Cryptanalysis on a Merkle-Damgård Based MAC

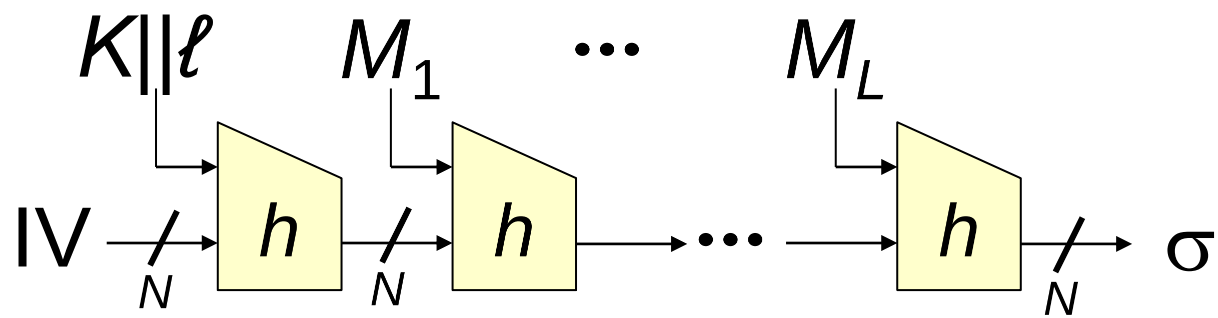
Almost Universal Forgery and Distinguishing-*H* Attacks

Yu Sasaki (NTT Corporation)

17.04.2012 @ Eurocrypt 2012

Research Summary

- Present two generic attacks against:
  LPMAC construction + narrow-pipe MD hash



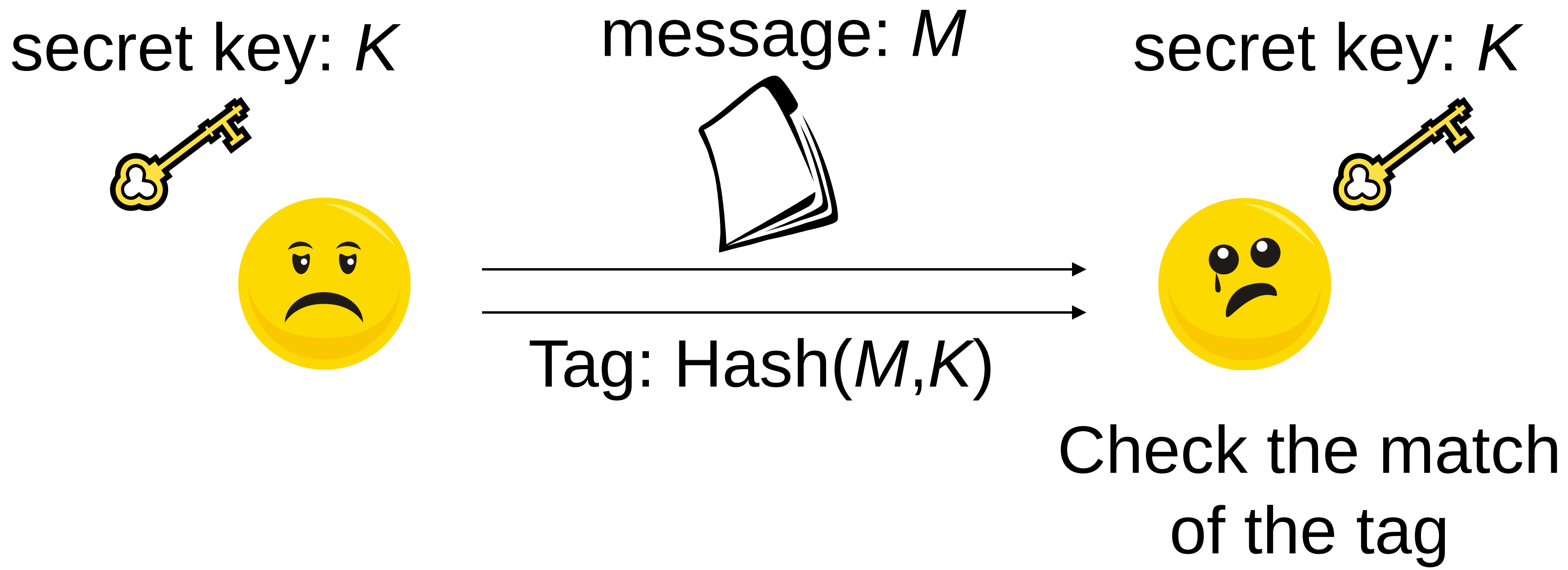
Generic Distinguishing-H	Queries	Time	Mem.
Previous	2^N	-	-
This paper	$\mathbf{3 \times 2^{N/2}}$	$\mathbf{2^{N/2}}$	**-**

Contents

- Background
- Generic distinguishing-*H* attack on LPMAC
- Almost universal forgery attack on LPMAC
- Conclusion / future work

