RSA Conference 2019
San Francisco | March 4 – 8 | Moscone Center

BETTER.

SESSION ID: CRYP-R02

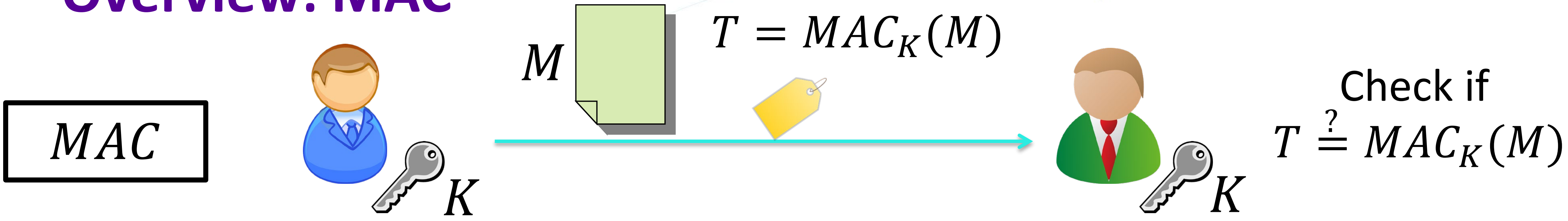
Universal Forgery and Multiple Forgeries of MergeMAC and Generalized Constructions

Tetsu Iwata[1], Virginie Lallemand[2], Gregor Leander[2], and **Yu Sasaki**[3]

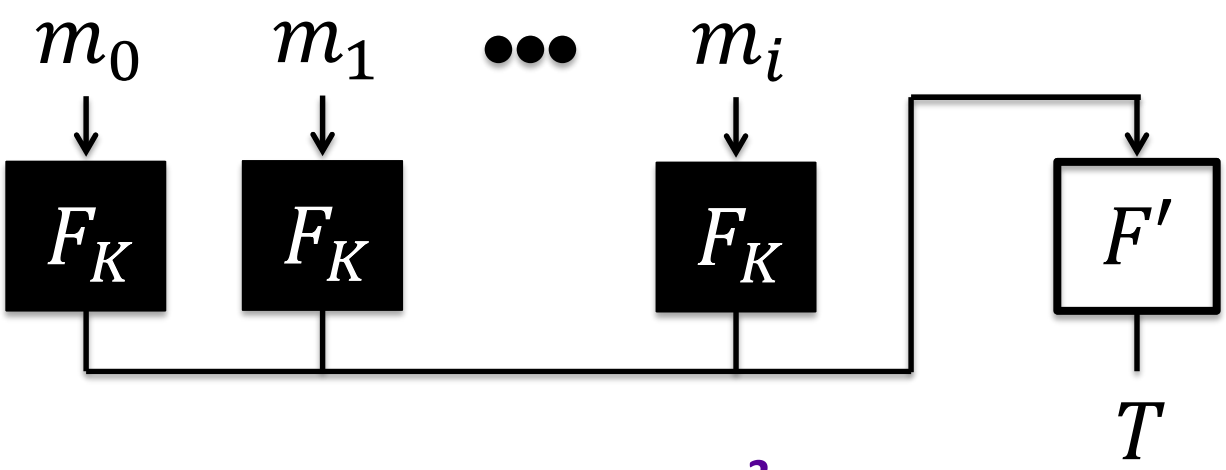
1: Nagoya University
2: Ruhr-Universität Bochum
3: NTT Secure Platform Laboratories

#RSAC

Overview: MAC



- MergeMAC: Lightweight MAC for IoT from ACNS2018 (June 2018)
- Unique Feature:



- Classic:
  - Finalization is keyed and strong.
- MergeMAC:
  - Finalization is public and weak.

Overview: Our Results

- MergeMAC was designed to provide 64-bit security
- We found universal forgery attacks with
  - 2^{32} data and 2^{32} offline comp for any (even keyed) finalization.
- IoT devices may not communicate 2^{32} data in lifetime. We can still apply universal forgery attacks with
  - 2^{8} data and $2^{58.6}$ offline comp by using MergeMAC's weak finalization.
  - 2^{24} data and 2^{48} offline comp even with secure hash function.
- Multiple Forgery:
  - The average attack cost becomes cheaper when we forge many tags.
  - Optimality of our attacks is proven in some particular setting.

