Improved Generic Attacks Against Hash-based MACs and HAIFA

Itai Dinur[1] Gaëtan Leurent[2]

[1] ENS, France

[2] Inria, France

Crypto 2014

HMAC with GOST

HMAC

- Very common MAC algorithm

$H(K \oplus \texttt{opad} \,\|\, H(K \oplus \texttt{ipad} \,\|\, M))$

GOST R 34.11-94

- Russian hash funct. standard
- Uses an internal checksum

HMAC-GOST

- Expect ℓ bit security for key-recovery (ℓ-bit state, key, tag)
- Key recovery attack in $2^{3\ell/4}$ [LPW, AC 2013]

HMAC with GOST

HMAC

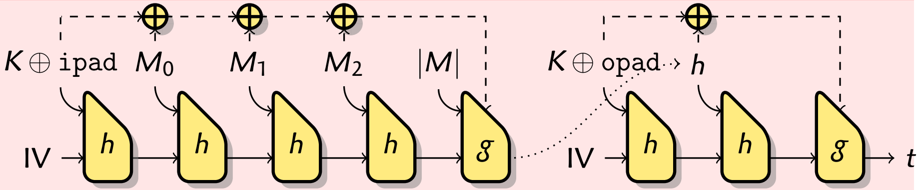
- Very common MAC algorithm

$H(K \oplus \mathtt{opad} \,\|\, H(K \oplus \mathtt{ipad} \,\|\, M))$

GOST R 34.11-94

- Russian hash funct. standard
- Uses an internal checksum

HMAC-GOST



- Expect ℓ bit security for key-recovery (ℓ-bit state, key, tag)
- Key recovery attack in $2^{3\ell/4}$ [LPW, AC 2013]

HMAC with GOST

HMAC

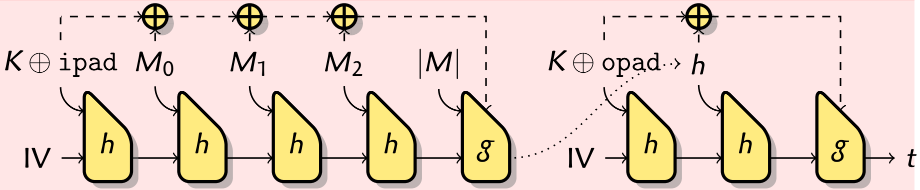
- Very common MAC algorithm

$H(K \oplus \texttt{opad} \,\|\, H(K \oplus \texttt{ipad} \,\|\, M))$

GOST R 34.11-94

- Russian hash funct. standard
- Uses an internal checksum

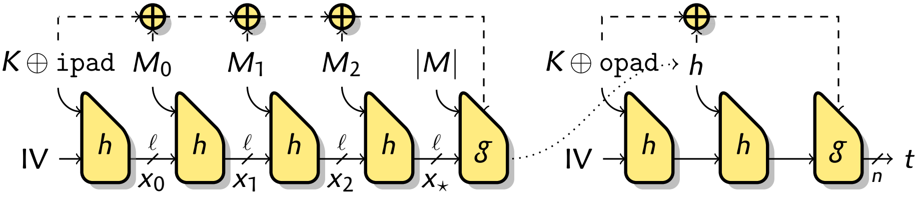
HMAC-GOST



- Expect ℓ bit security for key-recovery (ℓ-bit state, key, tag)
- Key recovery attack in $2^{3\ell/4}$ [LPW, AC 2013]

HMAC-GOST key recovery

- GOST uses an internal checksum

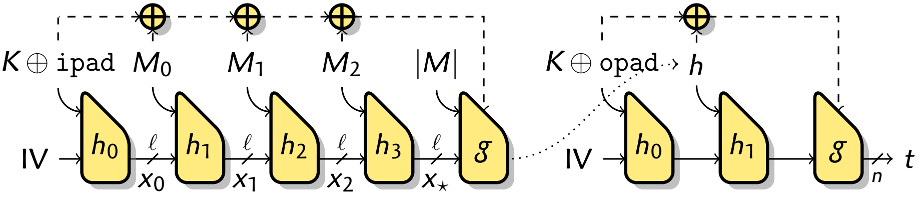


Key recovery attack

- Use a state-recovery attack to recover $x_\star$ [LPW, AC 2013]
- Chosen message difference gives chosen checksum difference
- "Related-key attack" on the finalization