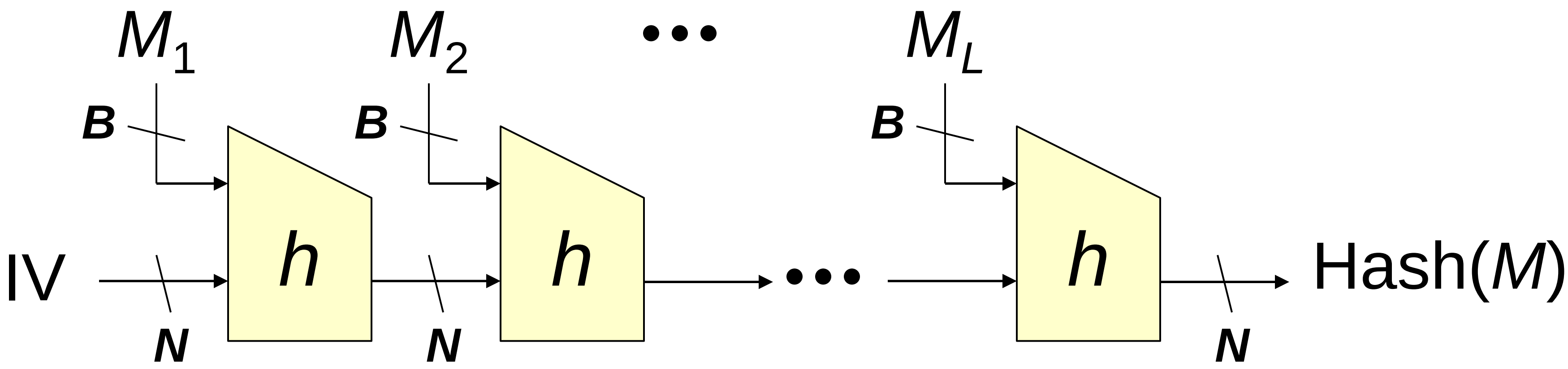
Hash Function Based MAC

- Message Authentication Codes (MAC) provides the integrity and authenticity.



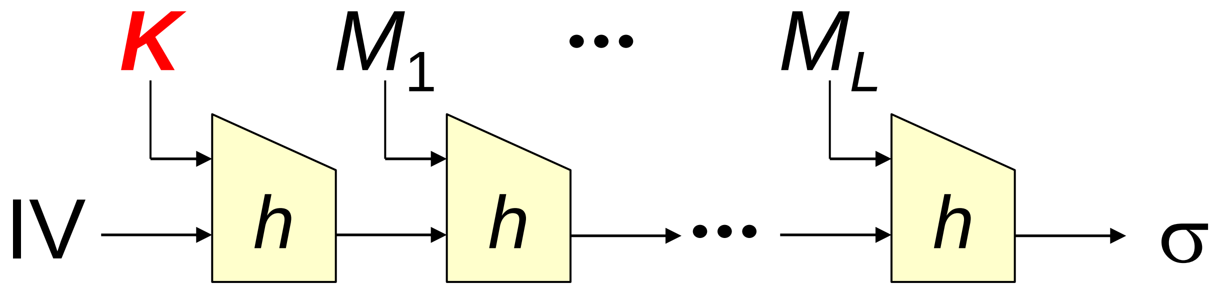
Hash Function Structure

- Merkle-Damgård Domain Extension:
  - Iteratively apply the fixed size compression function
- Narrow-pipe
  - Internal state size and the hash value size are identical

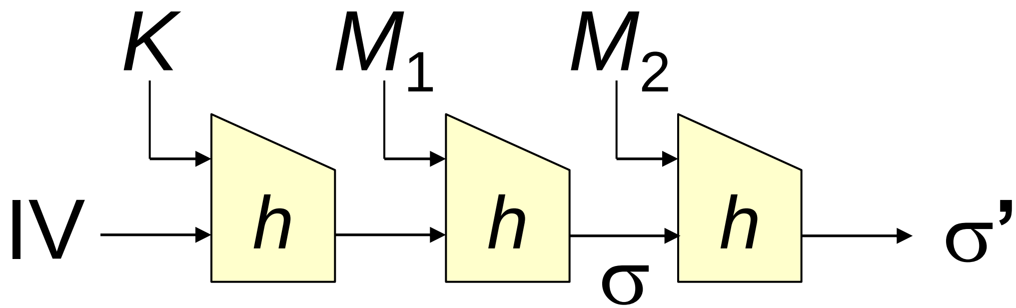


Classic MAC Construction and its Weakness

- Secret-prefix MAC



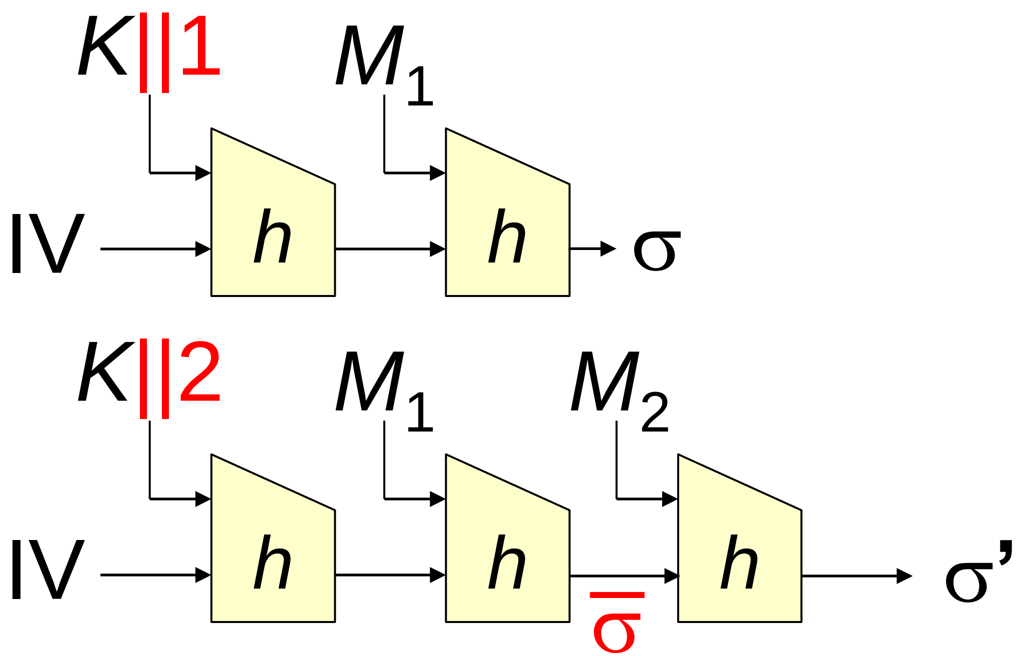
- Forgery attack with complexity 1
  - Query a tag for M_1, then a tag for $M_1||M_2$ can be computed at offline.