RSAConference2019

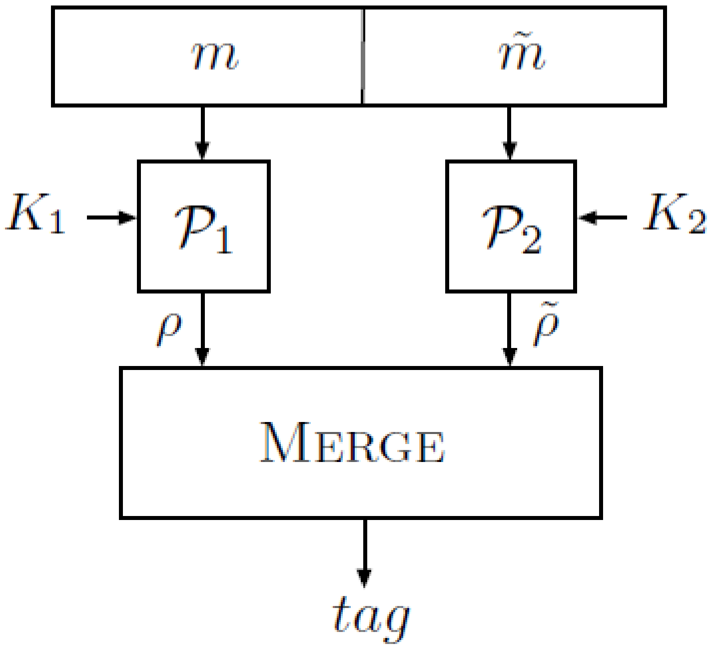
More Backgrounds of MergeMAC

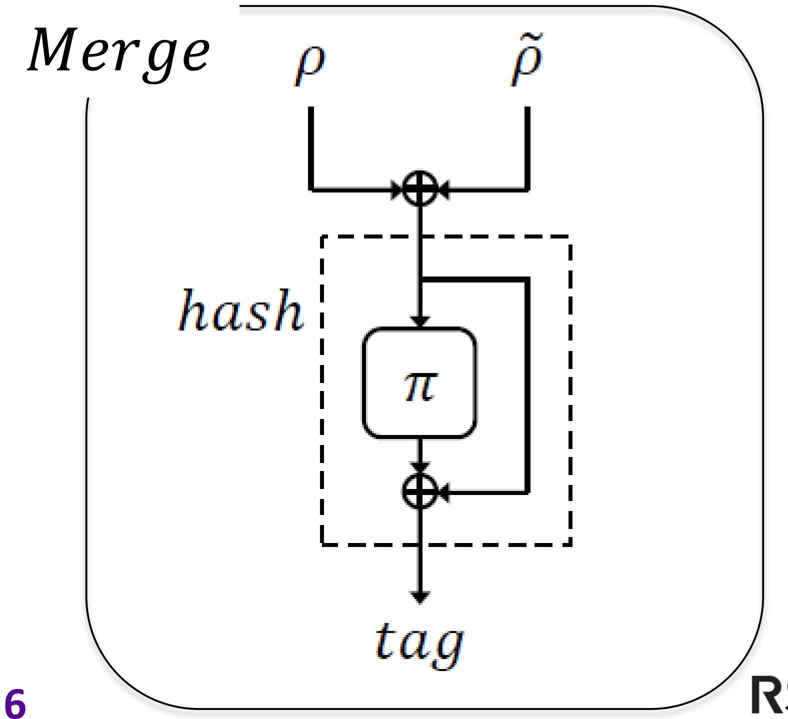
Overview: Our Results

- MergeMAC [Ankele et al. ACNS2018] is a MAC suitable when
  - bandwidth is limited
  - strict time constraints apply
- Assumed usage: CAN bus, a communication system in modern cars.
- The important feature is low latency. How to achieve it?
- Save bandwidth by not transmitting low-entropy bits of the msg.
- This allows speed up by storing frequently needed intermediate parts in the cache instead of computing them again.

