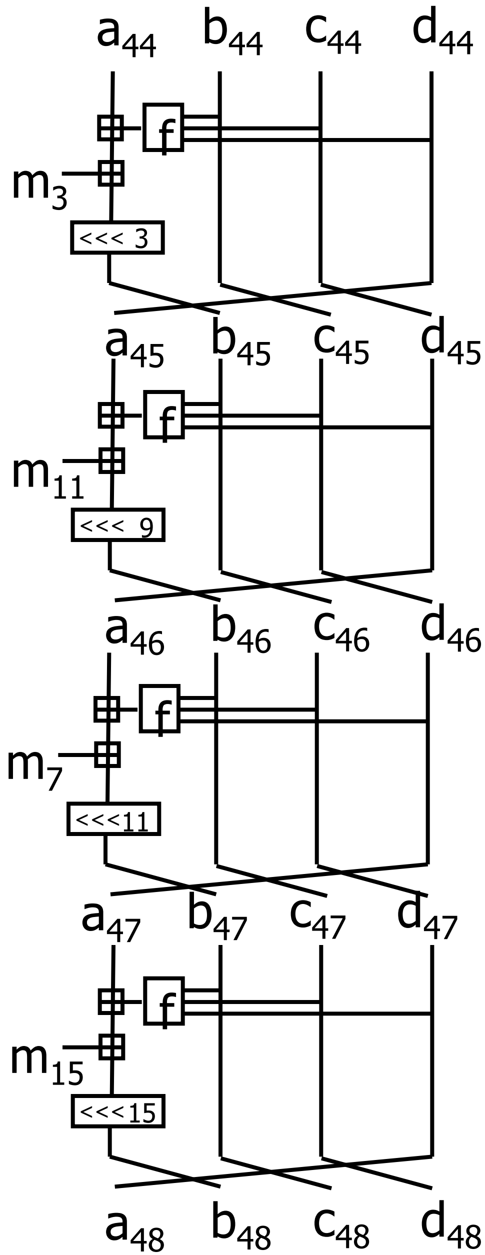
$$\Delta h_d = 2^{i+12};$$

$$\Delta a_{48} = 2^{i+3};$$

$$\Delta c_{48} = 2^{i+14}+2^{i+15}+2^{i+23};$$

$$\Delta d_{48} = 2^{i+12};$$

One Toy Example



collision

$m_3' - m_3 = 2^{i+3}$

near-collision shape:

$\Delta h_a = 2^{i+3}$;

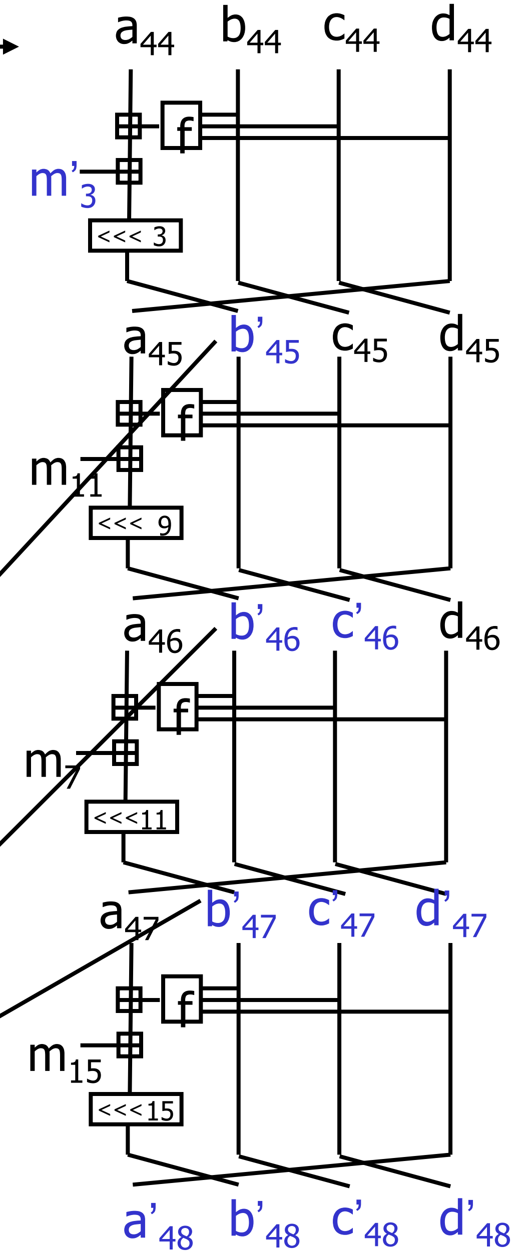
$\Delta h_c = 2^{i+14}+2^{i+15}+2^{i+23}$;