HMAC-GOST key recovery

- GOST uses an internal checksum



Question

- The is a new GOST hash function: Streebog
  - Also has a checksum
  - Uses a block-counter (HAIFA)
- Can we build a key-recovery attack against HMAC-Streebog?

Security of HMAC

- Security proof up to $2^{\ell/2}$
- Matching attack for existential forgery
- We used to assume that many harder attack should cost 2^{ℓ}

Recent work

- State-recovery attack
  - $2^{\ell}/\ell$ using multi-collisions [NSWY13]
  - $2^{\ell/2}$ using the cycle structure of random graphs [LPW12]
- Universal forgery attack
  - $2^{5\ell/6}$ using the cycle structure of random graphs [PW14]
  - $2^{3\ell/4}$ improvement [CPSW14]

Limitations of recent attacks

In this work we address two important limitations of recent attacks:

1. Attacks are not applicable to HAIFA-based hash function
   - Compression function tweak for each block (counter)
   - Used in Blake, Skein, Streebog, ...

2. Most of these attack use queries of length $\approx 2^{\ell/2}$
   - In practice, many hash functions limit the message length
     e.g. 2^{55} blocks for SHA-1 ($\ell = 160$) and SHA-256 ($\ell = 256$)

Outline

Introduction

- HMAC-GOST
- Recent work

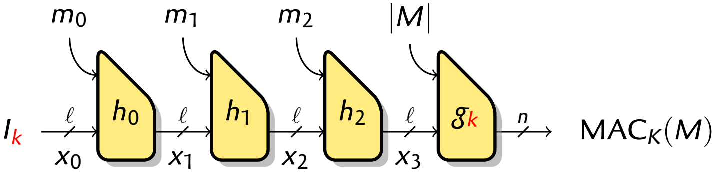
State-recovery for HMAC-HAIFA

- Previous work
- New results

Short message attacks

- State-recovery
- Universal forgery

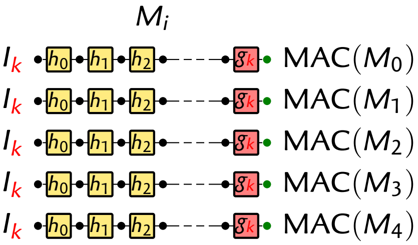
Hash-based MAC with a HAIFA hash function



- Generic model (HMAC, Sandwich-MAC, Envelope-Mac)
- Unkeyed compression functions h_i
  - Each compression function is different with HAIFA
- ℓ-bit internal state
- Key dependant initialization I_k
- Key dependant finalization g_k

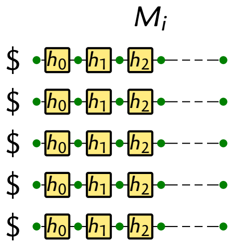
State-recovery attacks

- Send messages to the oracle



Online Structure

- Do some computations offline with the compression function



Offline Structure

- Match the sets of points?
  - How to test equality? Online chaining values unknown
  - How many equality test do we need?

Special states

Special states in a small set are more likely to match

Previous work *[LPW14]*

- Entry point of the main cycle (1 point)
- Collisions found with long chains ($2^{\ell-2s}$ points)

Not applicable to HAIFA

We use the entropy loss from iterations of random function

Theorem (Entropy loss)

Let $f_1, f_2, \ldots, f_{2^s}$ be a fixed sequence of random functions; the image of $g_{2^s} \triangleq f_{2^s} \circ \ldots \circ f_2 \circ f_1$ contains about $2^{\ell-s}$ points.

cf. [PK14]

