Most Recent Results on SHA-1

Christian Rechberger

Hash&Stream, Salzburg, 2007/02/01

Institute for Applied Information Processing and Communications (IAIK) - Krypto Group

Faculty of Computer Science
Graz University of Technology

Agenda

- Authentication using hash functions –
  Attacks on NMAC/HMAC-SHA-1

- New view on the problem of collision search in SHA-1
- New automated method – Results and Examples
- Extensions to (partly) meaningful collisions

- Conclusions

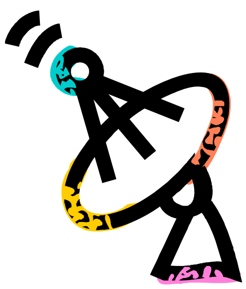
Message Authentication? Problem Description:

Alice

Bob

"Let's go to the pool at 17:00"

5242832

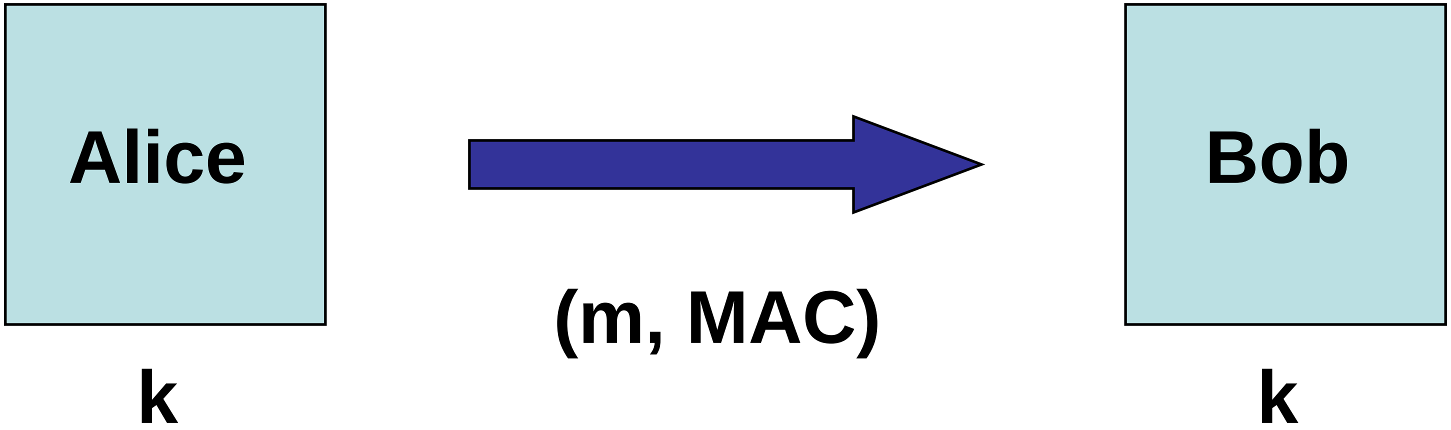


Eve

Message Authentication? Problem Description:

Message Authentication? Problem Description:

Message Authentication? Problem Description:



$$MAC = f(m,k)$$

Deducing the key should be infeasible

Alice

k

Bob

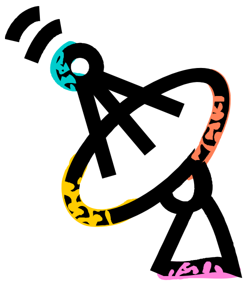
k

(m_1, MAC_1)

(m_2, MAC_2)

(m_3, MAC_3)

...