$\Delta h_d = 2^{i+12}$;

$\Delta a_{48} = 2^{i+3}$;

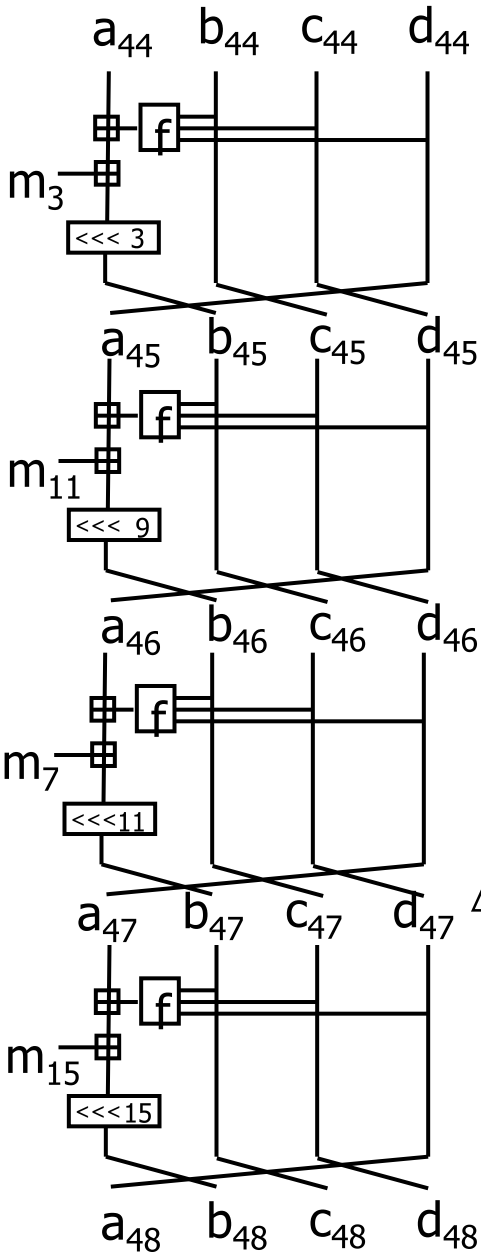
$\Delta c_{48} = 2^{i+14}+2^{i+15}+2^{i+23}$;

$\Delta d_{48} = 2^{i+12}$;