Strengthening Secret-Prefix MAC

- Append the length of the input message before the message is computed.

(Length-Prepended MAC ➡ LPMAC)



Security Proof for LPMAC

- LPMAC satisfies the *prefix-freeness*:

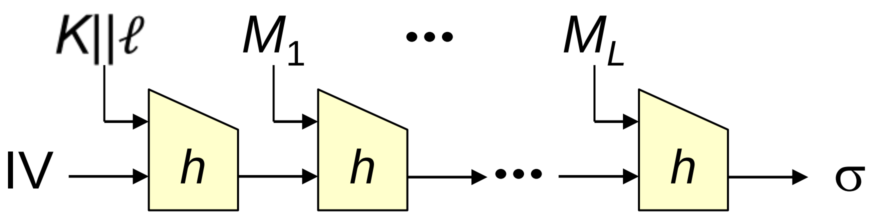
  Any message is not a prefix of other messages

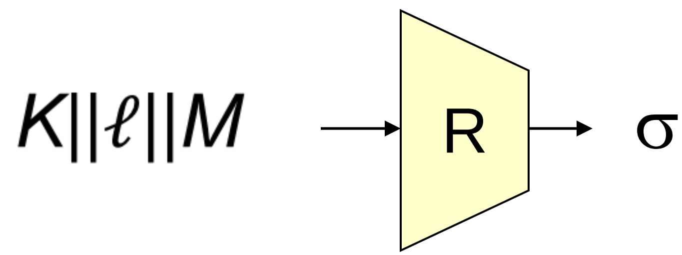
- Prefix-free MAC was proved to be a secure PRF up to $2^{N/2}$ queries [BCK96].

Security Evaluation for MAC Constructions

- Distinguishing-R (Generic attack with $2^{N/2}$)

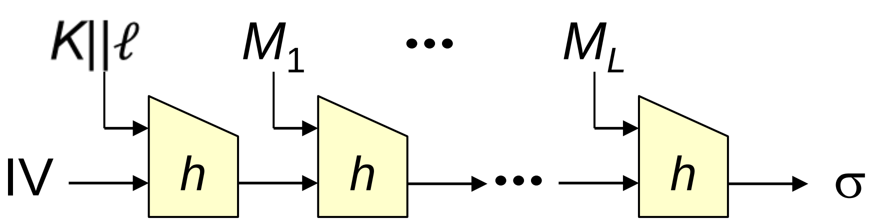
[PO95]

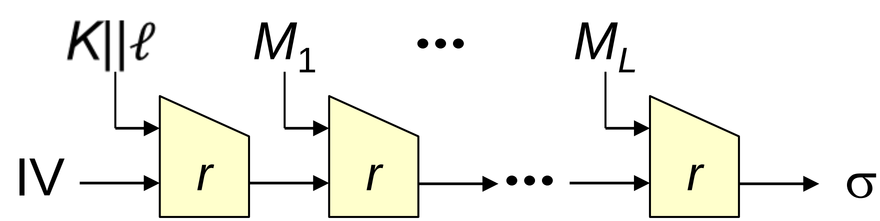




- Distinguishing-H (Generic attack with 2^{N} ?)

[folklore]





Previous Approach of Dist-H

- Tried to find a distinguisher which is faster than 2^N complexity.

- Combination of the generic birthday attack and dedicated differential cryptanalysis.

  – Due to the birthday attack, #queries is bigger than $2^{N/2}$.

  – Due to the differential cryptanalysis, attacking full rounds is hard.

NTT

Previous Distinguishing-*H* Attacks on LPMAC

Hash	Size(N)	Rounds	Queries	Reference
SHA-1	160	43/80	$2^{124.5}$	[WWJW09]
SHA-1	160	61/80	$2^{154.5}$	[WWJW09]
SHA-1	160	65/80	$2^{80.9}$	[QWJ09]
SHA-256	256	39/64	$2^{184.5}$	[YW09]
RIPEMD	128	48/48 Full	2^{66}	[W10]
RIPEMD256	256	58/64	$2^{163.5}$	[W10]
RIPEMD320	320	48/80	$2^{208.5}$	[W10]

All attacks require more than $2^{N/2}$ queries.

Our Results

- A generic distinguishing-*H* attack against LPMAC with a narrow-pipe MD hash.