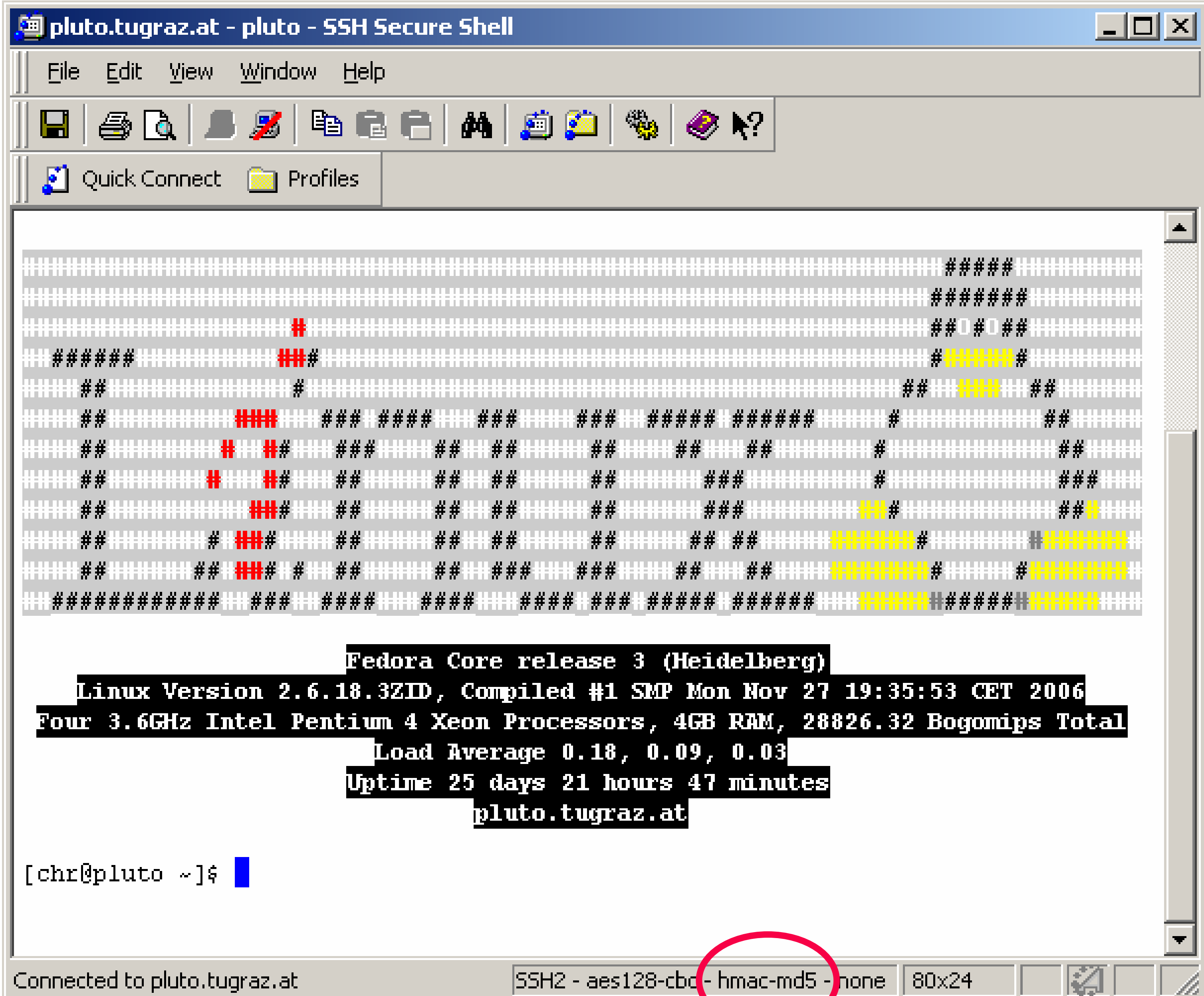


Eve

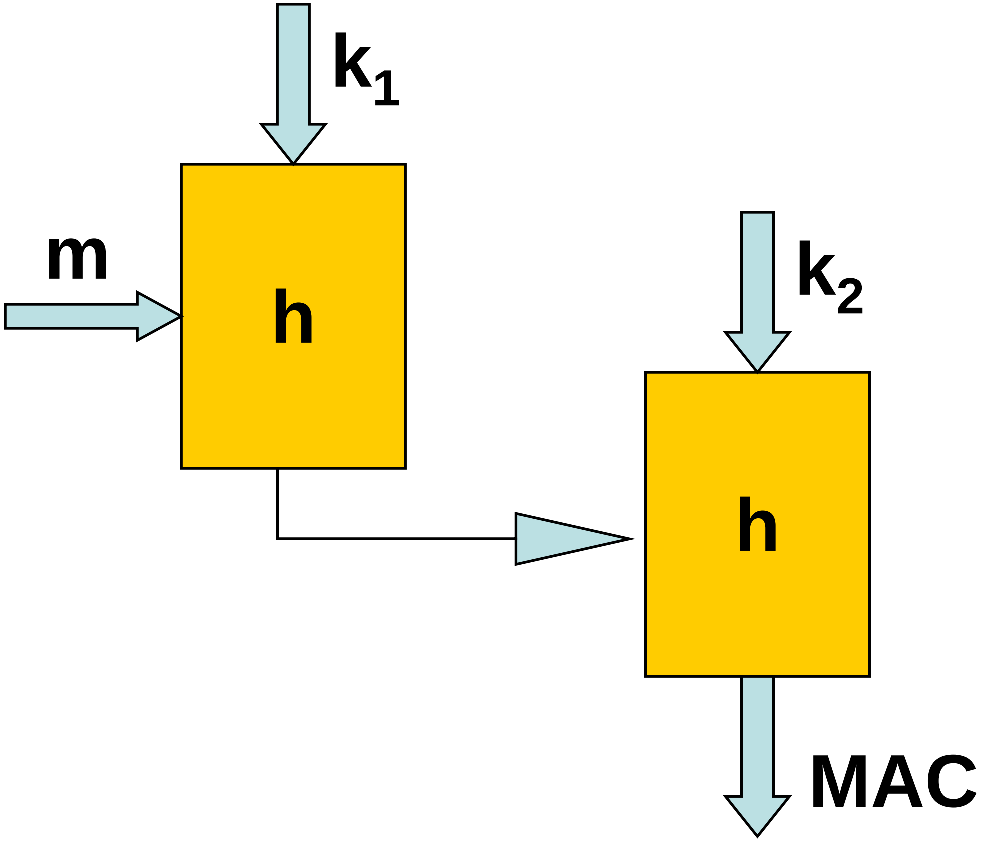
MAC ⇔ Hash

- Message Authentication Codes (MACs) based on hash functions started to be popular in the mid 90s

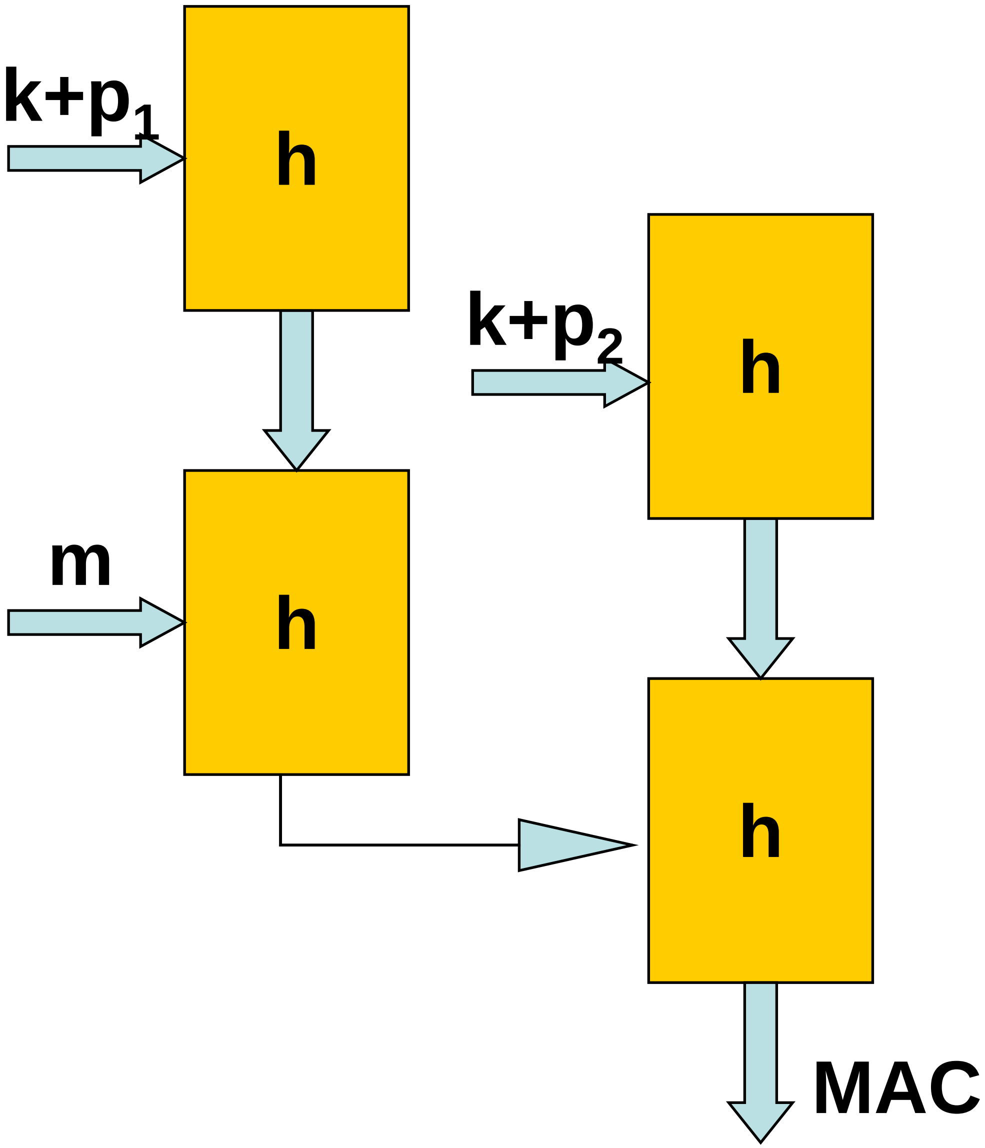
- HMAC is the most common example
  - employs hash functions like MD5 or SHA-1
  - standardized by ANSI, ETSI, FIPS, IETF, ISO,…
  - used in many products (SSH, SSL)


pluto.tugraz.at - pluto - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
Fedora Core release 3 (Heidelberg)
Linux Version 2.6.18.3ZID, Compiled #1 SMP Mon Nov 27 19:35:53 CET 2006
Four 3.6GHz Intel Pentium 4 Xeon Processors, 4GB RAM, 28826.32 Bogomips Total
Load Average 0.18, 0.09, 0.03
Uptime 25 days 21 hours 47 minutes
pluto.tugraz.at
[chr@pluto ~]$
Connected to pluto.tugraz.at
SSH2 - aes128-cbc - hmac-md5 - none
80x24