One Toy Example

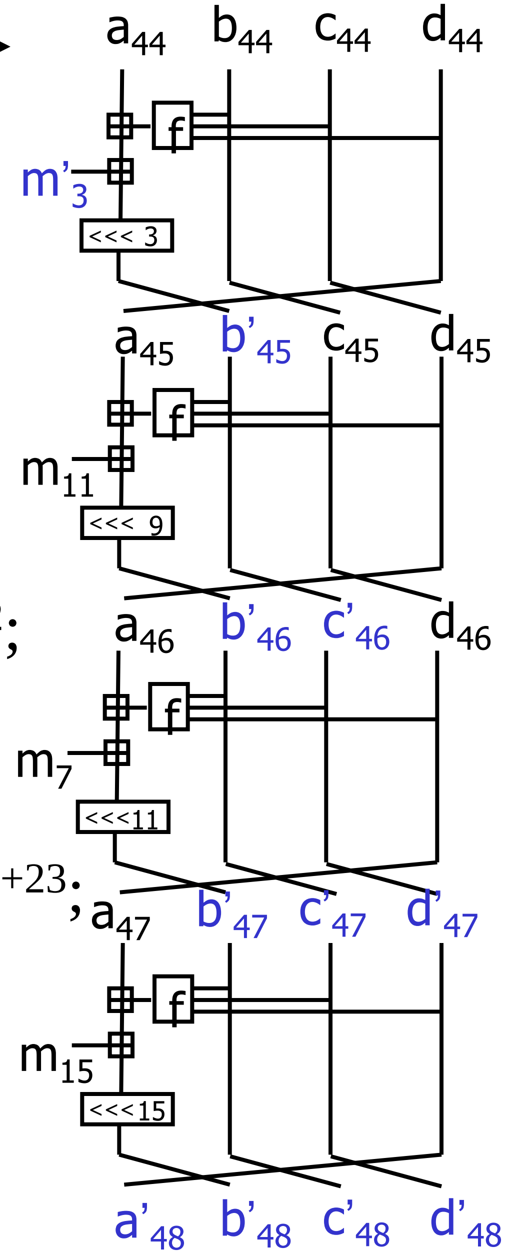
collision



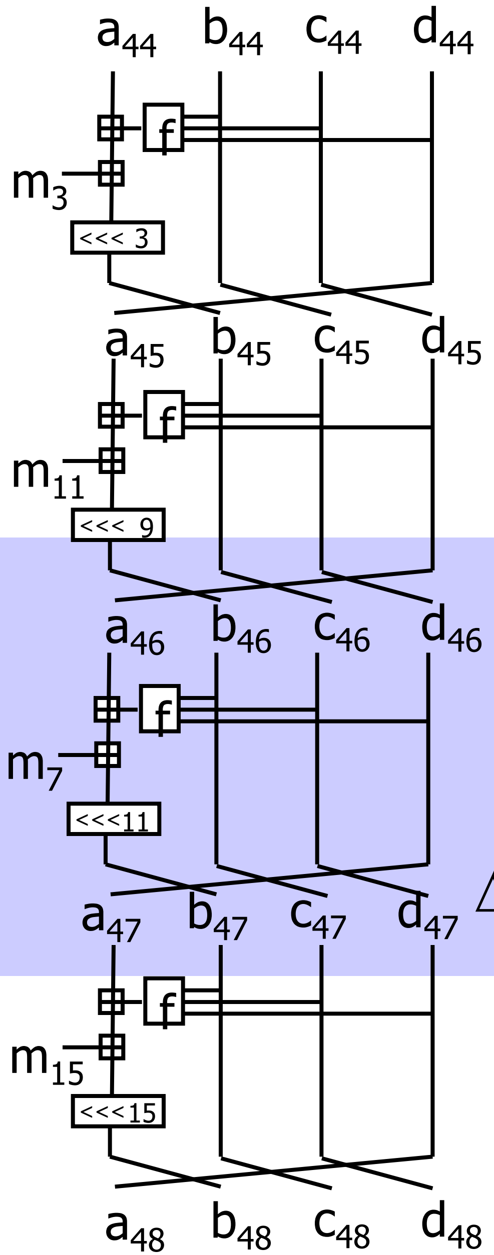
$\Delta b_{45} = \Delta a_{48} = 2^{i+3};$