Special states

Special states in a small set are more likely to match

Previous work *[LPW14]*

- Entry point of the main cycle (1 point)
- Collisions found with long chains ($2^{\ell-2s}$ points)

Not applicable to HAIFA

We use the entropy loss from iterations of random function

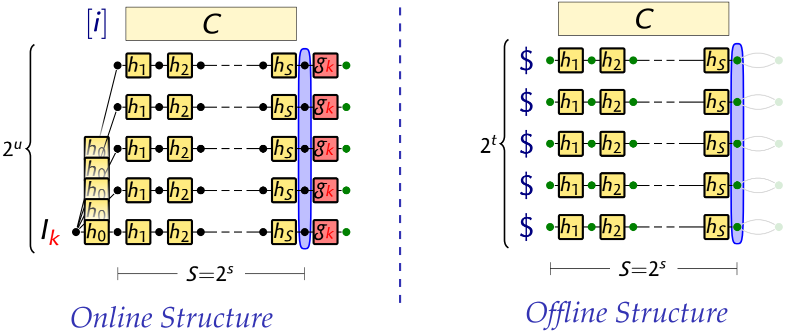
Theorem (Entropy loss)

Let $f_1, f_2, \ldots, f_{2^s}$ be a fixed sequence of random functions; the image of $g_{2^s} \triangleq f_{2^s} \circ \ldots \circ f_2 \circ f_1$ contains about $2^{\ell-s}$ points.

cf. [PK14]

First attempt

- Chains of length 2^s, with a fixed message C



1. Evaluate 2^t chains offline
   Build filters for endpoints
2. Query 2^u message $M_i = [i] \parallel C$
   Test endpoints with filters

$s + t + u = \ell$

Cplx: 2^{s+t+u}

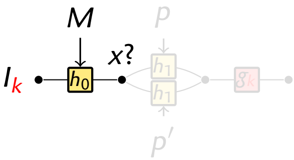
Building filters

Filters to compare online and online states

Test whether the state reached after processing M is equal to x

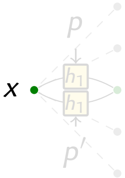
- Collisions are preserved by the finalization
  (for same-length messages)

2 $\mathrm{MAC}(M \| p) \stackrel{?}{=} \mathrm{MAC}(M \| p')$



Online Structure

1 Find a collision:
$h(x, p) = h(x, p')$



Offline Structure

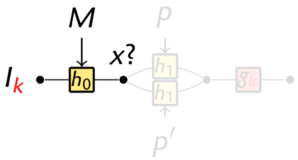
Building filters

Filters to compare online and online states

Test whether the state reached after processing M is equal to x

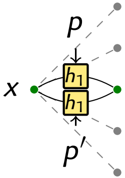
- Collisions are preserved by the finalization (for same-length messages)

2 $\mathrm{MAC}(M \| p) \stackrel{?}{=} \mathrm{MAC}(M \| p')$



Online Structure

1 Find a collision: $h(x, p) = h(x, p')$



Offline Structure

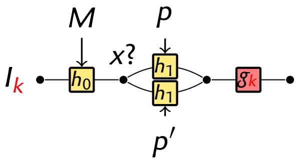
Building filters

Filters to compare online and online states

Test whether the state reached after processing M is equal to x

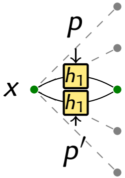
- Collisions are preserved by the finalization (for same-length messages)

2 $\mathrm{MAC}(M \| p) \stackrel{?}{=} \mathrm{MAC}(M \| p')$



Online Structure

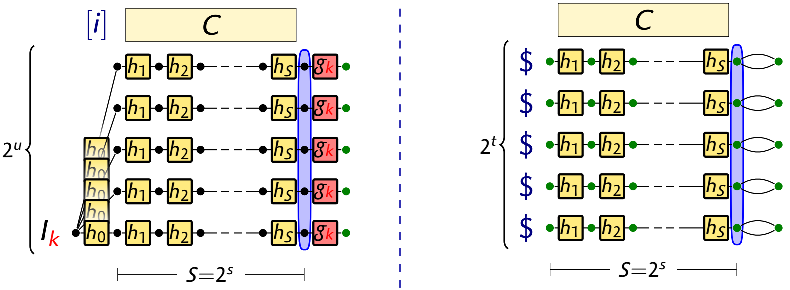
1 Find a collision:
$h(x, p) = h(x, p')$