MergeMAC Specification 1

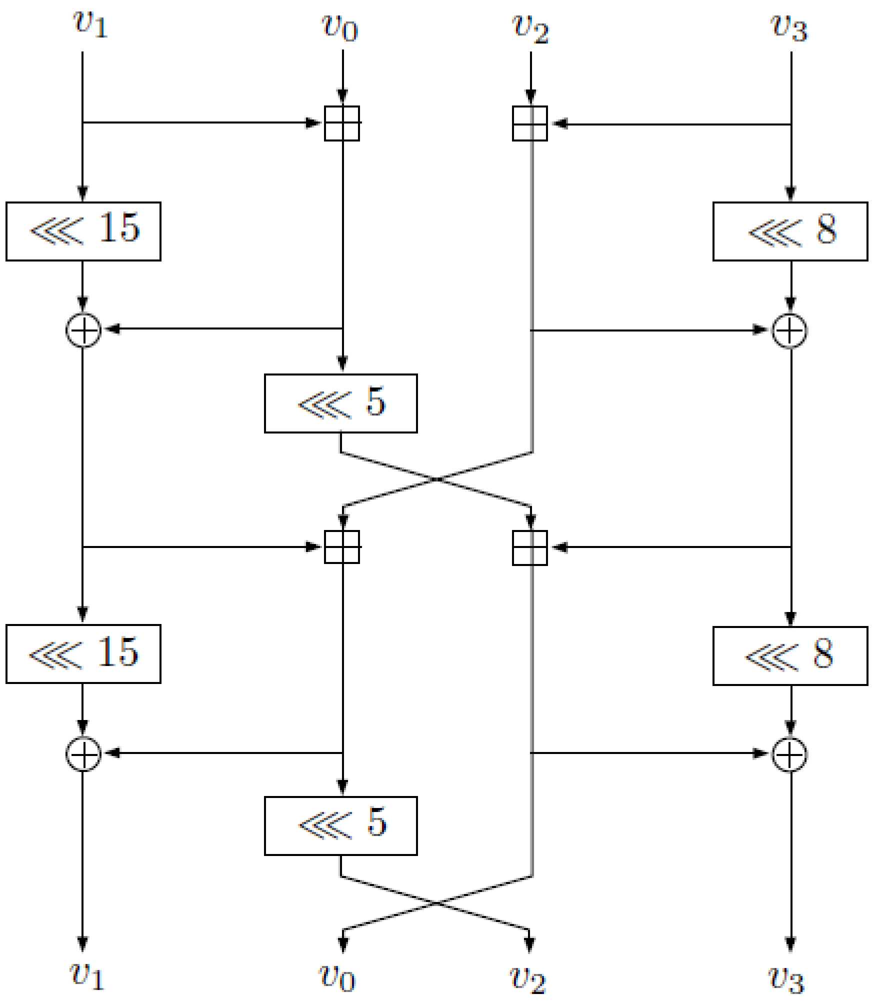
- Separate message into two independent parts and process each of them with two PRFs (CMAC with PRINCE or PRESENT).
- XOR two PRF outputs and apply a public one-way function.





MergeMAC Specification 2

- π of MergeMAC is 3 rounds of the Chaskey permutation.
- Chaskey consists of 8 rounds and is attacked up to 7 rounds.
- 3-round Chaskey itself is weak.



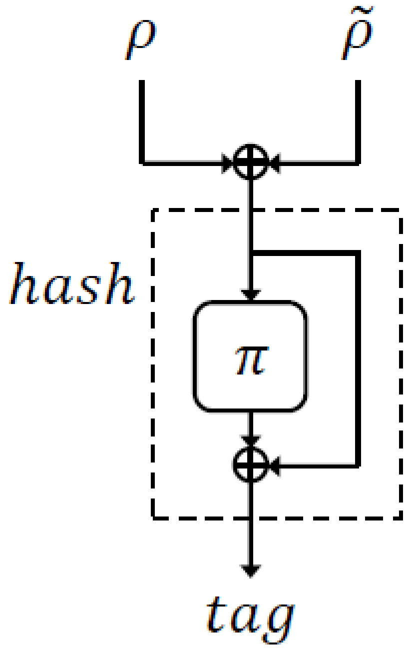
1-round Chaskey

Security Analysis by the Designers

- The main feature to ensure security is the entropy reduction from $2n$ bits to n bits when XORing two PRF's outputs.
- For any tag, it is impossible to know the correct ρ and $\tilde{\rho}$.

Table 3. Security claims according to the underlying primitives [1, Table 1].