$\Delta b_{46} = \Delta d_{48} = 2^{i+12};$

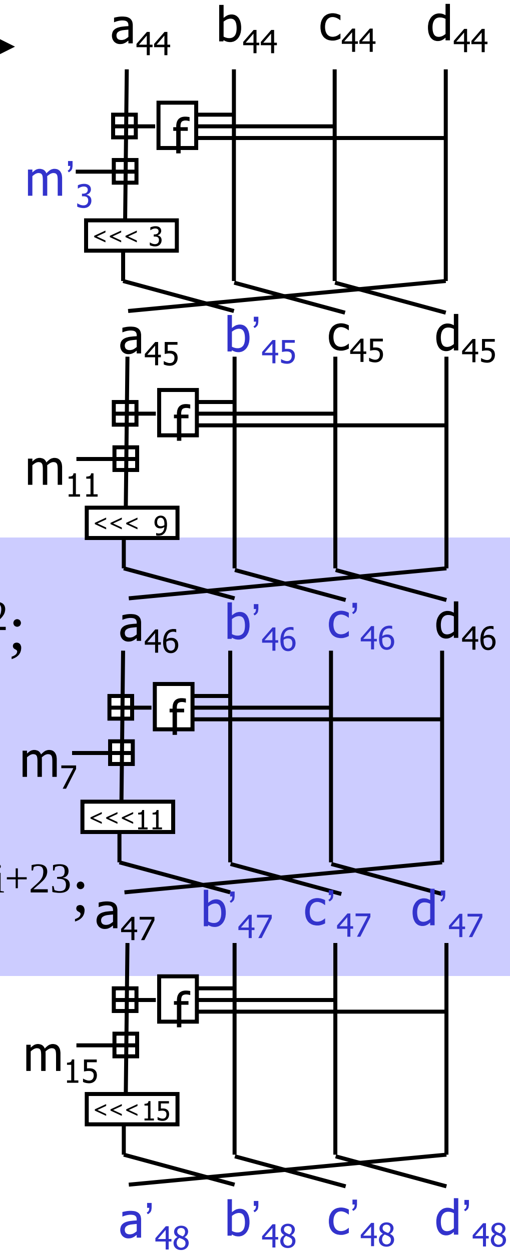
$\Delta b_{47} = \Delta c_{48} = 2^{i+14} + 2^{i+15} + 2^{i+23};$