NMAC



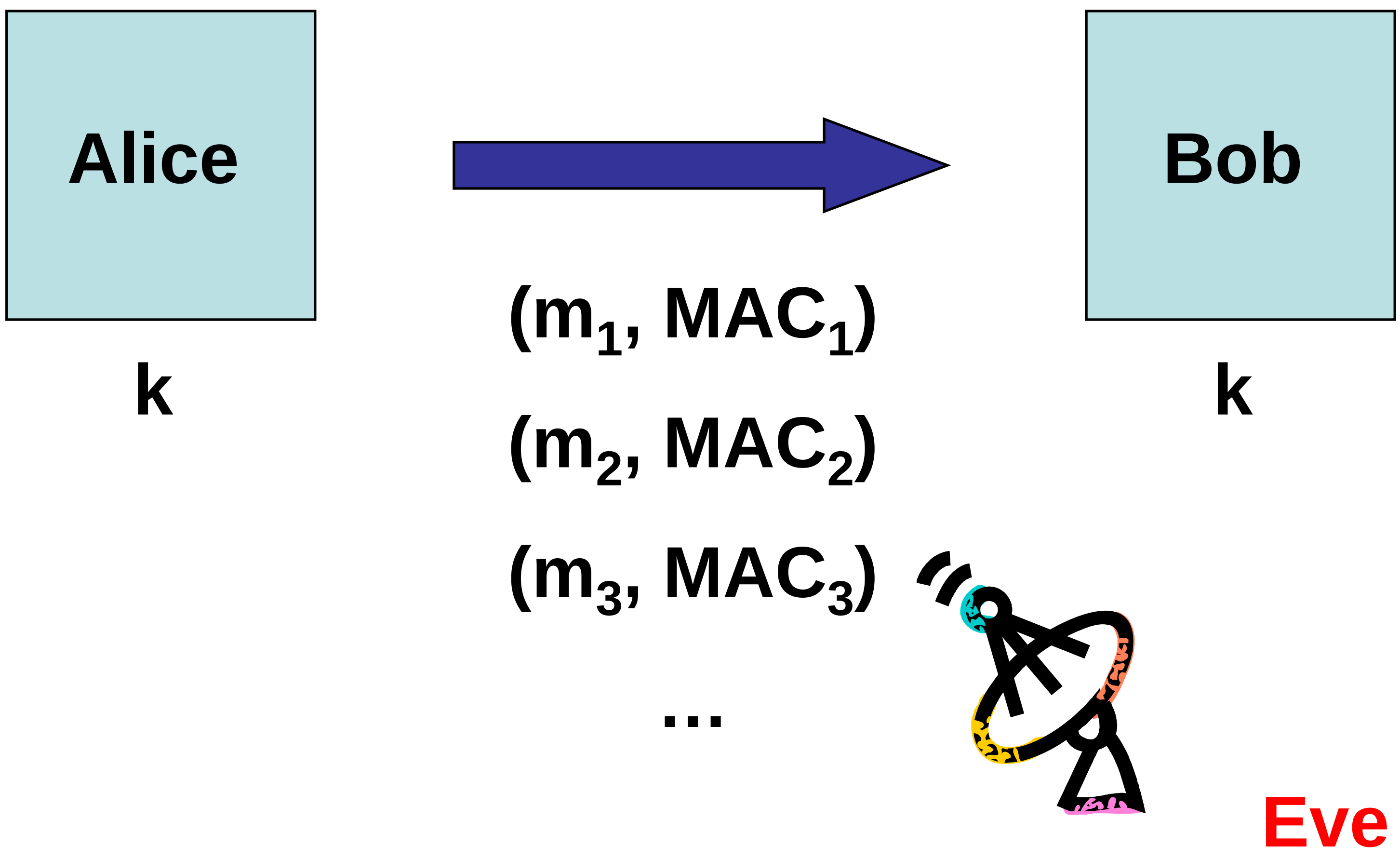
HMAC



Proven secure assuming:

- ~~h is collision resistant~~
- h has some pseudorandom properties

Deducing the key should be infeasible



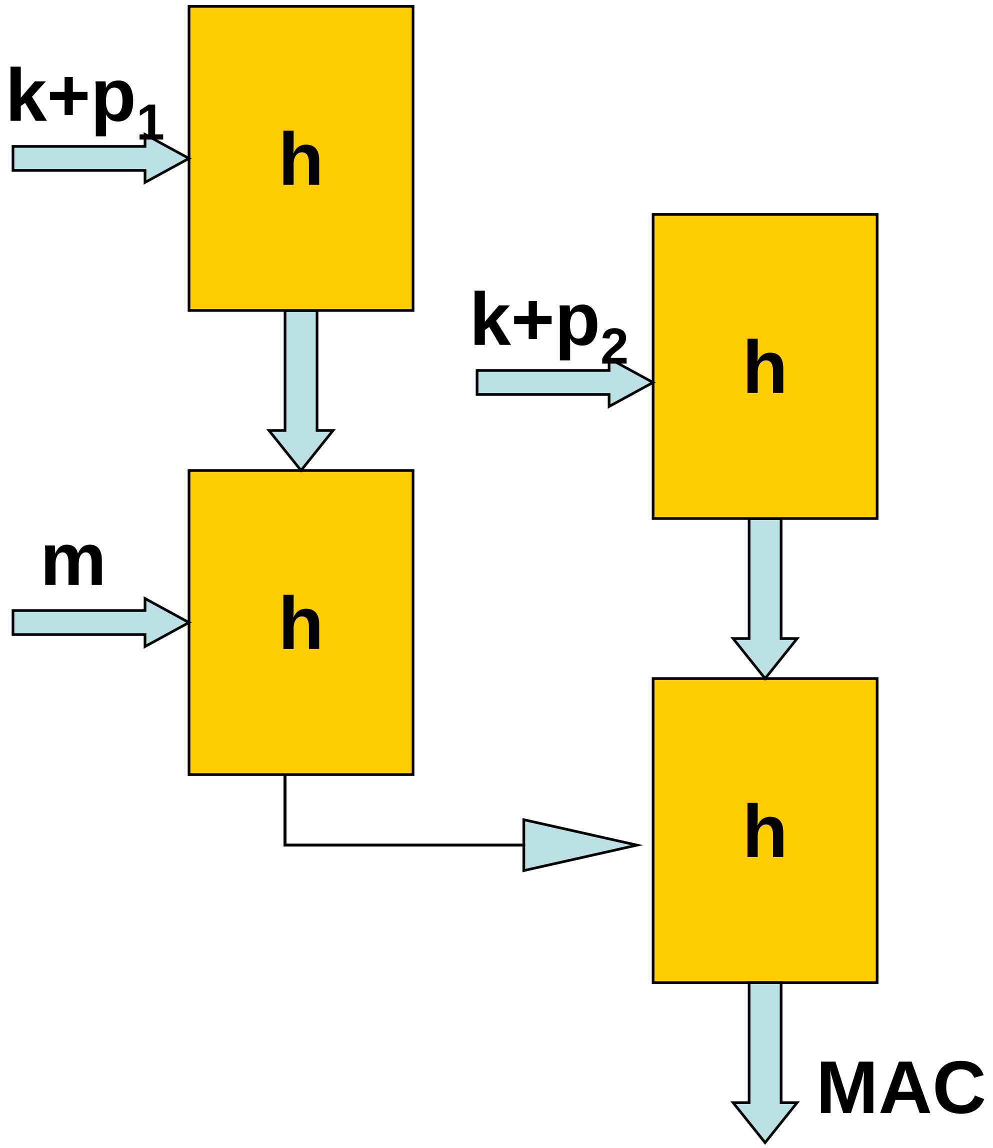
Attack

(m_1, MAC_1)

(m_2, MAC_2)

(m_3, MAC_3)

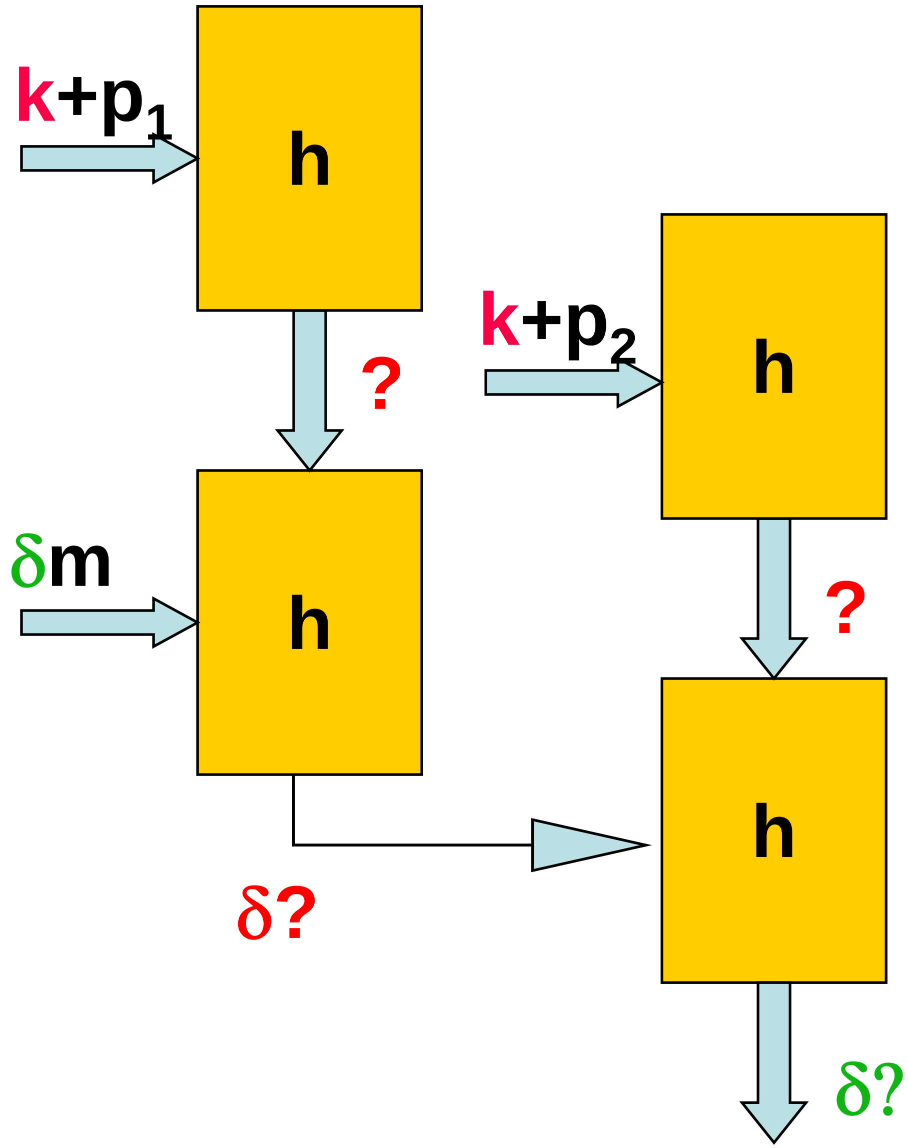
...