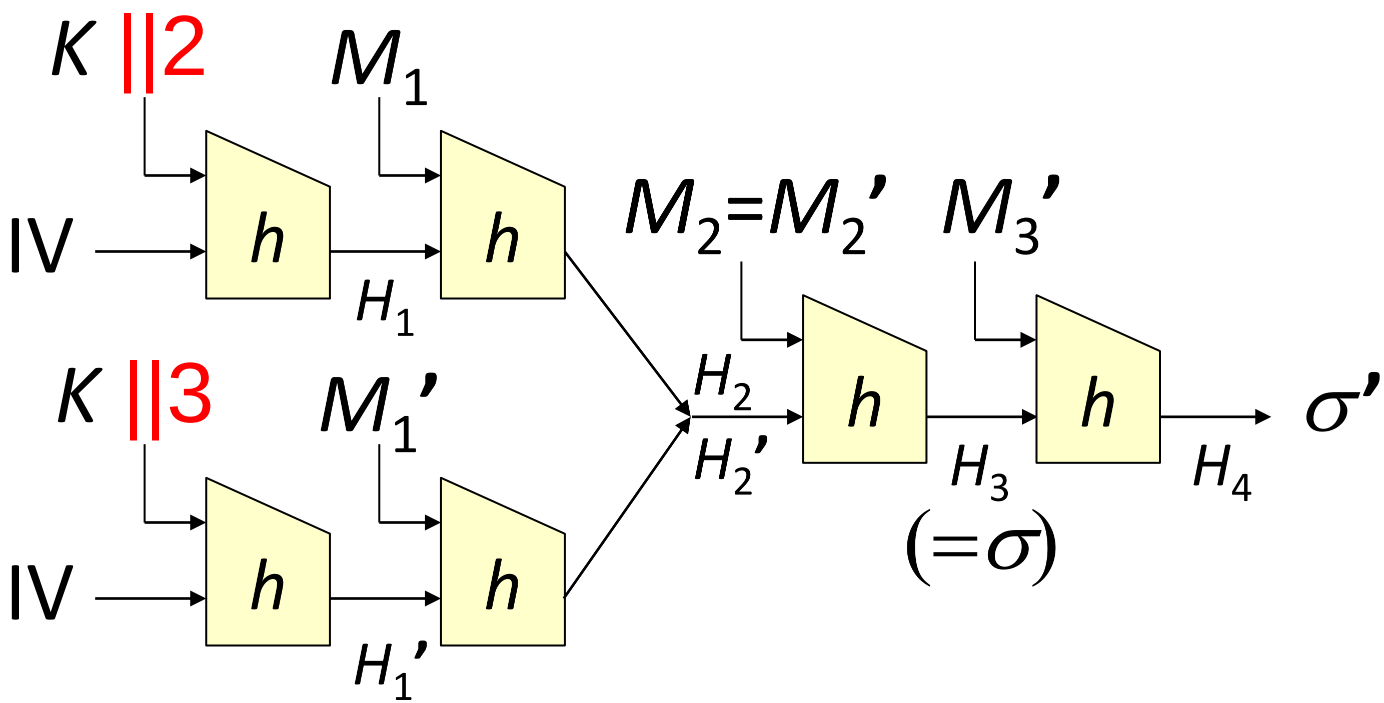
Hash	Size	Rounds	Queries
Generic narrow-pipe MD	N	Full	$3 \times 2^{N/2}$

The folklore was incorrect.

New Distinguishing-*H* attack on LPMAC

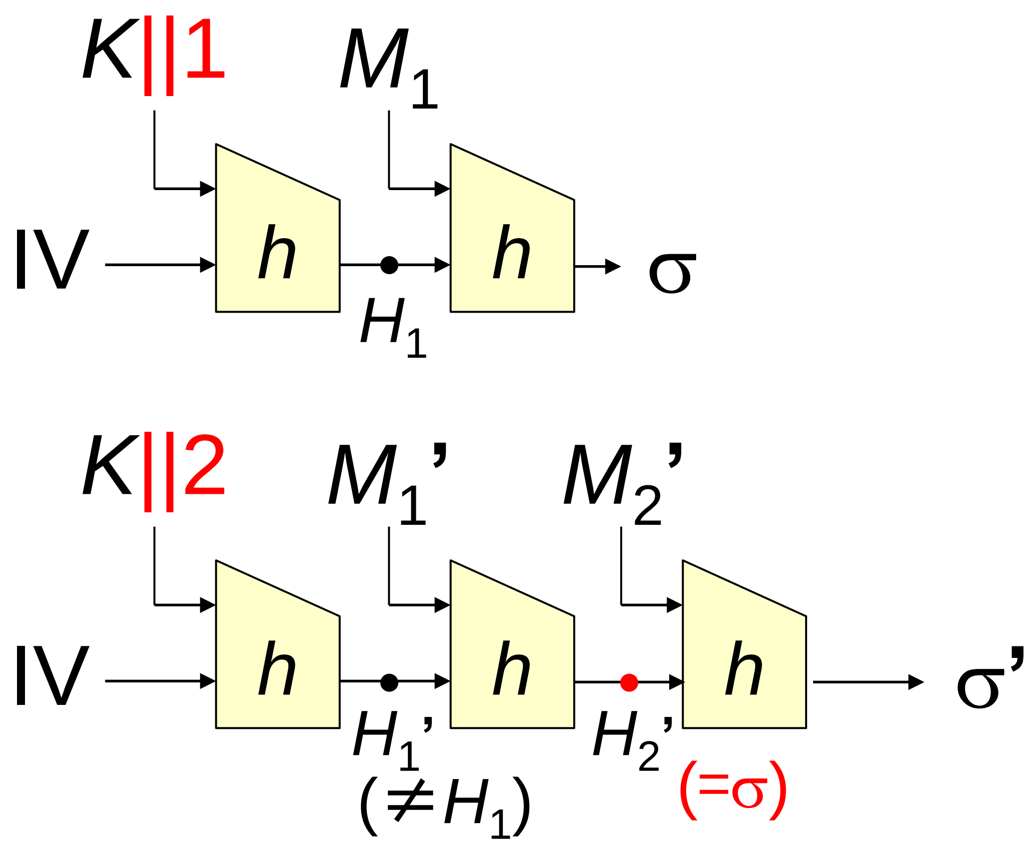
Basic Idea

- Assume that an internal collision starting from different length-prepend strings are generated.
- Querying $M_1||M_2$ reveals H_3.
- Querying $M_1'||M_2'||M_3'$ reveals H_4.
- All information for the last block is obtained.