One Toy Example



collision

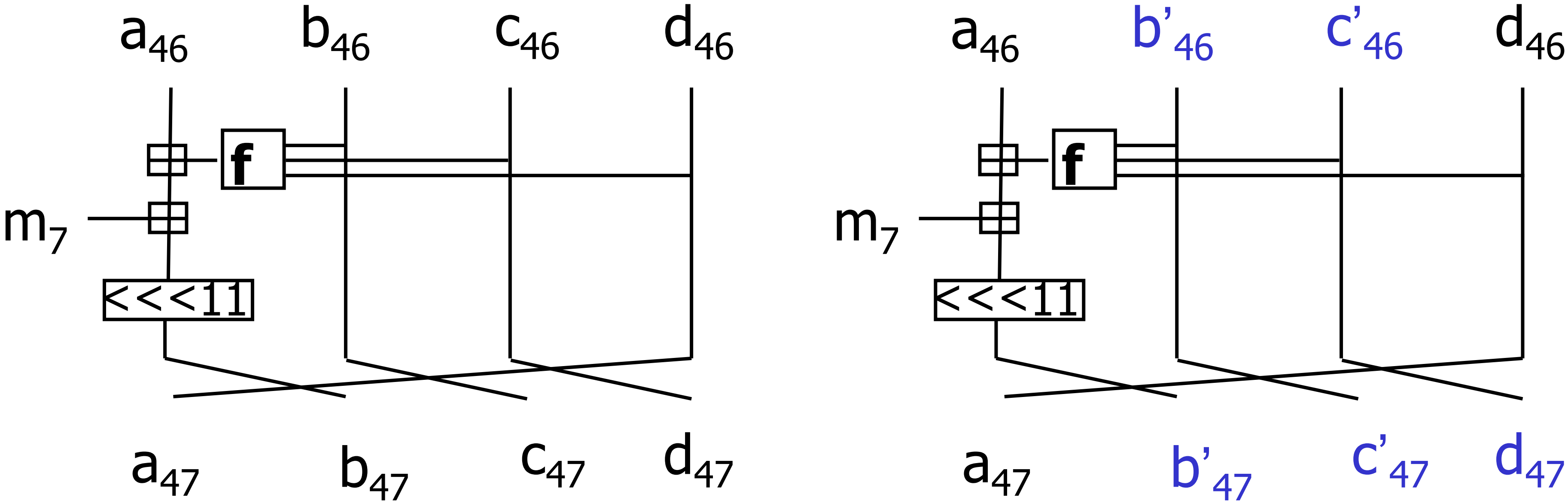


$\Delta b_{45} = \Delta a_{48} = 2^{i+3};$

$\Delta b_{46} = \Delta d_{48} = 2^{i+12};$

$\Delta b_{47} = \Delta c_{48} = 2^{i+14} + 2^{i+15} + 2^{i+23};$

One Toy Example



Δb_{47} should be caused by Δb_{46} and Δc_{46}:

$\Delta c_{46} = 2^{i+3}$;

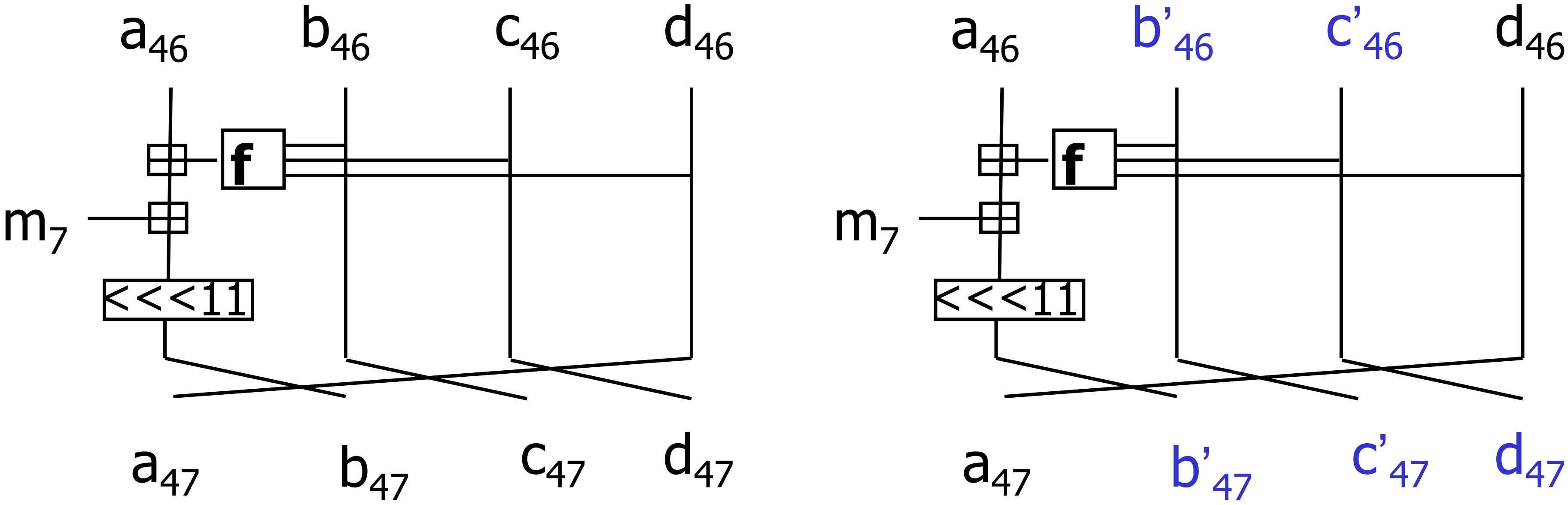
$\Delta b_{46} = 2^{i+12}$;

$\Delta b_{47} = 2^{i+14}+2^{i+15}+2^{i+23}$;