Attack

Known

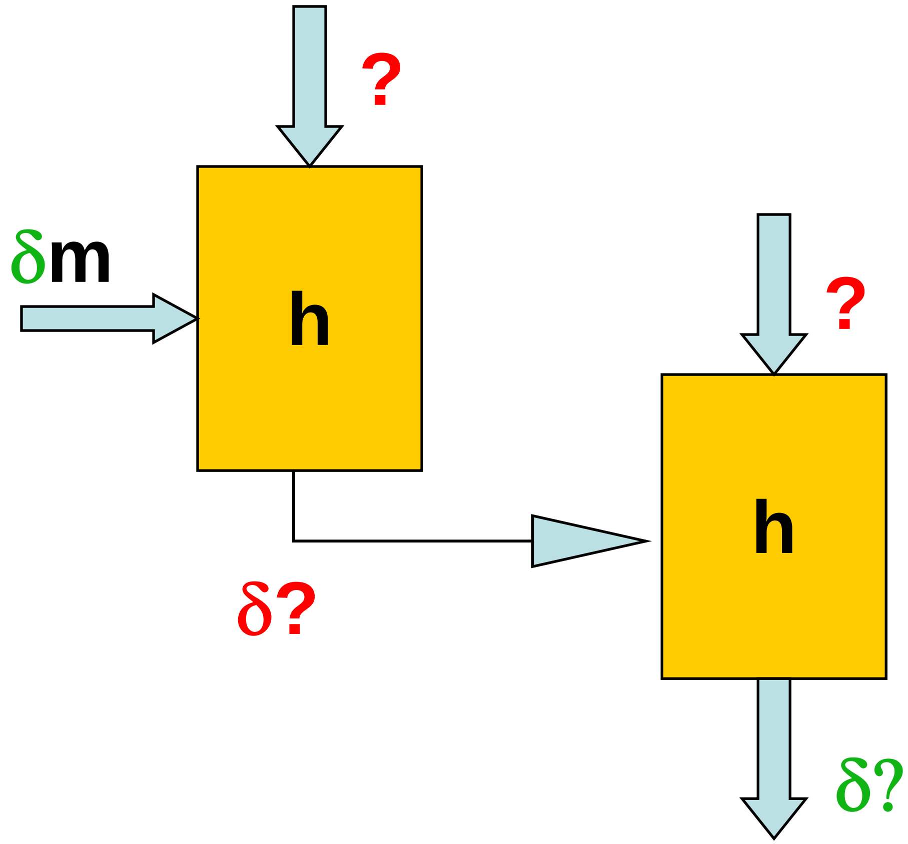
Unknown



Attack

Known

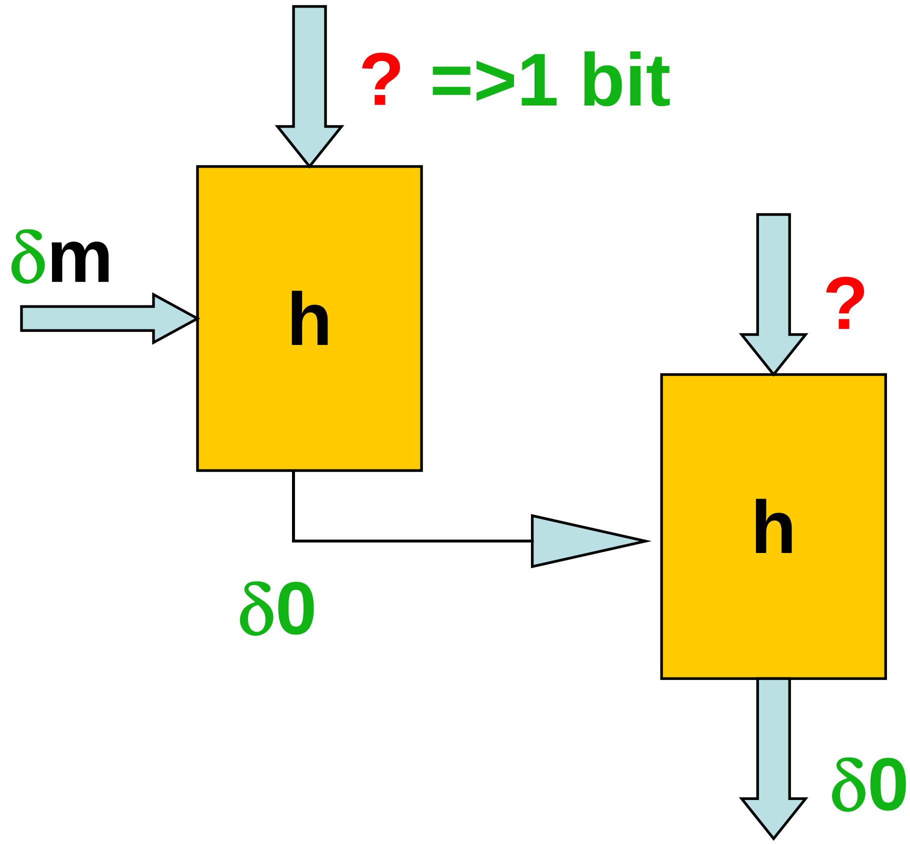
Unknown



Attack

Known

Unknown



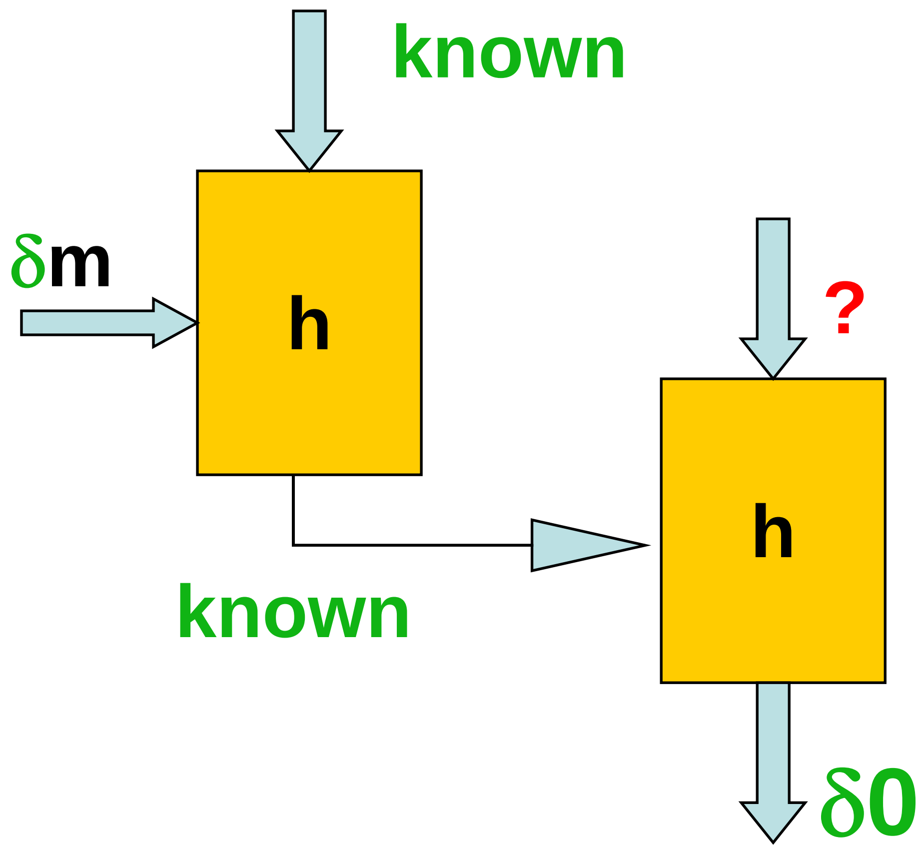
After a number of trials…

➔ 1 bit of key information recovered

Attack

Known

Unknown



Inner key recovered...

Outer key?