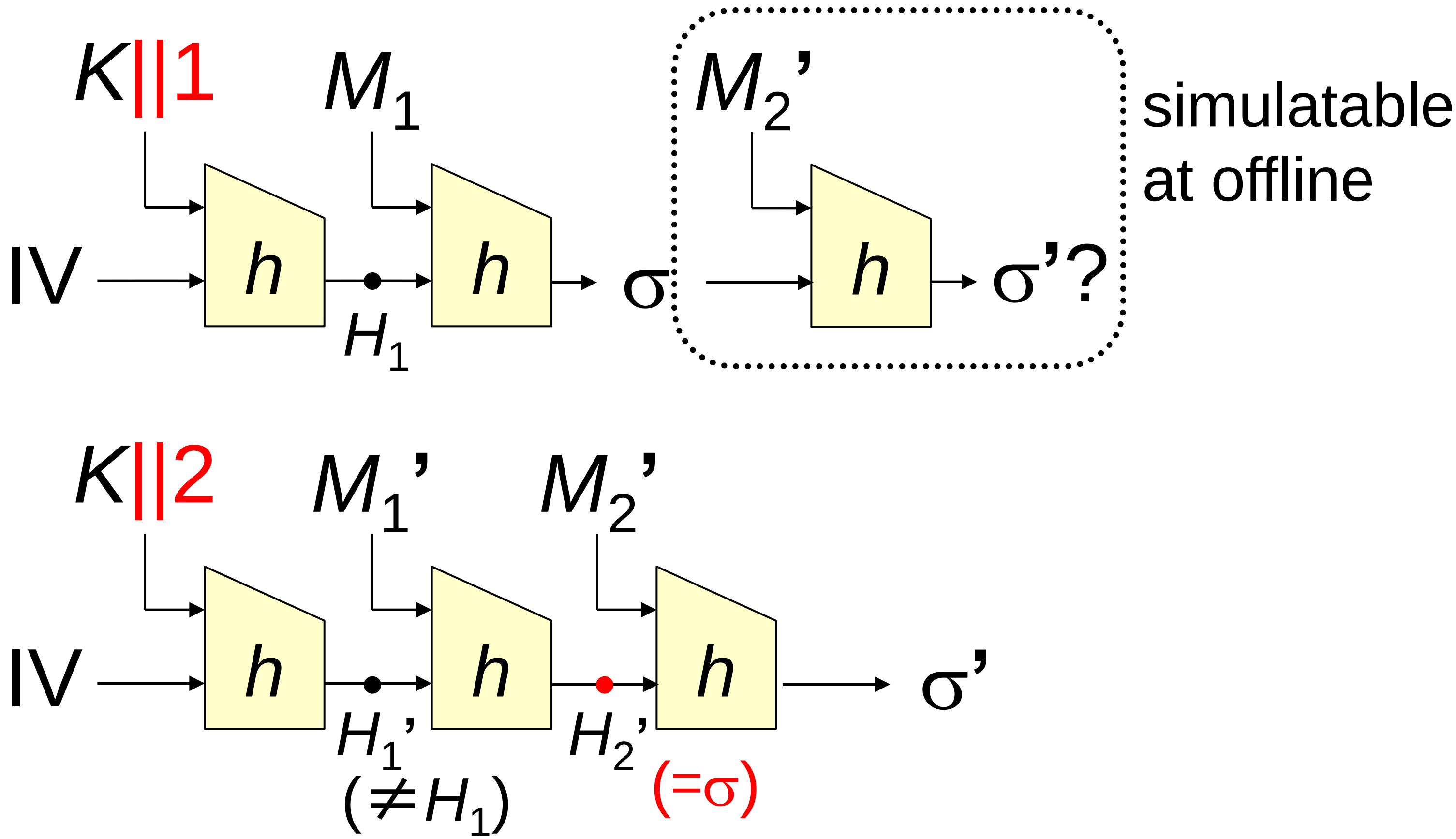
Basic Idea

- How to detect the internal collision only with queries of different lengths?