Underlying BC	Block size	Key size	Existential forgery resistance
PRESENT	64	80	2^{-64}
PRESENT	64	128	2^{-64}
PRINCE	64	128	2^{-64}



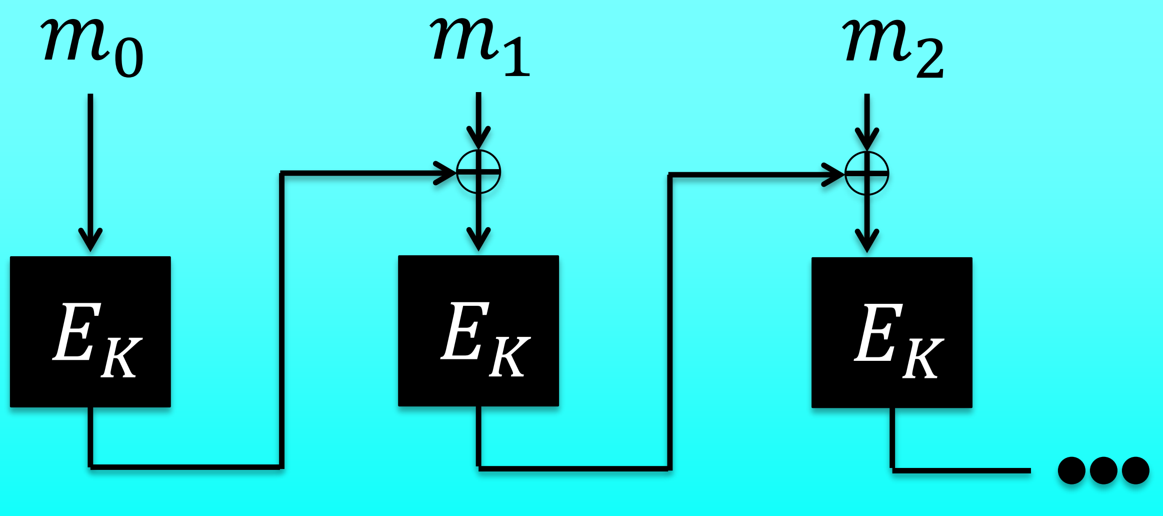
RSAConference2019

Generic Universal Forgery with High Data Complexity

General Universal Forgery (1/2)

- Two PRFs are based on CMAC.
- CMAC allows universal forgery with $O(2^{\frac{n}{2}})$ data complexity.
- It applies to MergeMAC directly.

$CMAC$ for
$m = m_0|m_1|m_2|\cdots$



General Universal Forgery (2/2)

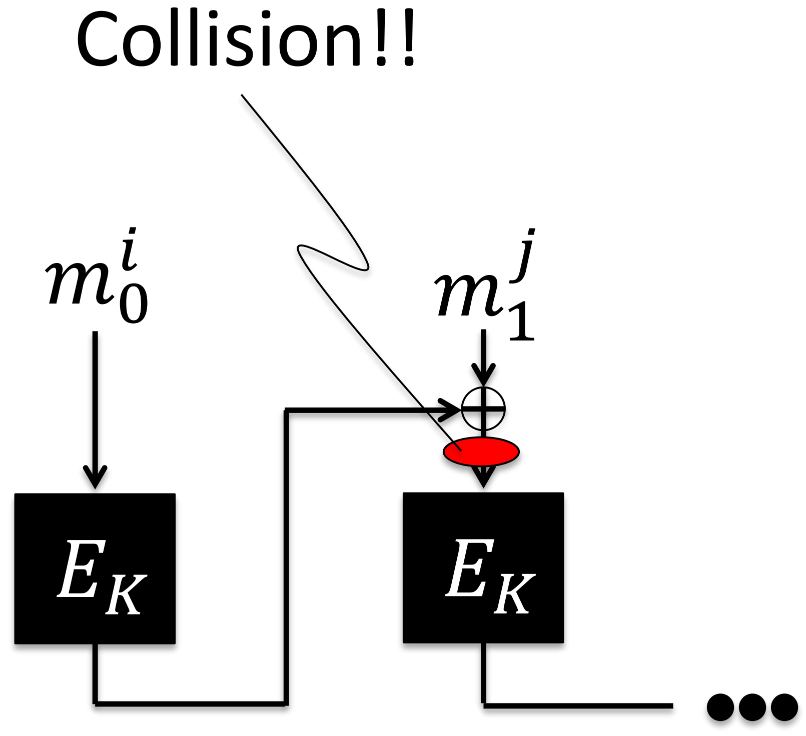
Suppose that the target message is $m = m_0 | m_1 | m_2 | \cdots$