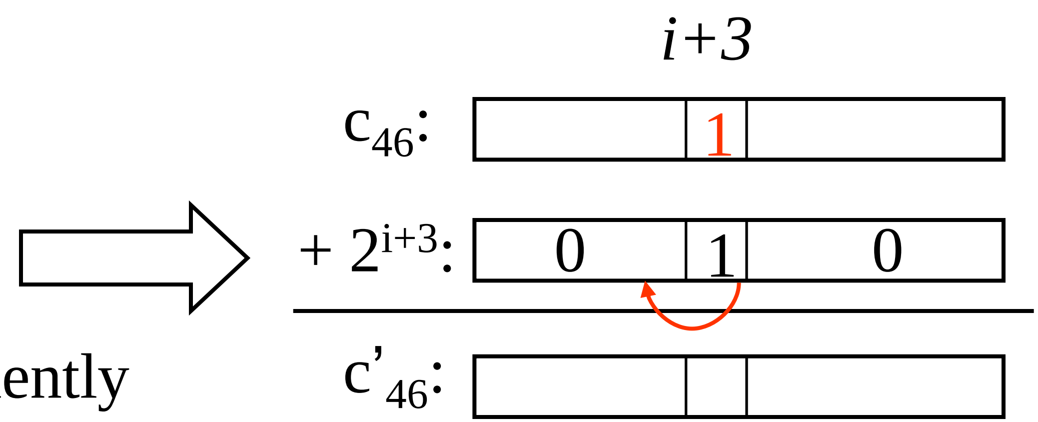
⇒ Both 2^{i+14} and 2^{i+15} are caused by 2^{i+3} of Δc_{46}.

f function works bit-independently

One Toy Example

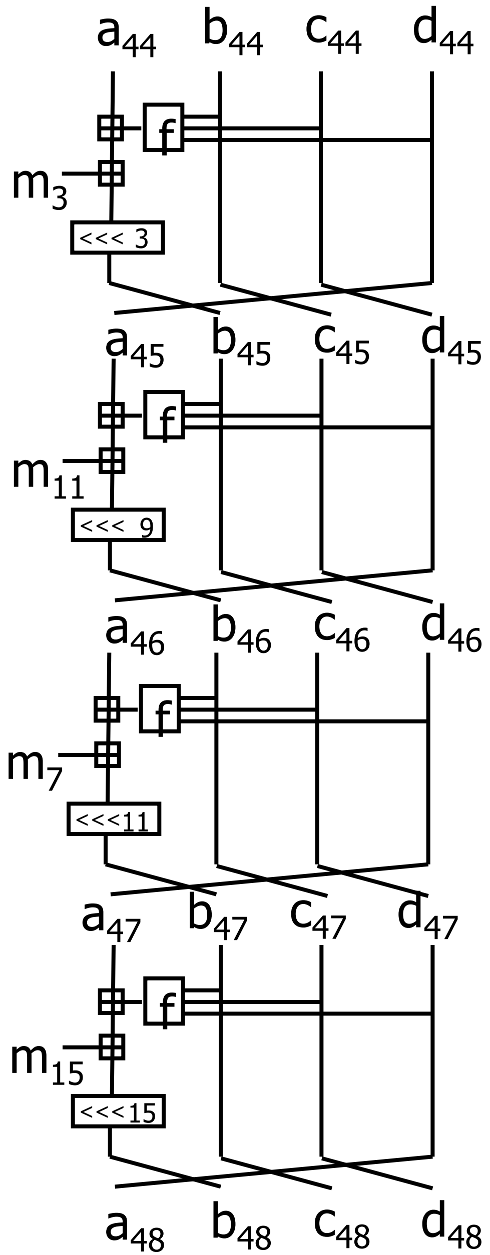


Both 2^{i+14} and 2^{i+15} are caused by 2^{i+3} of Δc_{46}.



f function works bit-independently

One Toy Example



near-collision shape:

$$\Delta h_a = 2^{i+3};$$

$$\Delta h_c = 2^{i+14}+2^{i+15}+2^{i+23};$$

$$\Delta h_d = 2^{i+12};$$

$$a_{48,i+3} = c_{46,\ i+3} = 1;$$

By similar way, we will obtain many messages such that bit-values of output of E has been recovered.