Results on NMAC/HMAC

- Attacks exploit non-random properties
- Compared to unkeyed hash: less severe

- Applies to NMAC/HMAC with hash functions like MD4, MD5 and reduced SHA-1
- Theoretical attacks for up to 61 steps of NMAC-SHA-1

- Some security margin left

- To be presented at Financial Cryptography 2007 (joint work with Vincent Rijmen)

Agenda

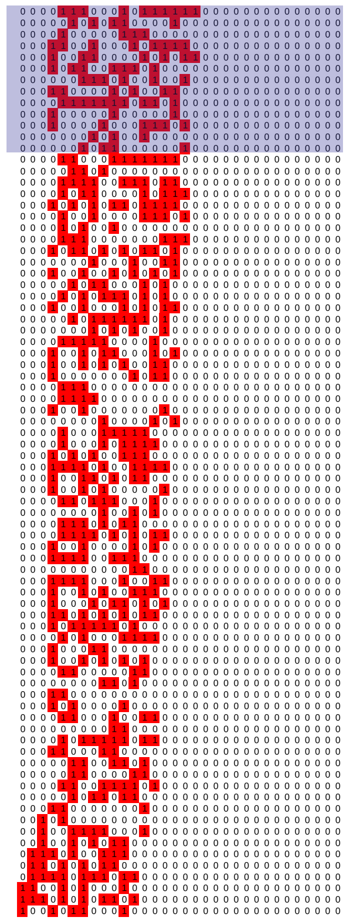
- Authentication using hash functions –
  Attacks on NMAC/HMAC-SHA-1

- New view on the problem of collision search in SHA-1
- New automated method – Results and Examples
- Extensions to (partly) meaningful collisions