1. Make $O(2^{\frac{n}{2}})$ queries of form $m_0^i | m_1 | m_2 | \cdots$.
2. Make $O(2^{\frac{n}{2}})$ queries of form $m_0 \, | m_1^j | m_2 | \cdots$.
3. Find a collision of the tags between Steps 1 and 2.
4. Query $m_0^i | m_1^j | m_2 | \cdots$, which collides with m.

$$E_K(m_0^i) \oplus m_1 = E_K(m_0) \oplus m_1^j$$

$$\Downarrow$$

$$E_K(m_0 \,) \oplus m_1 = E_K\left(m_0^i\right) \oplus m_1^j$$

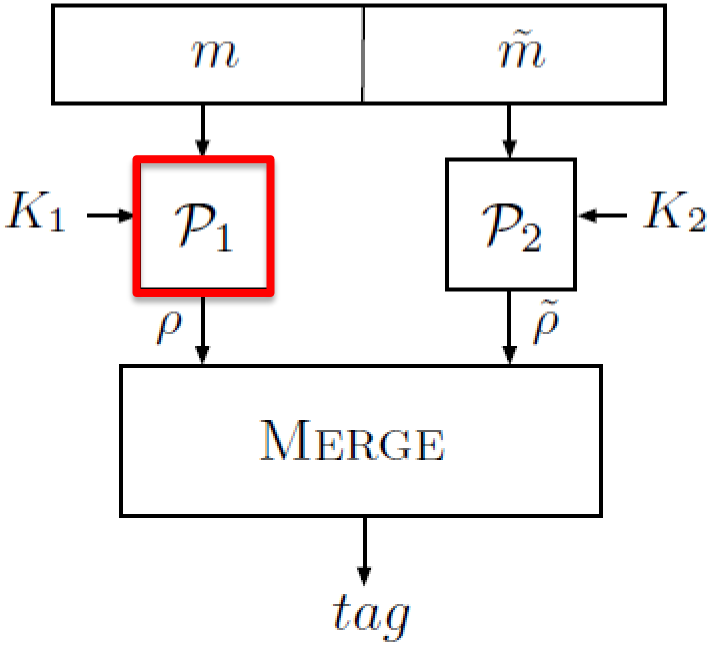


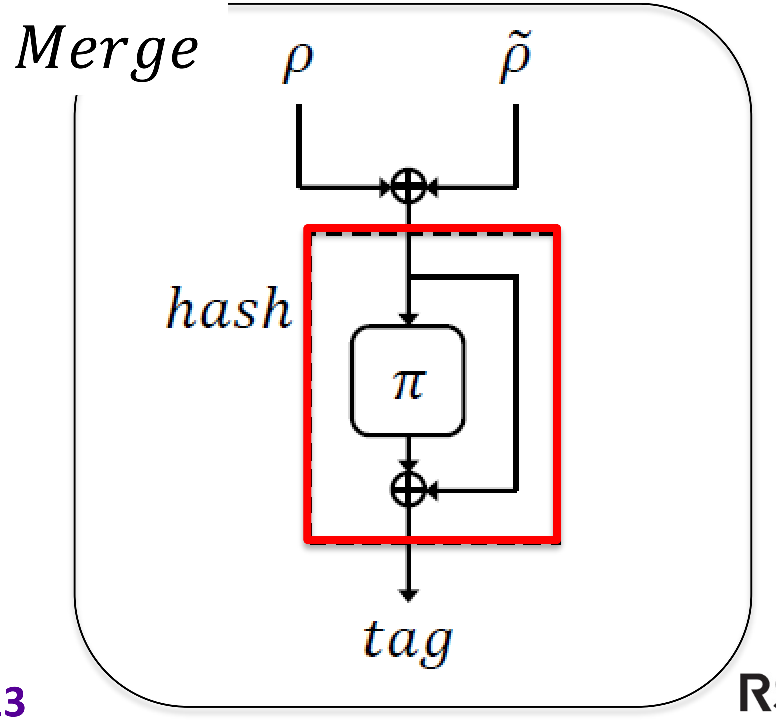
RSAConference2019

Universal Forgery on MergeMAC with Low Data Complexity

MergeMAC Specification 1

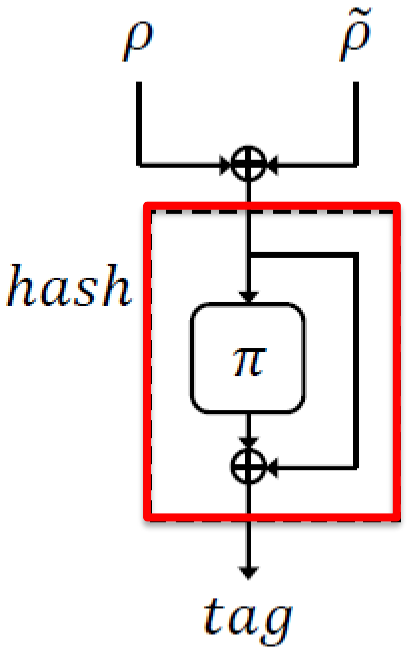
- Previous attack exploits the property of the underlying PRF, and requires about 2^{32} data, which may be too high for IoT usage.
- To go a different direction, we now invert the merge function, which is weak (3-round Chaskey).





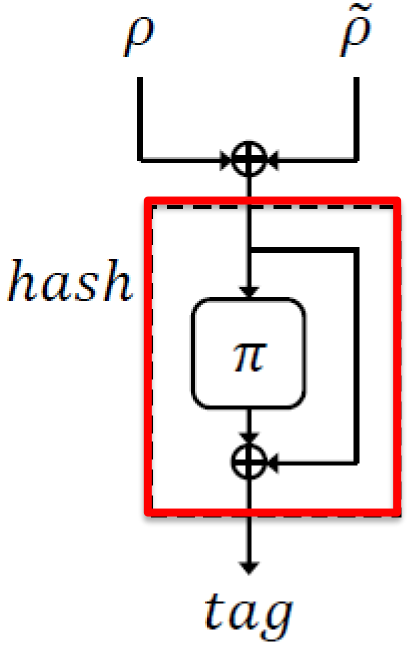
Overview

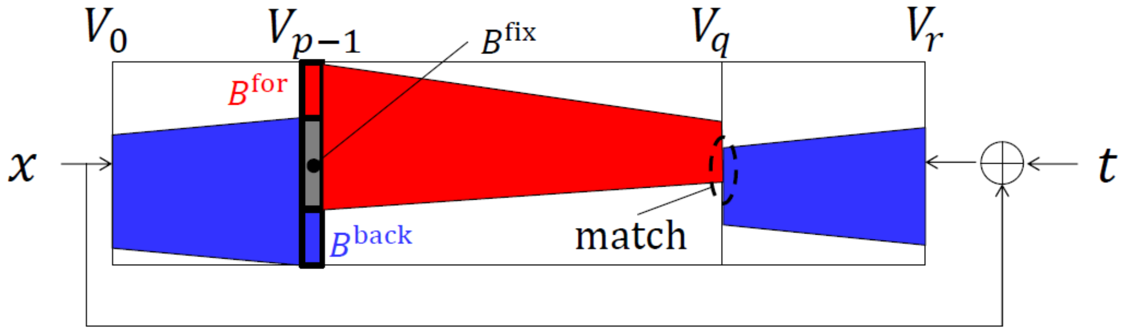