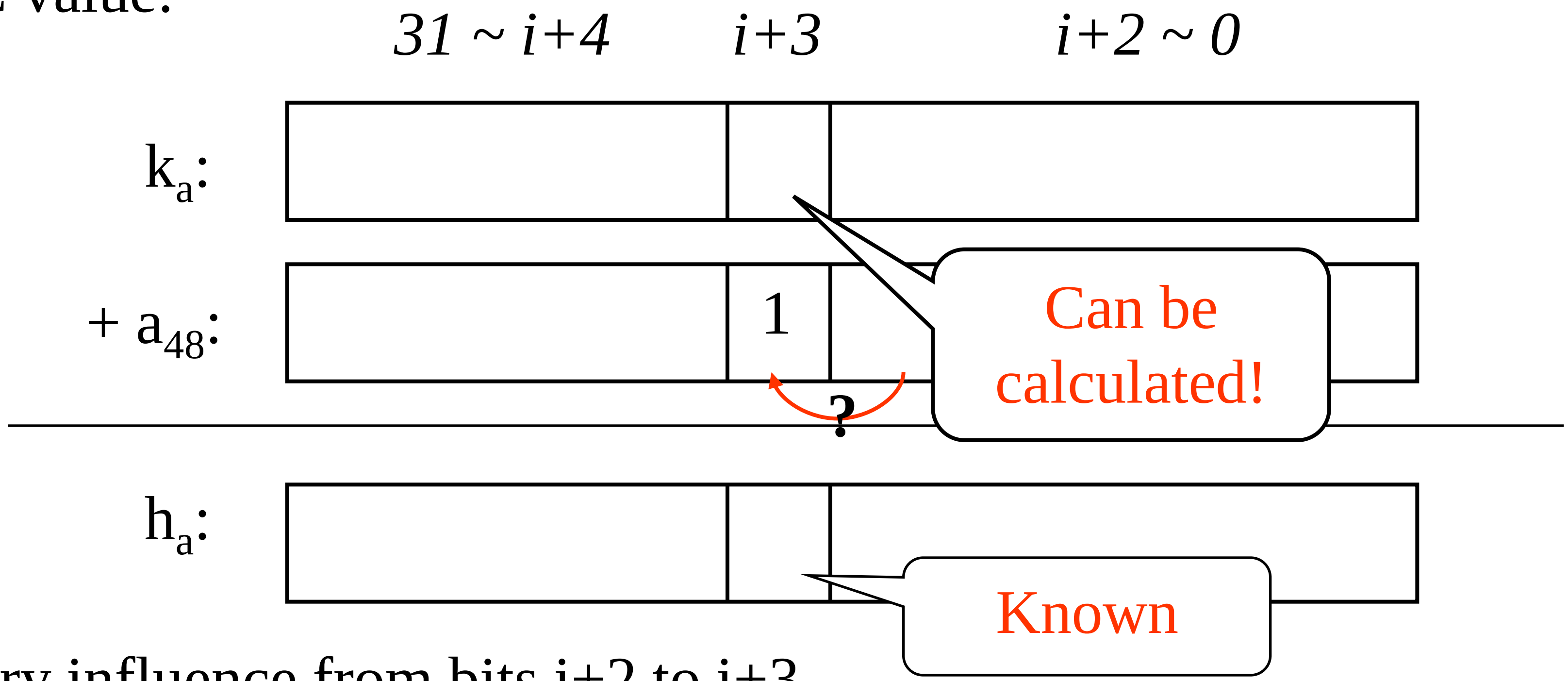
Procedure of Our Attack

1. Obtain output of E of the outer MD4.

2. Recover the outer key using output of E of the outer MD4.

The Toy Example

We obtained one message such that $a_{48, i+3} = 1$, and its corresponding MAC value:



Carry influence from bits i+2 to i+3

$h_{a,(i+2)\sim 0} \geq k_{a,(i+2)\sim 0}$: no carry during $k_{a,(i+2)\sim 0} + a_{48,(i+2)\sim 0}$

$h_{a,(i+2)\sim 0} < k_{a,(i+2)\sim 0}$: a carry during $k_{a,(i+2)\sim 0} + a_{48,(i+2)\sim 0}$

The Toy Example

We obtained one message such that $a_{48, i+3} = 1$, and its corresponding MAC value:

1. Guess the values $k_{a, (i+2)\sim 0}$.

By similar way, we can recover the outer key partially using the obtained messages.