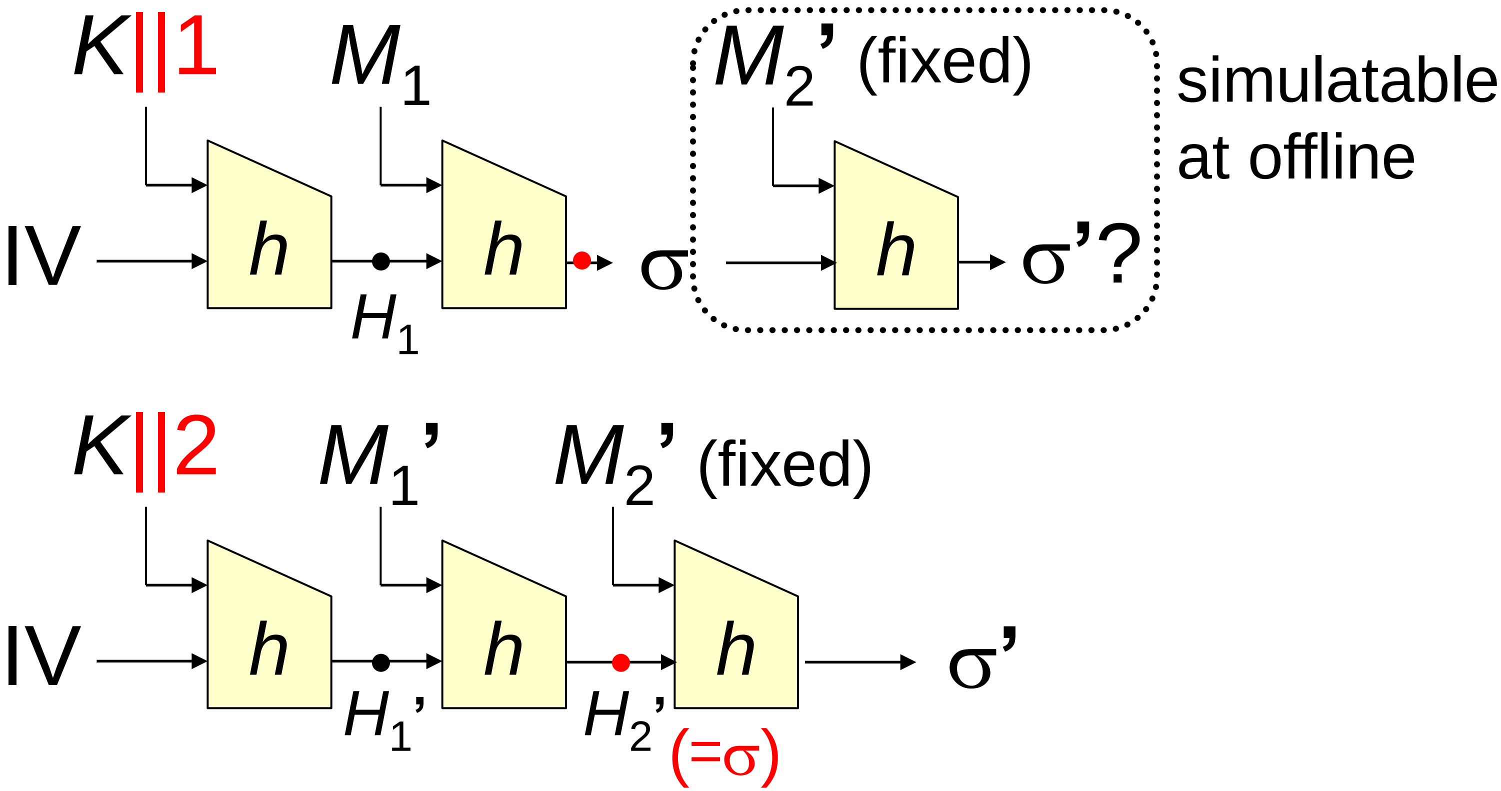
Basic Idea

- How to detect the internal collision only with queries of different lengths?



Attack Procedure

1. Fix M_2' to a randomly chosen value.
2. Query $2^{N/2}$ M_1 and get σ. Compute $h(\sigma, M_2')$ and store them.
3. Query $2^{N/2}$ $M_1'||M_2'$ and get σ'. Check the match with Step 2.
4. For the matched (M_1, M_1'), check the match with different M_2'.



Evaluation of the Attack

1. Fix M_2' to a randomly chosen value.
2. Query $2^{N/2}$ M_1 and get σ. Compute $h(\sigma, M_2')$ and store them.
3. Query $2^{N/2}$ $M_1'||M_2'$ and get σ'. Check the match with Step 2.
4. For the matched (M_1, M_1'), check the match with different M_2'.

If Step 4 succeeds, h is the target hash function.

Step 1: Negligible

Step 2: Query=$2^{N/2}$, Time=$2^{N/2}$, Mem.=$2^{N/2}$

Step 3: Query=$2 \times 2^{N/2}$

Step 4: Negligible (Query=4, Time=2, Mem.=1)

Total cost: Query=$3 \times 2^{N/2}$, Time=$2^{N/2}$, Mem.=$2^{N/2}$

(can be memoryless with the memoryless MitM)