- Suppose that we can invert the hash function from a tag t_i.
- Suppose that the target message is $m||\widetilde{m}$.
- $\rho \oplus \tilde{\rho}$ can be recovered by 3 queries and 3 preimage attack.
  - From t_1 for $x||\widetilde{m}$, we obtain $\mathcal{P}_1(x) \oplus \mathcal{P}_2(\widetilde{m})$.
  - From t_2 for $m||\tilde{y}$, we obtain $\mathcal{P}_1(m) \oplus \mathcal{P}_2(\tilde{y})$.
  - From t_3 for $x||\tilde{y}$, we obtain $\mathcal{P}_1(x) \oplus \mathcal{P}_2(\tilde{y})$.
- The XOR of three gives $\mathcal{P}_1(m) \oplus \mathcal{P}_2(\widetilde{m})$.



Preimage Attacks on Davies-Mayer Constructions

- A lot of preimage attacks against the Davies-Mayer constructions ($H(a) \oplus a$) were studied around 2008 to 2010, e.g. preimage resistance of MD5 was broken in 2009 [SA09].
- The finalization of MergeMAC can be seen as Davies-Mayer.
- The same technique can be applied!
- Meet-in-the-Middle Preimage Attacks
  - Splice-and-Cut, Partial-fixing [AS08], Initial-Structure [SA09]