Offline Structure

First attempt

- Chains of length 2^s, with a fixed message C



Online Structure

Offline Structure

1. Evaluate 2^t chains offline
   Build filters for endpoints
2. Query 2^u message $M_i = [i] \parallel C$
   Test endpoints with filters

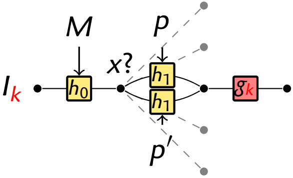
$s + t + u = \ell$

Cplx: 2^{s+t+u}

Online filters

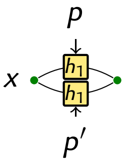
- Using the filters is too expensive.
- If we build filters online, using them is cheap.

1 Find p, p' s.t.
$\mathrm{MAC}(M \| p) = \mathrm{MAC}(M \| p')$



Online Structure

2 $h(x, m) \stackrel{?}{=} h(x, m')$

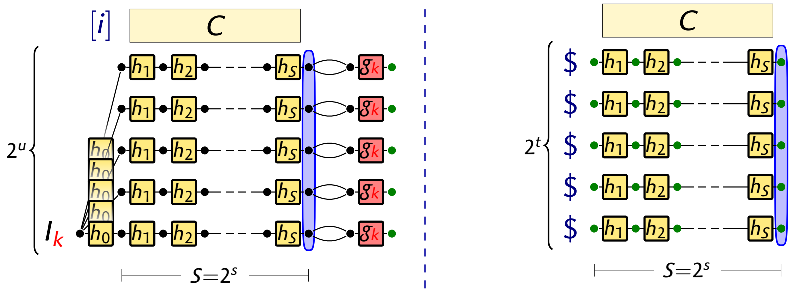


Offline Structure

Cost	Build	Test
Offline filter	$2^{\ell/2}$	2^s
Online filter	$2^{\ell/2+s}$	1

First attack on HMAC-HAIFA

- Chains of length 2^s, with a fixed message C



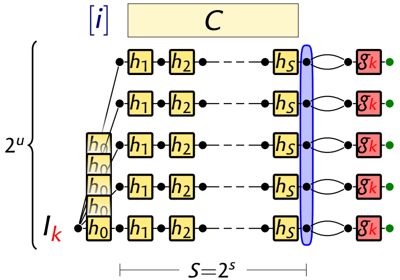
Online Structure | *Offline Structure*

1. Query 2^u message $M_i = [i] \parallel C$
   Build filters for M_i — Cplx: $2^{s+u+\ell/2}$
2. Evaluate 2^t chains offline — Cplx: 2^{t+s}
   Test endpoints with filters — Cplx: 2^{t+u}

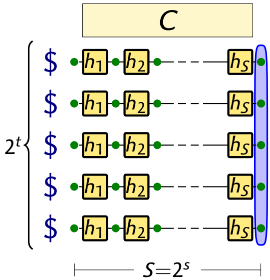
$s + t + u = \ell$

First attack on HMAC-HAIFA

- Chains of length 2^s, with a fixed message C



Online Structure



Offline Structure

1. Query 2^u message $M_i = [i] \parallel C$
   Build filters for M_i
2. Evaluate 2^t chains offline
   Test endpoints with filters

Optimal complexity

$2^{\ell - s}$, for $s \leq \ell/6$
(using $u = s$)
Minimum: $2^{5\ell/6}$

Diamond filters

- Building filers is a bottleneck.
- Can we amortize the cost of building many filters?