More Cryptanalysis on LPMAC

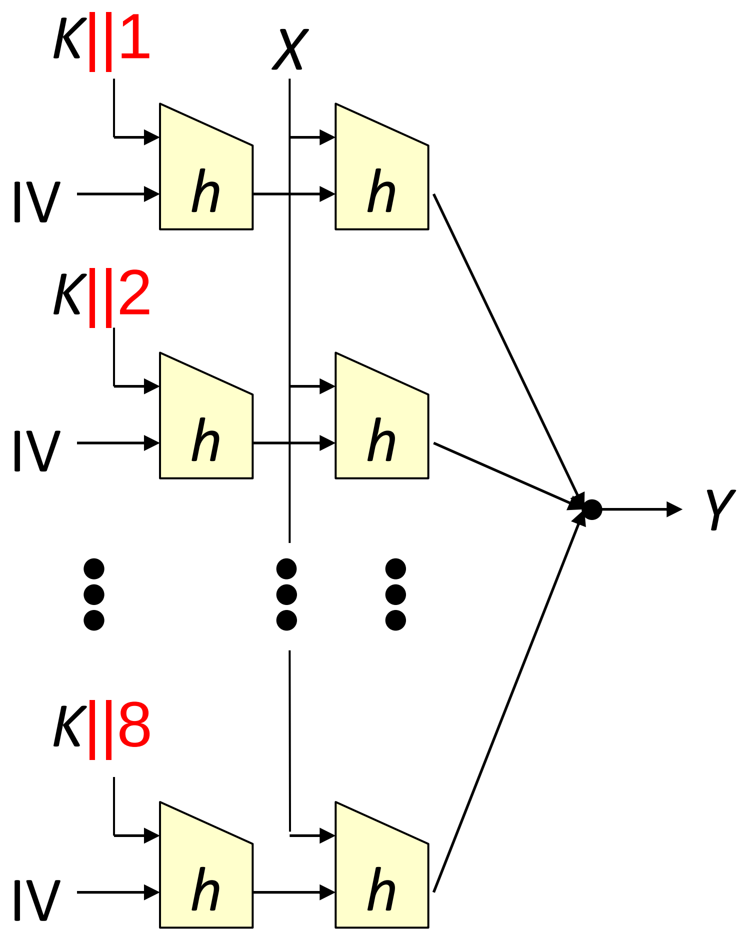
Almost Universal Forgery Attack

Almost Universal Forgery (AUF)

- Introduced by [DKS11]
  1. Do some pre-computation (and pre-query).
  2. A target message is randomly given.
  3. Attacker modifies 1-block of the given message.
  4. Perform the forgery on the modified target.
- In our attack, the first $log_2 L$ blocks are replaced (L is a size of the message).
- For LPMAC, precomputation must be done without knowing the target message length.

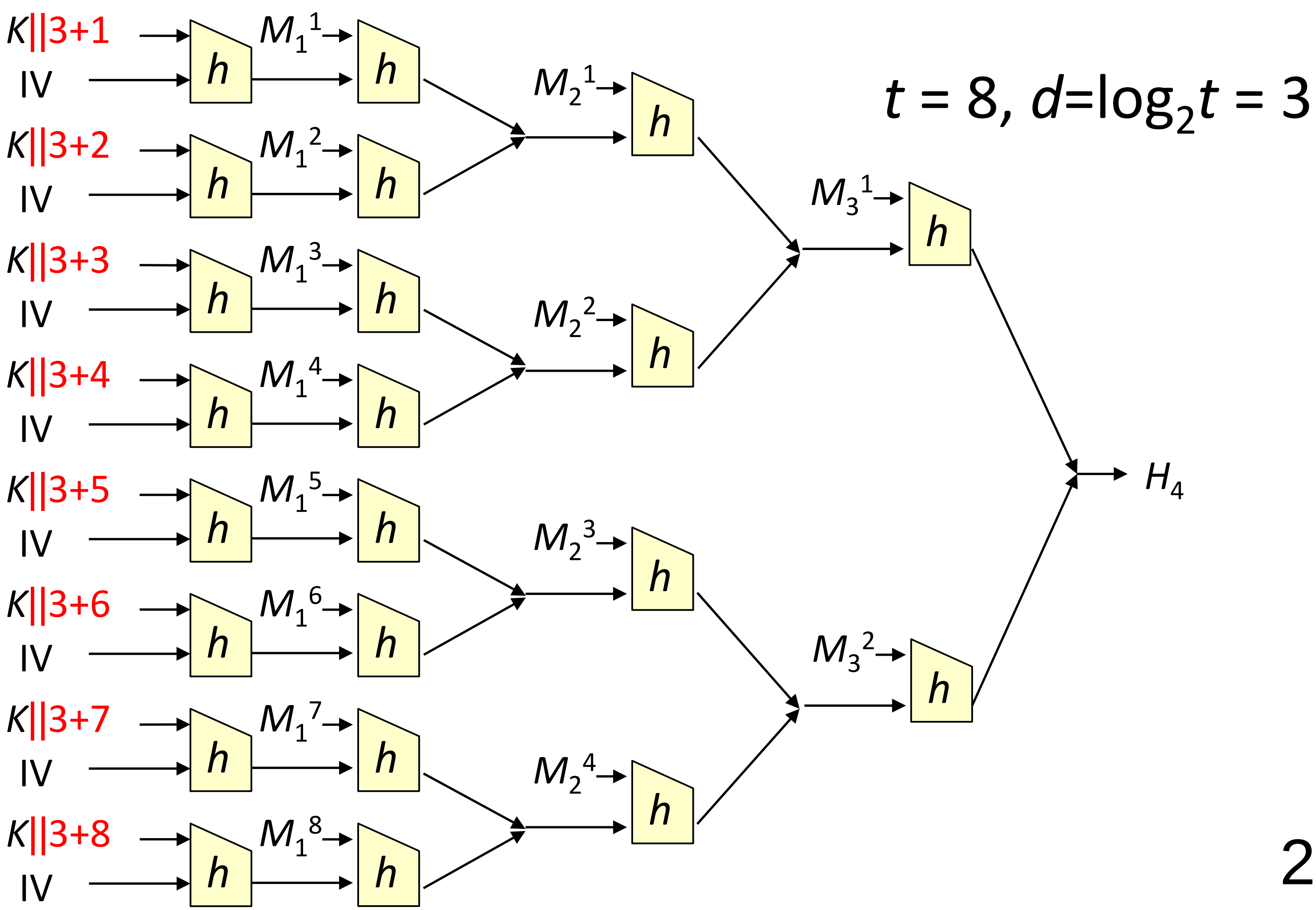
Basic Idea

- Use a multi-collision starting from various length-prepend values.