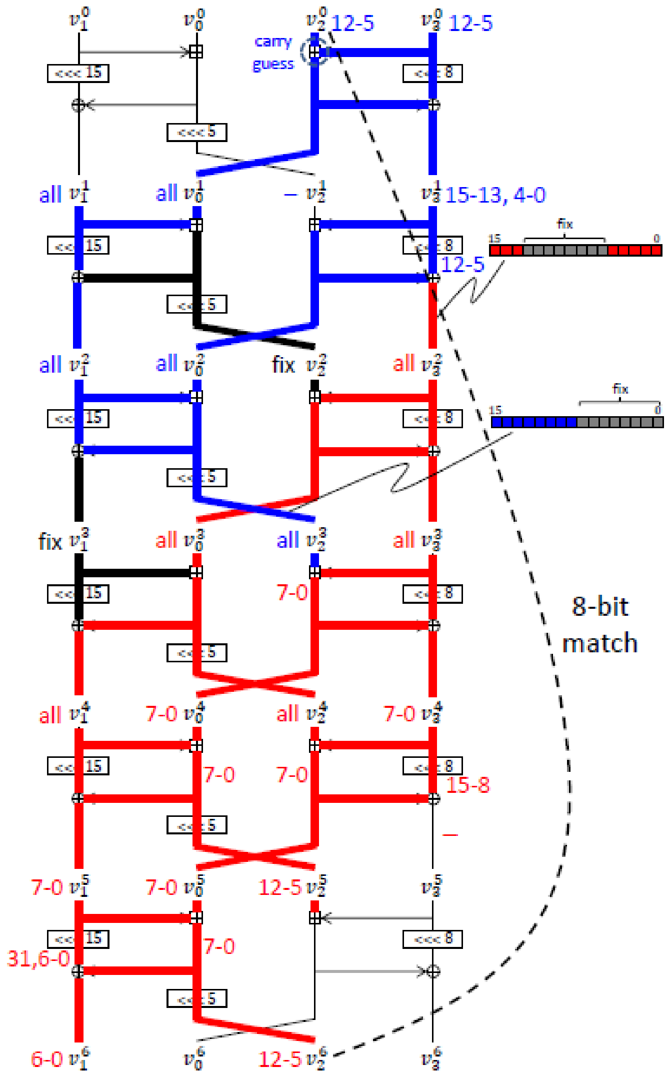


MitM Preimage Attacks on 3-Round Chaskey in DM

Intuition

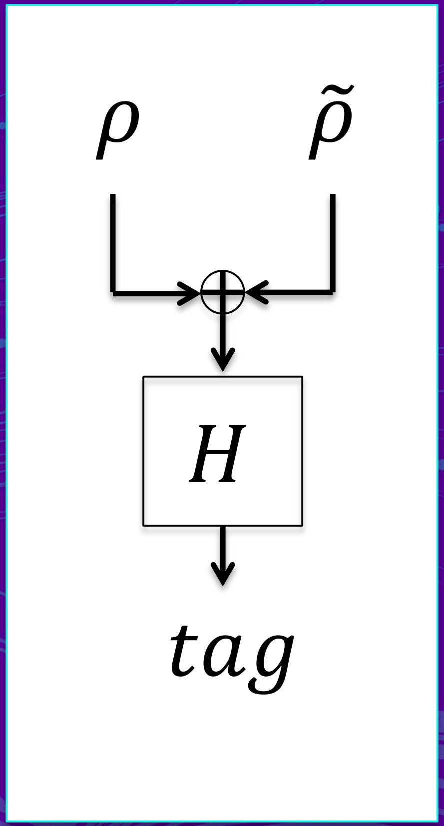
- Computation of 3-round Chaskey is divided into three parts.
- The blue part is independently computed from 8 internal state bits.
- The red part is independently computed from 8 internal state bits.
- Two independent computation can match at 8 bits.