Diamond structure [Kelsey & Kohno, EC'06]

Herd N initial states to a common state

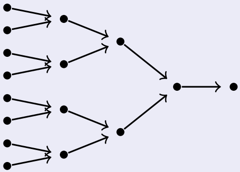
- Try $\approx 2^{\ell/2}/\sqrt{N}$ msg from each state.
- Whp, the initial states can be paired
- Repeat...

Total $\approx \sqrt{N} \cdot 2^{\ell/2}$

Diamond filters

- Building filers is a bottleneck.
- Can we amortize the cost of building many filters?

Diamond structure [Kelsey & Kohno, EC'06]



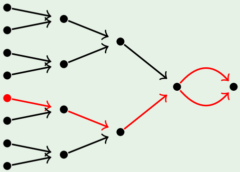
Herd N initial states to a common state

- Try $\approx 2^{\ell/2}/\sqrt{N}$ msg from each state.
- Whp, the initial states can be paired
- Repeat... Total $\approx \sqrt{N} \cdot 2^{\ell/2}$

Diamond filters

- Building filers is a bottleneck.
- Can we amortize the cost of building many filters?

Diamond filter



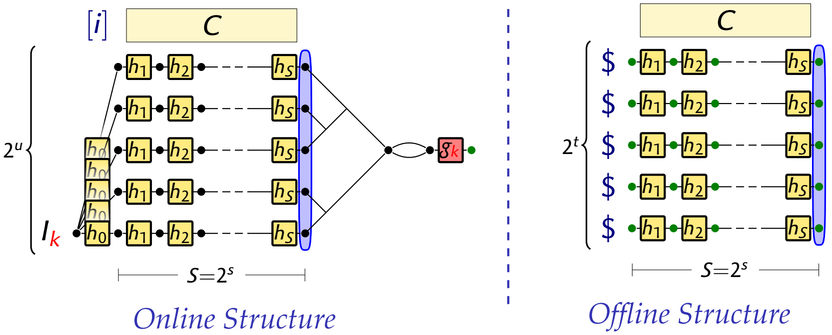
1. Build a diamond structure
2. Build a collision filter for the final state

- Can also be built online

- Building N offline filters: $\sqrt{N} \cdot 2^{\ell/2}$ rather than $N \cdot 2^{\ell/2}$
- Building N online filters: $\sqrt{N} \cdot 2^{\ell/2+s}$ rather than $N \cdot 2^{\ell/2+s}$

Improved attack on HMAC-HAIFA

- Chains of length 2^s, with a fixed message C



1. Query 2^u message $M_i = [i] \parallel C$
   Build diamond filter for M_i
2. Evaluate 2^t chains offline
   Test endpoints with filters