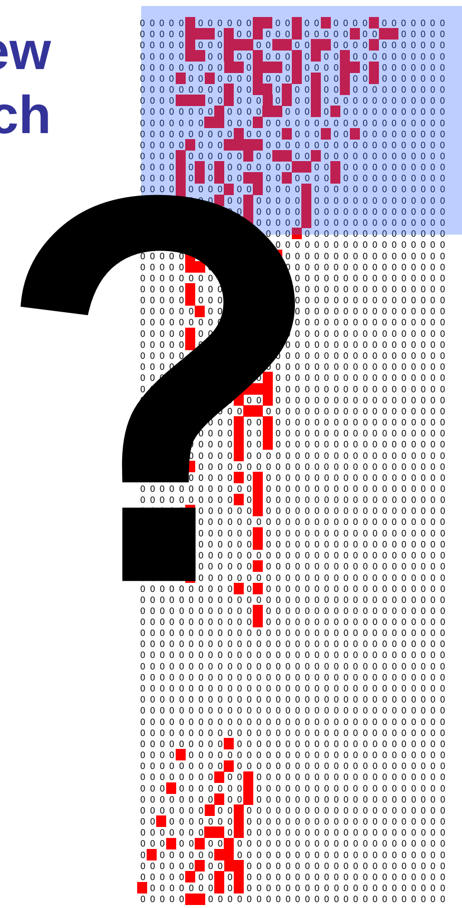
- Conclusions

Open Problem

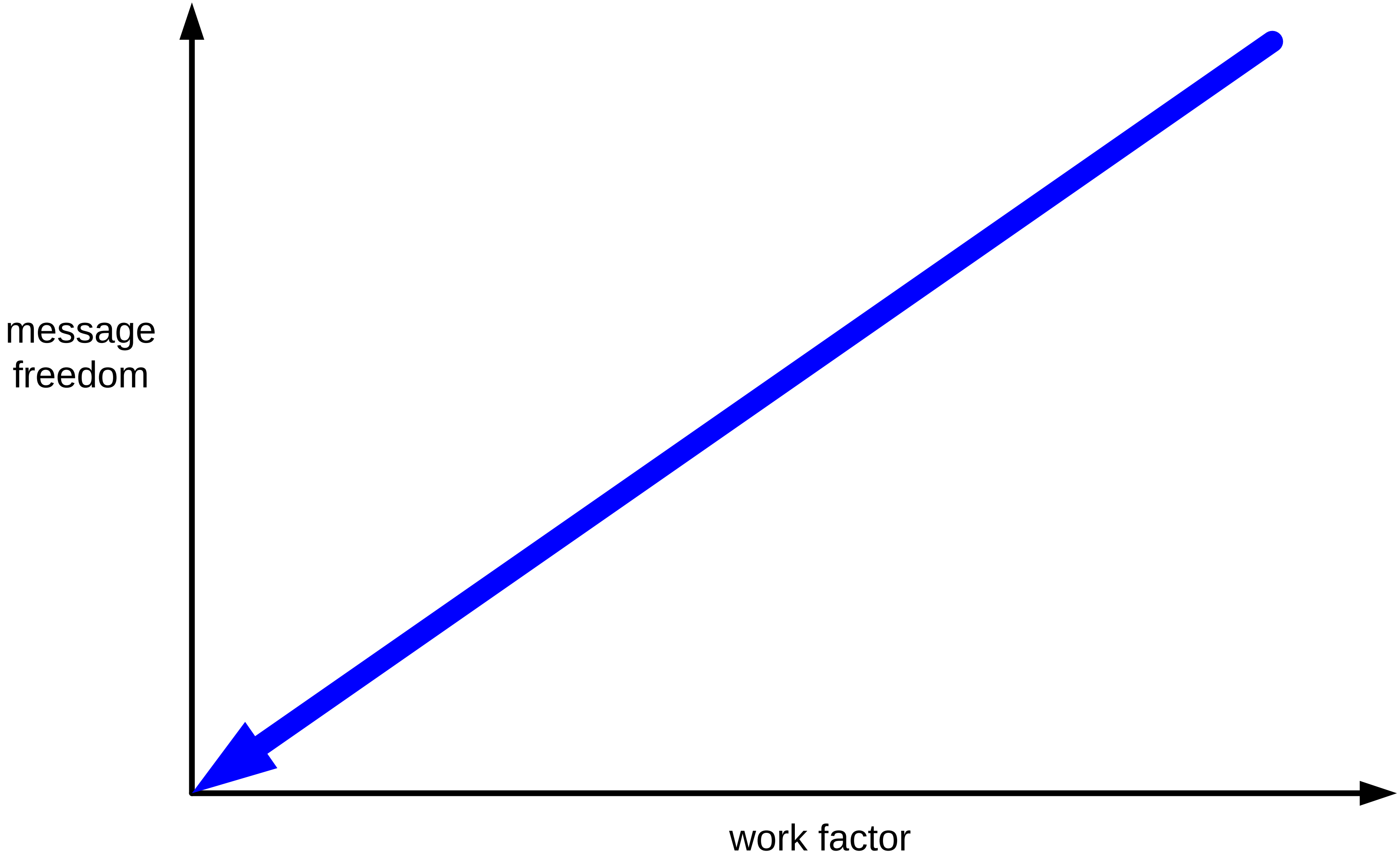
SHA-1 old approach

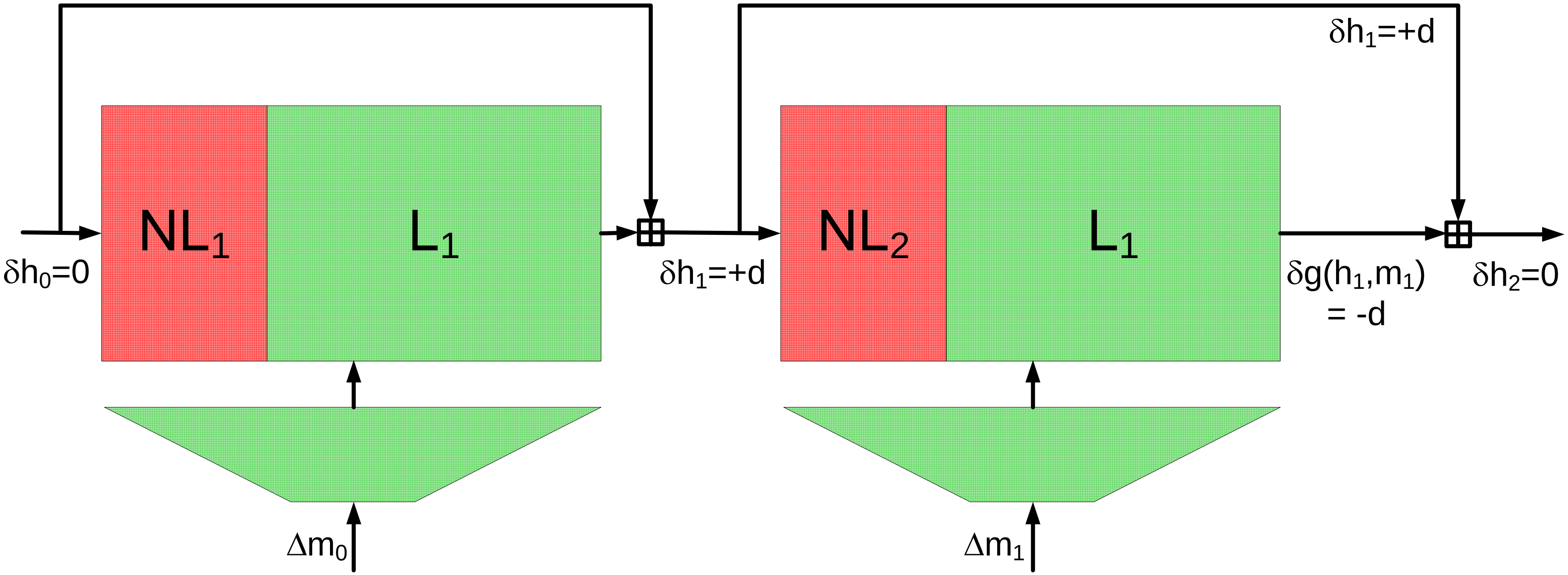


SHA-1 new approach



Finding Collisions as a Continuing Optimization Process

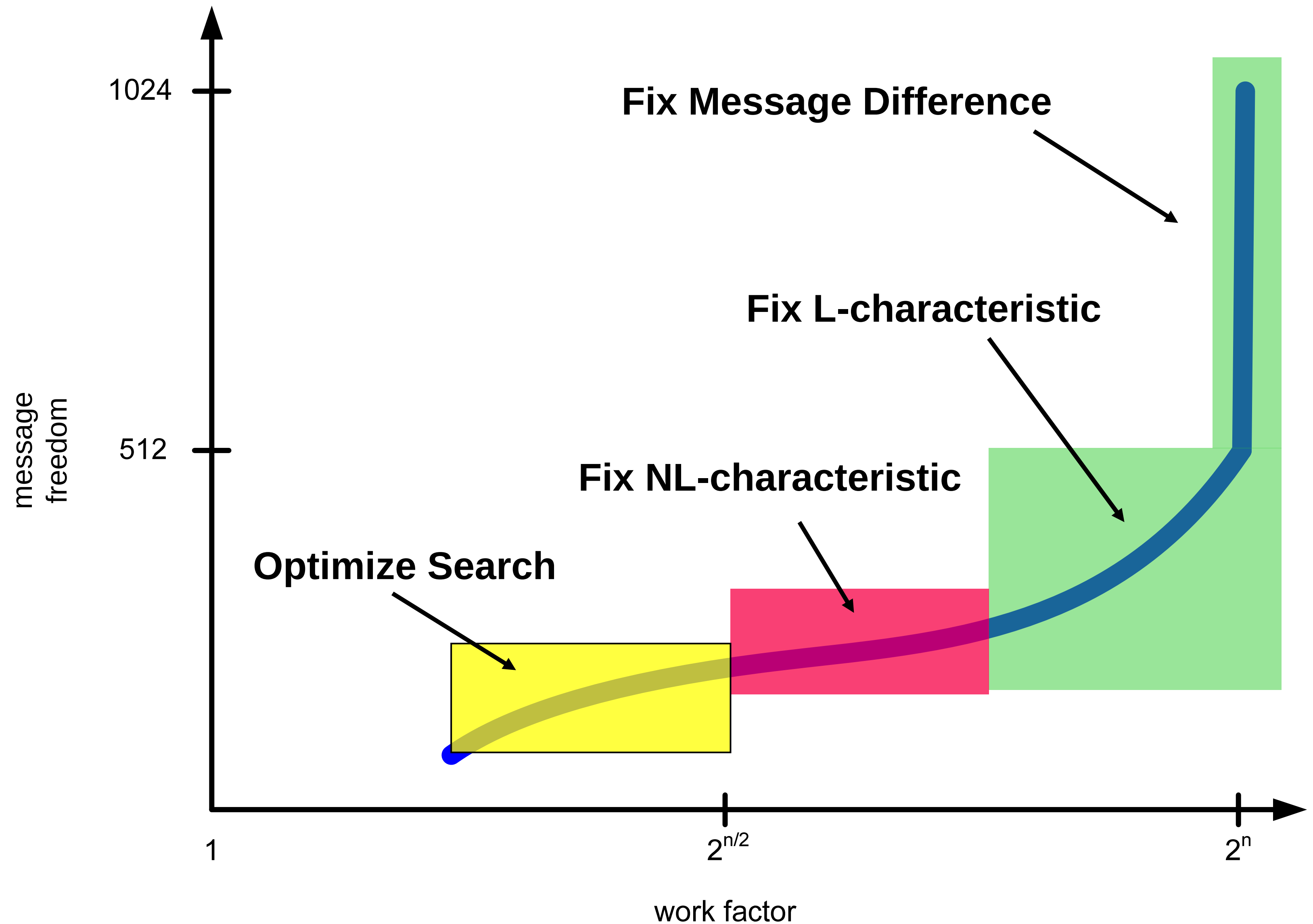




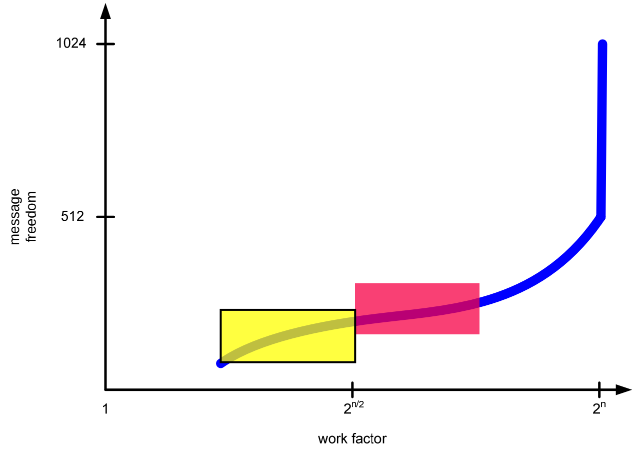
- Two key techniques of Wang et al.:
  - Manually find suitable complex characteristic NL_1 and NL_2
  - Advanced message modification to improve work factor

- Methods are rather ad hoc (manual)
- Optimization?

New View – Roughly Illustrated



Principles



- Generalized conditions

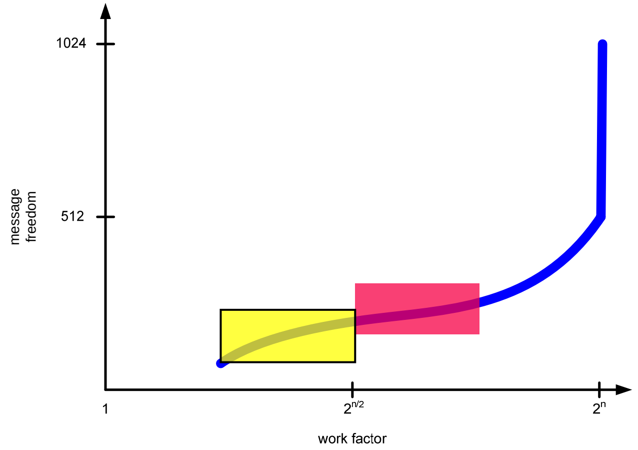
x_i	x_i^*
0	0
0	1
1	0
1	1

Type	**Possibilities**
XOR	2
Signed-bit	4-6
Generalized:	**16**

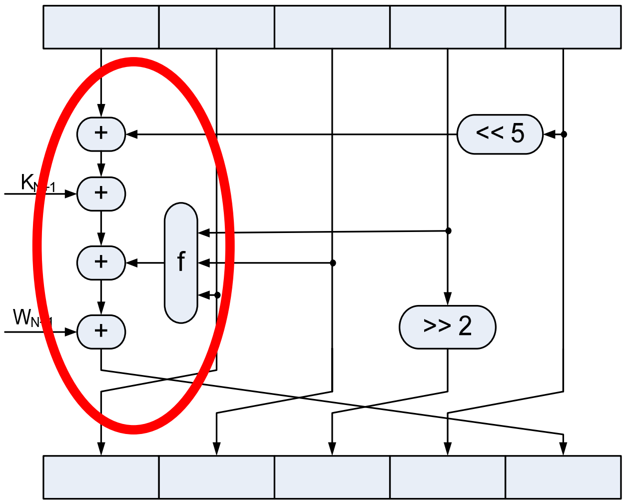
Generalized Conditions - Notation

(x_i, x_i^*)	$(0,0)$	$(1,0)$	$(0,1)$	$(1,1)$
?	✓	✓	✓	✓
-	✓	-	-	✓
x	-	✓	✓	-
0	✓	-	-	-
u	-	✓	-	-
n	-	-	✓	-
1	-	-	-	✓
#	-	-	-	-