$$(Time, Memory) = (2^{57}, 2^{8})$$



RSAConference2019

Attacks on Generalized Variants: with secure hash function

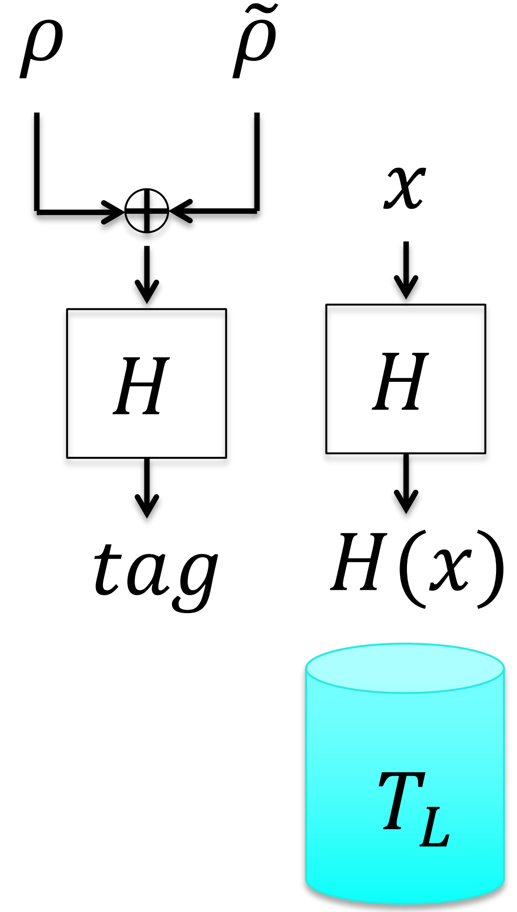


Offline Computations

- Preimage attacks no longer work.
- Precompute a look-up table.
  - H is public. $H(x)$ for many x can be computed offline.
  - $(x, H(x))$ are stored as a lookup table T_L.
- In the online phase, if tag is stored in T_L, we know the input value to H with a good probability.
- Tradeoff:

$$T^{3/2} \cdot D = 2^{3/2n}$$

- When $T < 2^{2/3n}$, D becomes less than $2^{n/2}$.
- Example: $(Data, Time) = (2^{24}, 2^{48})$



Reforgeability and 2-Dimensional Table Representation

- Our attacks require 3 queries to forge a tag for 1 message.
- Consider the ratio r :

$$\boxed{r = \frac{\#\ \text{queries}}{\#\ \text{forgeries}}}$$

- The ratio can be improved when multiple tags are forged.
- Recall that we query $m_1||\widetilde{m_1}$, $m_1||\widetilde{m_2}$, $m_2||\widetilde{m_1}$ to forge $m_2||\widetilde{m_2}$.
- This can be represented in the matrix.

$$\begin{array}{cc|cc} & & \multicolumn{2}{c}{j} \\ & & 1 & 2 \\ \hline \multirow{2}{*}{i} & 1 & Q & Q \\ & 2 & Q & X \end{array}$$

Reforgeability (Existential Forgery)

- Extension to 5 queries.
  - 4 tags can be forged.

$$
\begin{array}{cc|ccc}
 & & & j & \\
 & & 1 & 2 & 3 \\
\hline
 & 1 & Q & Q & Q \\
i & 2 & Q & & \\
 & 3 & Q & &
\end{array}
\qquad
\begin{array}{cc|ccc}
 & & & j & \\
 & & 1 & 2 & 3 \\
\hline
 & 1 & Q & Q & Q \\
i & 2 & Q & X & X \\
 & 3 & Q & &
\end{array}
$$

- Generalization to $2q - 1$ queries.
  - $(q-1)^2$ tags can be forged.
  - #forgery is quadratic to #queries.

$$
\begin{array}{cc|cccc}
 & & & j & & \\
 & & 1 & 2 & \cdots & q \\
\hline
 & 1 & Q & Q & \cdots & Q \\
 & 2 & Q & & & \\
i & \vdots & \vdots & & & \\
 & q & Q & & &
\end{array}
$$

Reforgeability (Universal Forgery)

- Given multiple targets are represented in the diagonal.
- All of them are forged with $2q-1$ queries.

$i \backslash j$	1	2	3	4	5	6
1	X					
2		X				
3			X			
4				X		
5					X	
6						X

$i \backslash j$	1	2	3	4	5	6
1	X	Q				
2	Q	X	Q			
3		Q	X	Q		
4			Q	X	Q	
5				Q	X	Q
6					Q	X

$i \backslash j$	1	2	3	4	5	6
1	X_2	Q	Q			
2	Q	X_1	Q			
3		Q	X_3	Q		
4			Q	X_4	Q	
5				Q	X_5	Q
6					Q	X_6

- The ratio r is 2, which is better than single-target case.

RSAConference2019

Concluding Remarks

Concluding Remarks: Lessons from This Talk

- We presented several attacks on MergeMAC and its generalized construction.
- Do not implement MergeMAC, because it is not secure.
- When you design a new MAC scheme, do not remove the key from the finalization function.
- To design lightweight MAC schemes is still a challenging topic.