$s + t + u = \ell$

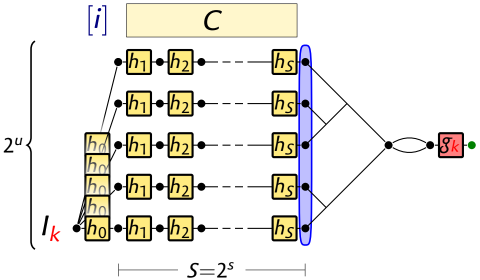
Cplx: $2^{s+u/2+\ell/2}$

Cplx: 2^{t+s}

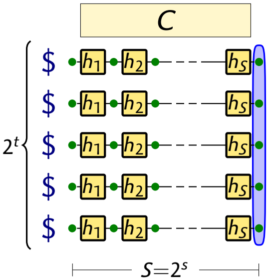
Cplx: 2^{t+u}

Improved attack on HMAC-HAIFA

- Chains of length 2^s, with a fixed message C



Online Structure



Offline Structure

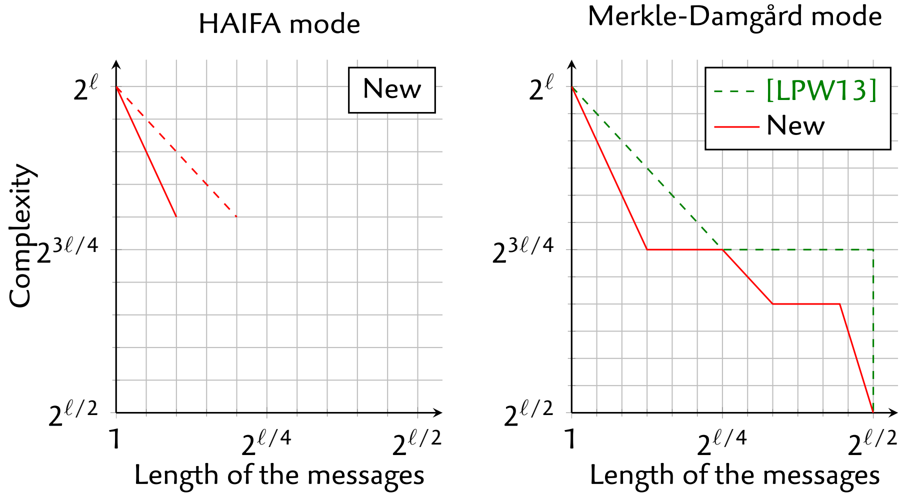
1. Query 2^u message $M_i = [i] \parallel C$
   Build diamond filter for M_i
2. Evaluate 2^t chains offline
   Test endpoints with filters

Optimal complexity

$2^{\ell-s}$, for $s \leq \ell/5$
(using $u = s$)
Minimum: $2^{4\ell/5}$

Outline

Improved trade-offs for state-recovery attacks



Improved universal forgery

- Previous universal forgery attacks require long message
- Using the techniques developed in this paper,
  we show attacks with short messages.

	Length			
Ref	Challenge	Queries	Complexity	Min
[PW14]	2^t	$2^{\ell/2}$	$2^{\ell-t}, t < \ell/6$	$2^{5\ell/6}$
[CPSW14]	2^t	$2^{\ell/2}$	$2^{\ell-t}, t < \ell/4$	$2^{3\ell/4}$
New	2^t	2^{2t}	$2^{\ell-t}, t < \ell/7$	$2^{6\ell/7}$
New	2^{2t}	2^{2t}	$2^{\ell-t}, t < \ell/5$	$2^{4\ell/5}$

Conclusion

1. Improved state-recovery attacks on HMAC with Merkle-Damgård
   - Reduced complexity when the message length is limited
     e.g. SHA-1, SHA-2, HAVAL, Whirlpool, ...

2. Improved universal-forgery on HMAC with Merkle-Damgård
   - Applicable with limited message length
     e.g. SHA-1, SHA-2, HAVAL, Whirlpool, ...

3. State-recovery attack on HMAC with HAIFA
   - Key-recovery against HMAC-Streebog-512 with complexity 2^{419}
   - State-recovery for BLAKE, Skein, ...

Attack complexity

Function	Mode	ℓ	s	State-recovery [LPW13]	State-recovery New	Universal forgery [CSPW14]	Universal forgery New
SHA-1	MD	160	2^{55}	2^{120}	2^{107}	N/A	2^{132}
SHA-256	MD	256	2^{55}	2^{201}	2^{192}	N/A	2^{228}
SHA-512	MD	512	2^{118}	2^{394}	2^{384}	N/A	2^{453}
HAVAL	MD	256	2^{54}	2^{202}	2^{192}	N/A	2^{229}
WHIRLPOOL	MD	512	2^{247}	2^{384}	2^{283}	N/A	2^{446}
BLAKE-256	HAIFA	256	2^{55}	N/A	2^{213}	N/A	N/A
BLAKE-512	HAIFA	512	2^{118}	N/A	2^{419}	N/A	N/A
Skein-512	HAIFA	512	2^{90}	N/A	2^{419}	N/A	N/A
						Key recovery [LPW13]	Key recovery New
Streebog	HAIFA+σ	512	∞	N/A	2^{419}	N/A	2^{419}

Extra Slides

Collisions as special states

Short message attack on HMAC-HAIFA

Collisions as special states

Observation: collision finding algorithms return biased collisions.