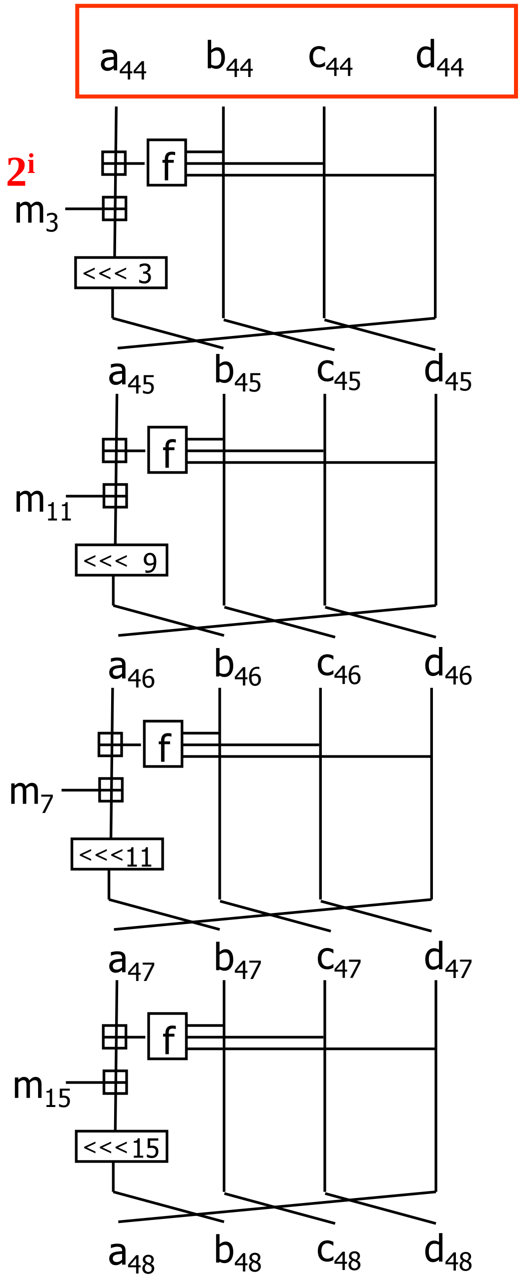
2. Compa

$h_{a,(i+2)\sim 0} \geq k_{a, (i+2)\sim 0}$: no carry during $k_{a, (i+2)\sim 0} + a_{48, (i+2)\sim 0}$

$h_{a,(i+2)\sim 0} < k_{a, (i+2)\sim 0}$: a carry during $k_{a, (i+2)\sim 0} + a_{48, (i+2)\sim 0}$

3. Calculate the bit-value of $k_{a, i+3}$.

Experiment



It is impossible to apply the real experiment because of complexity.

Instead, we did two separate experiments:

- Confirm the correctness of differential path of the local collision in first and second rounds.

- Confirm the correctness of key-recovery technique: randomly generate chaining variables in step 44.

Conclusion

We proposed new outer-key recovery attacks on HMAC/NMAC-MD4 and NMAC-MD5:

There might be two interesting points:

- New approach of key-recovery attack: using feed-forward operation of MD4 and MD5.

- One near-collision model: local collisions + the other difference propagation in last several steps.

Complexity Comparison

Comparison		Fouque et al.'s work	Our results
HMAC/NMAC-MD4 **Standard Attack**	Online complexity	2^{88}	2^{72}
	Recovered bit-values (online)	22	51
	Offline complexity	2^{95}	2^{77}
	Total complexity	2^{95}	2^{77}
NMAC-MD5 **Related-Key Attack**	Online compexity	2^{51}	2^{75}
	Recovered bit-values (online)	28	53
	Offline complexity	2^{100}	2^{75}
	Total complexity	2^{100}	2^{76}

Thank you & Question