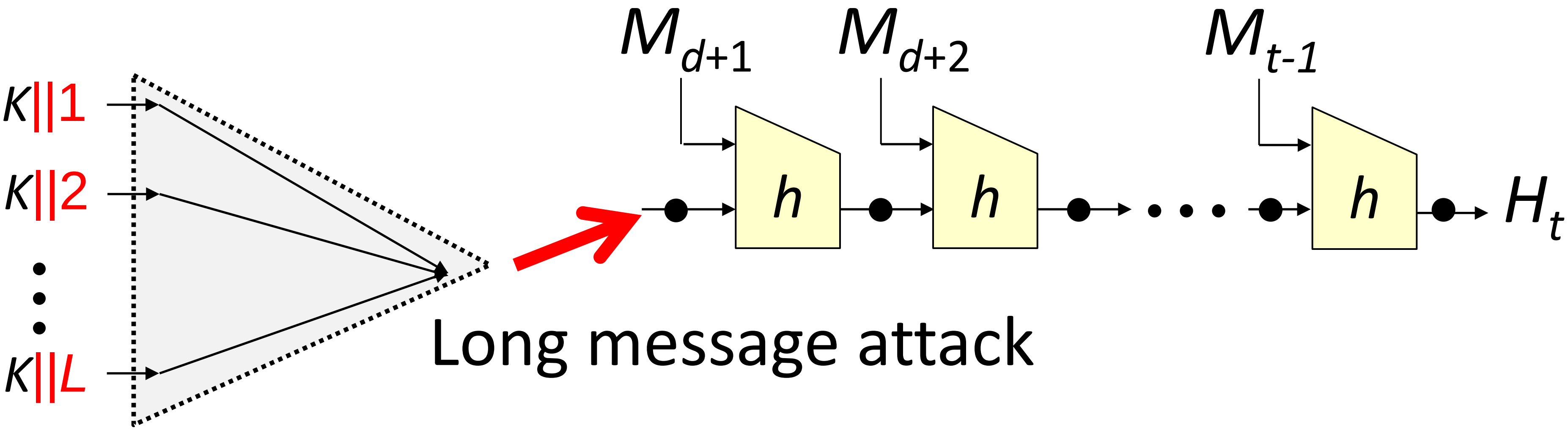
Basic Idea

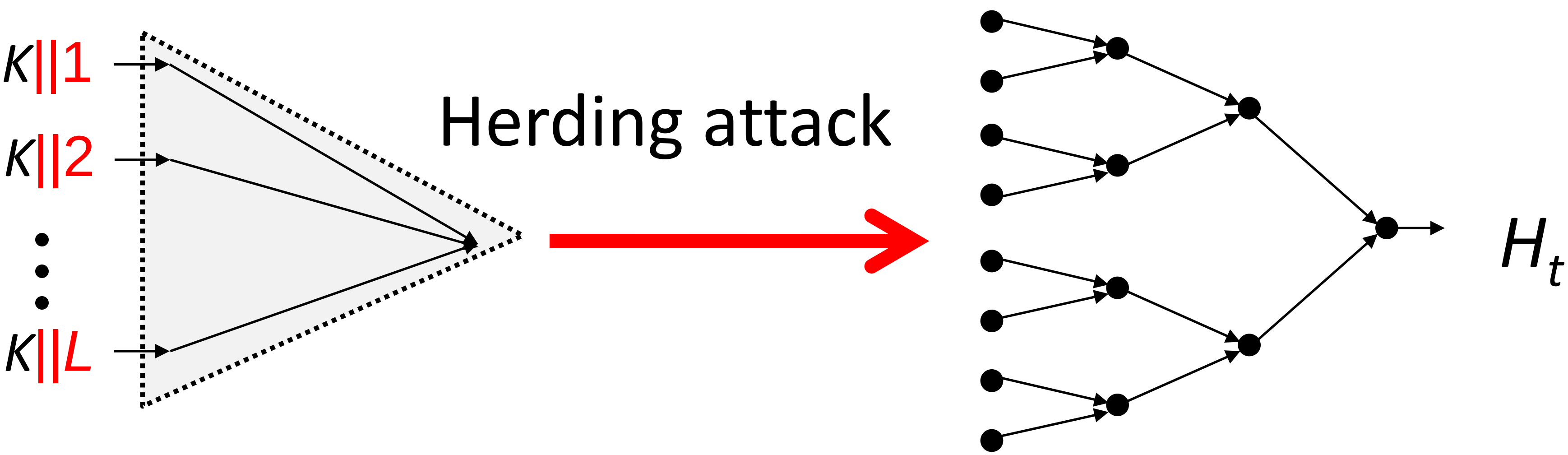
- 1-block multi-collision is inefficient.
  → Use the diamond structure.



Potential Applications

- Find a message connecting a given internal state to multiple targets with various message length.





Conclusion / Future Work

Concluding Remarks

- Proposed a generic distinguishing-*H* attack on LPMAC.

	Queries	Time	Mem.
Our generic Dist-*H*	$3 \times 2^{N/2}$	$2^{N/2}$	-

- The “N-bit security folklore” is incorrect.

- Showed more cryptanalysis on LPMAC. Prefix-freeness is broken with $2^{N/2}$ queries.

Future Research Directions

- Finding a new problem on MAC in which a generic attack costs between $2^{N/2}$ and 2^{N}

- Finding a new application of a differential with Pr. > 2^{-N}

A differential with Pr. > 2^{-N} $\not\Rightarrow$ Dist-*H* on LPMAC

A differential with Pr. > 2^{-N} $\Rightarrow$???

Thank you for your attention !!