- For a fixed function, using chains of length 2^s, the entropy of collisions decreases as $2^{\ell-2s}$
  - Conjectured in [LPW14], proven here
- For a sequence of independent functions, using chains of length 2^s, the entropy of collisions decreases as $2^{\ell-s}$
  the entropy of collisions at a fixed index decreases as $2^{\ell-2s}$

Lemma (Entropy of HAIFA collision with messages of length 2^s)

Let (x, x') and (y, y') be two pairs of chains, colliding at the same step i, with $\hat{x} = x_i = x'_i$, $\hat{y} = y_i = y'_i$.
Then $\Pr[\hat{x} = \hat{y}] = \Theta(2^{2s-\ell})$

Collisions as special states

Observation: collision finding algorithms return biased collisions.

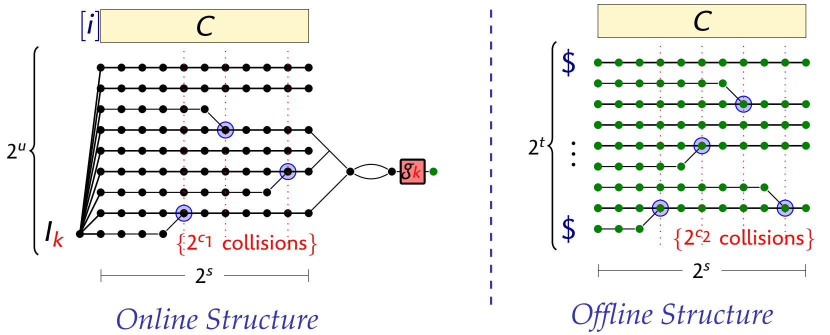
- For a fixed function, using chains of length 2^s, the entropy of collisions decreases as $2^{\ell-2s}$
  - Conjectured in [LPW14], proven here
- For a sequence of independent functions, using chains of length 2^s, the entropy of collisions decreases as $2^{\ell-s}$ the entropy of collisions at a fixed index decreases as $2^{\ell-2s}$

Lemma (Entropy of HAIFA collision with messages of length 2^s)

Let (x, x') and (y, y') be two pairs of chains, colliding at the same step i, with $\hat{x} = x_i = x'_i$, $\hat{y} = y_i = y'_i$.
Then $\Pr[\hat{x} = \hat{y}] = \Theta(2^{2s-\ell})$

Short message attack on HMAC-HAIFA

- Chains of length 2^s, with a fixed message C

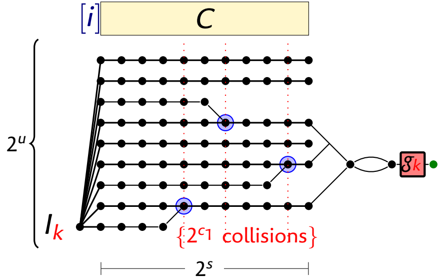


$$c_1 + c_2 + s = \ell$$

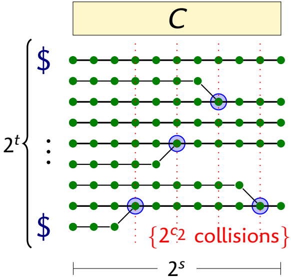
1. Locate 2^{c_1} collisions online
   Build diamond filter
   Cplx: $2^{s+c_1/2+\ell/2}$
2. Locate 2^{c_2} collisions offline
   Cplx: $2^{s/2+c_2/2+\ell/2}$
   Test with filters
   Cplx: $2^{c_1+c_2-s}$

Short message attack on HMAC-HAIFA

- Chains of length 2^s, with a fixed message C



Online Structure



Offline Structure

1. Locate 2^{c_1} collisions online
   Build diamond filter
2. Locate 2^{c_2} collisions offline
   Test with filters

Optimal complexity

$2^{\ell-2s}$, for $s \leq \ell/10$
(using $u = s$)
Minimum: $2^{4\ell/5}$