Principles

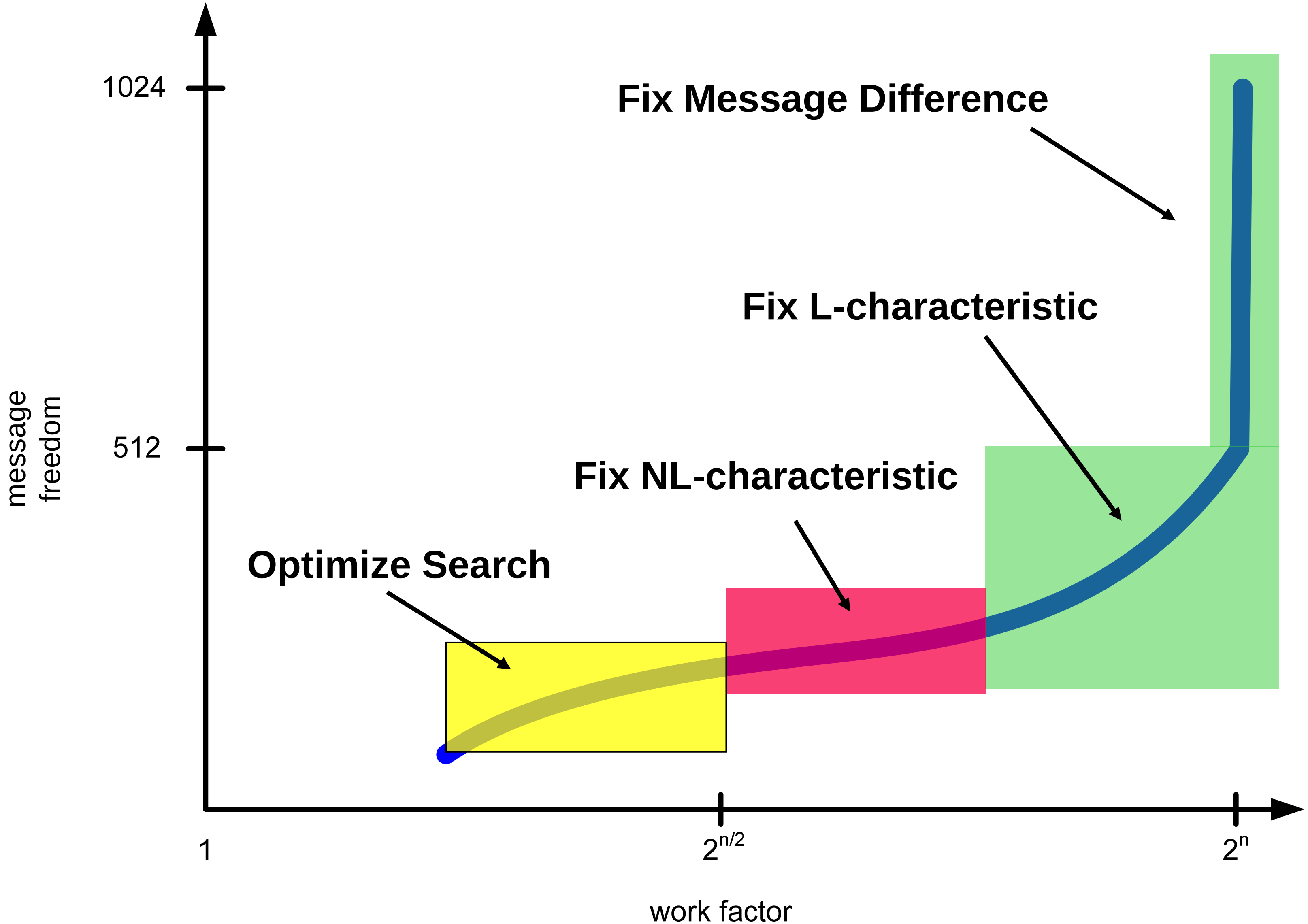


- Generalized conditions
- Use “bit-sliced design” to efficiently
  - Propagate conditions *within one* step transformation
  - Propagate conditions *among all* step transformations

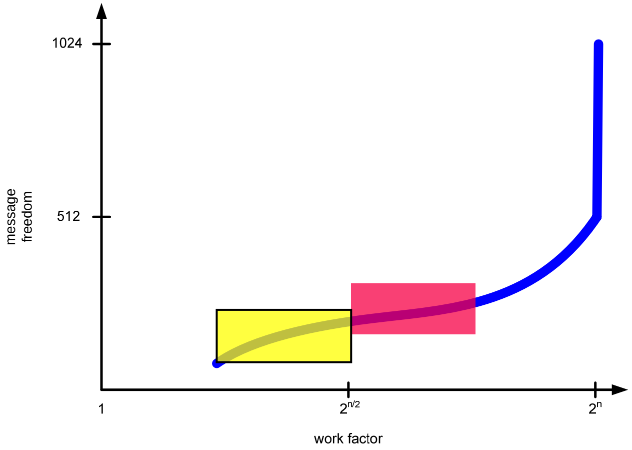


Animated Illustration

New View – Roughly Illustrated



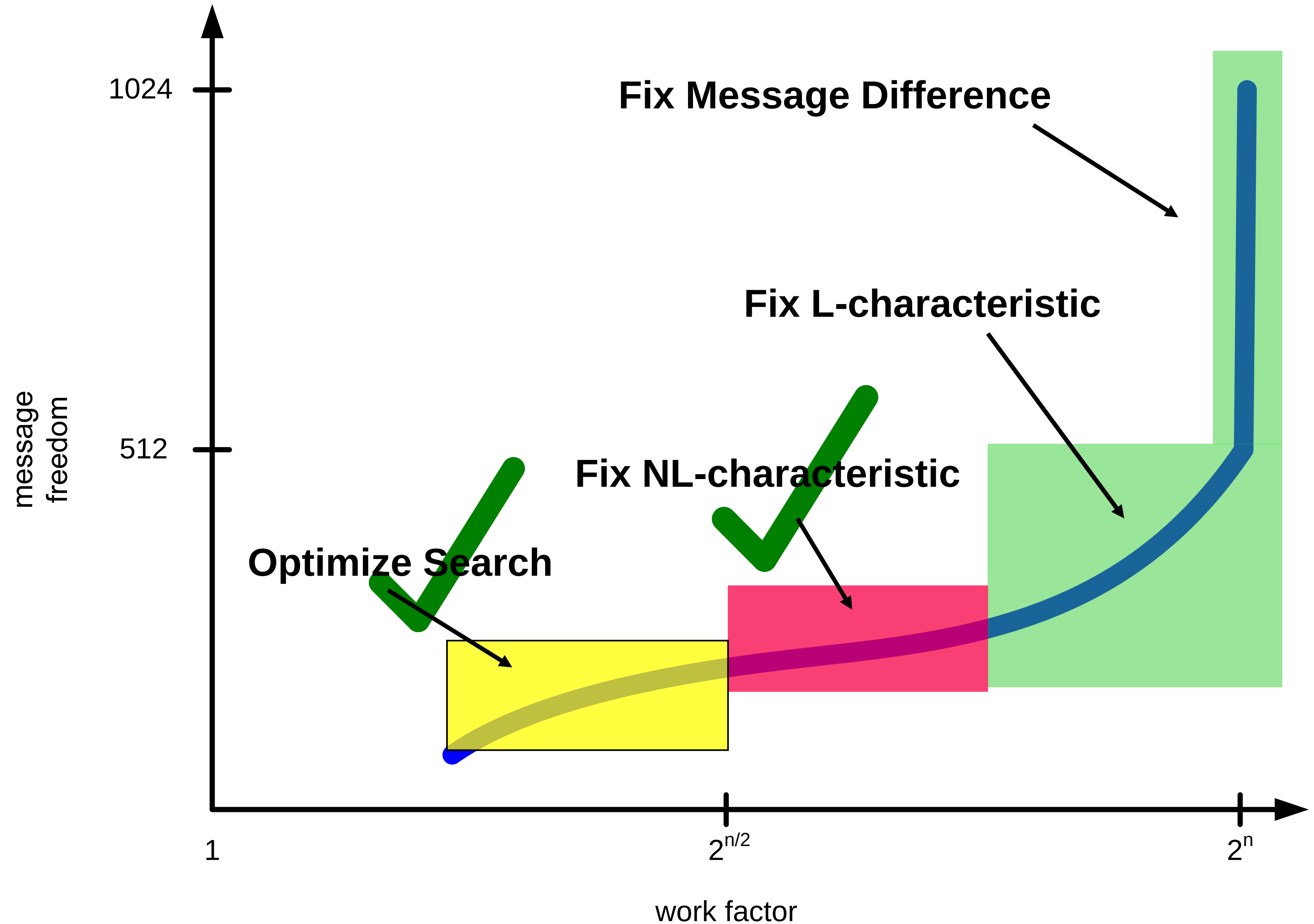
Principles



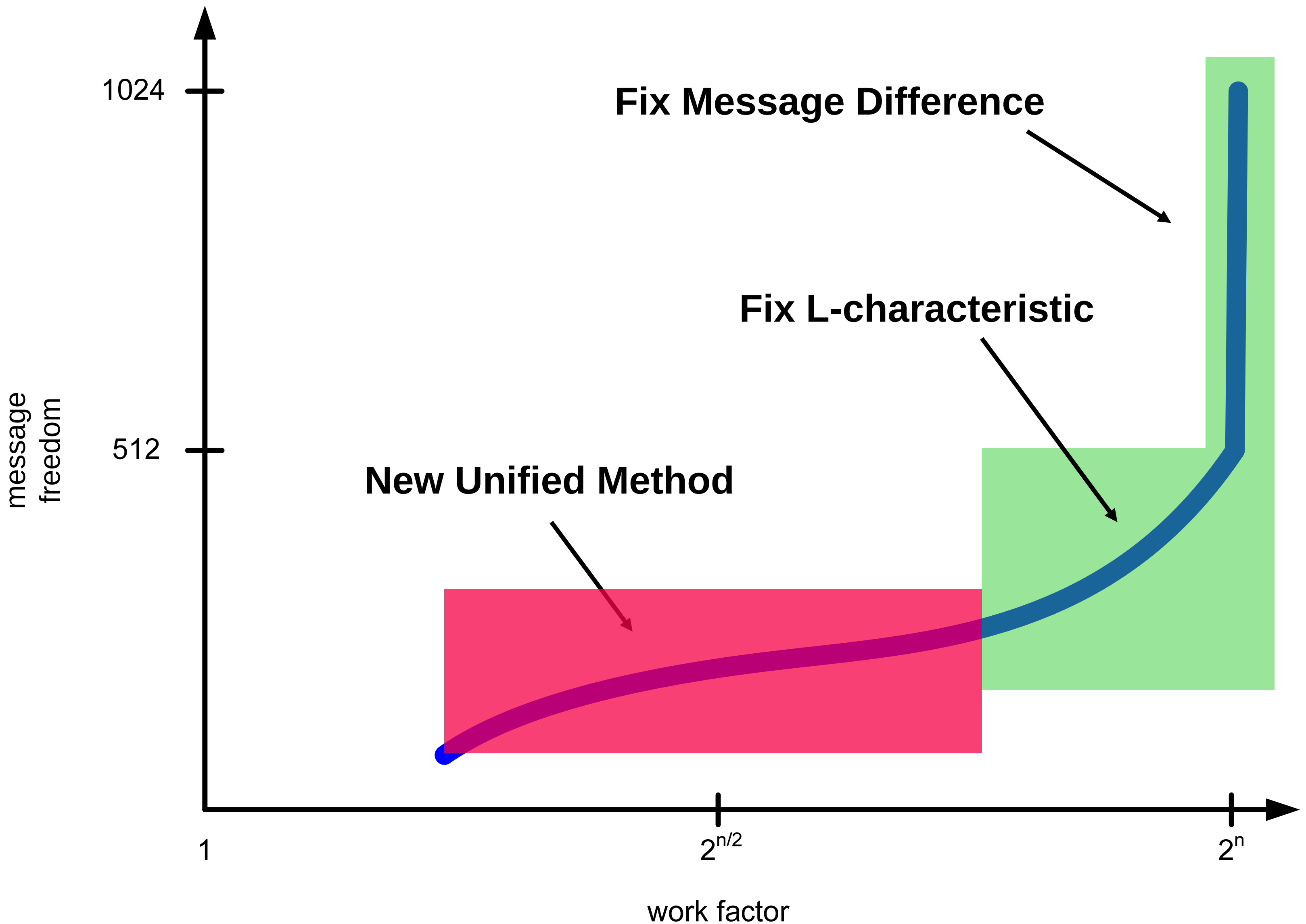
- Generalized conditions
- Use "bit-sliced design" to efficiently
  - Propagate conditions *within one* step transformation
  - Propagate conditions *among all* step transformations

- **Continuously add more conditions to improve work factor**

New View – Roughly Illustrated



New View – Roughly Illustrated



Example: 64-step SHA-1 collision

i	Message 1, first block			
1-4	63DAEFDD	30A0D167	52EDCDA4	90012F5F
5-8	0DB4DFB5	E5A3F9AB	AE66EE56	12A5663F
9-12	D0320F85	8505C67C	756336DA	DFFF4DB9
13-16	596D6A95	0855F129	429A41B3	ED5AE1CD

i	Message 1, second block			
1-4	3B2AB4E1	AAD112EF	669C9BAE	5DEA4D14
5-8	1DBE220E	AB46A5E0	96E2D937	F3E58B63
9-12	BE594F1C	BD63F044	50C42AA5	8B793546
13-16	A9B24128	816FD53A	D1B663DC	B615DD01

i	Message 2, first block			
1-4	63DAEFDE	70A0D135	12EDCDE4	70012F0D
5-8	ADB4DFB5	65A3F9EB	8E66EE57	32A5665F
9-12	50320F84	C505C63E	B5633699	9FFF4D9B
13-16	596D6A96	4855F16B	829A41F0	2D5AE1EF

i	Message 2, second block			
1-4	3B2AB4E2	EAD112BD	269C9BEE	BDEA4D46
5-8	BDBE220E	2B46A5A0	B6E2D936	D3E58B03
9-12	3E594F1D	FD63F006	90C42AE6	CB793564
13-16	A9B2412B	C16FD578	11B6639F	7615DD23

i	XOR-difference for both blocks			
1-4	00000003	40000052	40000040	E0000052
5-8	A0000000	80000040	20000001	20000060
9-12	80000001	40000042	C0000043	40000022
13-16	00000003	40000042	C0000043	C0000022

i	The colliding hash values			
1-4	A750337B	55FFFDBB	C08DB36C	0C6CFD97
5	A12EFFE0			

- 64-step 2-block colliding pair of messages
- Work factor (both blocks) was less than 2^{35} SHA-1 computations
(1st block much faster)
- Underlying method recently presented at NIST Hash Workshop and Asiacrypt 2006
(joint work with Christophe De Cannière)

Agenda

- Authentication using hash functions –
  Attacks on NMAC/HMAC-SHA-1

- New view on the problem of collision search in SHA-1
- New automated method – Results and Examples
- Extensions to (partly) meaningful collisions

- Conclusions

Motivation

- Setting: Collisions for a hash function can be constructed
- Cryptanalyst perspective: Some more interesting things to find out w.r.t. collision resistance?
  - Constructing collisions **faster**
  - Finding and exploiting degrees of freedom to construct (partially) **meaningful** collisions

  Practically relevant if hash function is widely deployed

Color Code

Under control, attacker can freely choose → meaningful

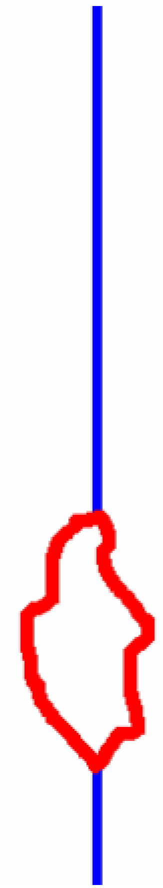
Not under direct control, determined by the collision search algorithm → not meaningful

Meaningful Collisions: Challenges for MD4-style Hash Functions

1. One Commonly Chosen Prefix
2. One Commonly Chosen Prefix + Partial Control over Colliding Blocks
3. Two Arbitrary Different Chosen Prefixes
4. Two Arbitrary Different Chosen Prefixes + Partial Control over Colliding Blocks

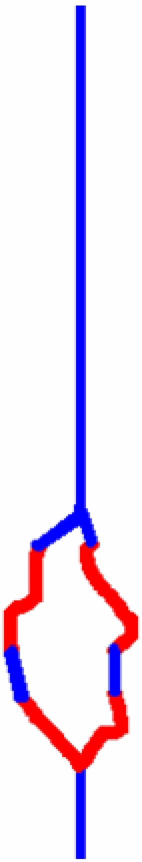
Meaningful Collisions: Challenges for MD4-style Hash Functions

1. **One Commonly Chosen Prefix**
2. One Commonly Chosen Prefix + Partial Control over Colliding Blocks
3. Two Arbitrary Different Chosen Prefixes
4. Two Arbitrary Different Chosen Prefixes + Partial Control over Colliding Blocks



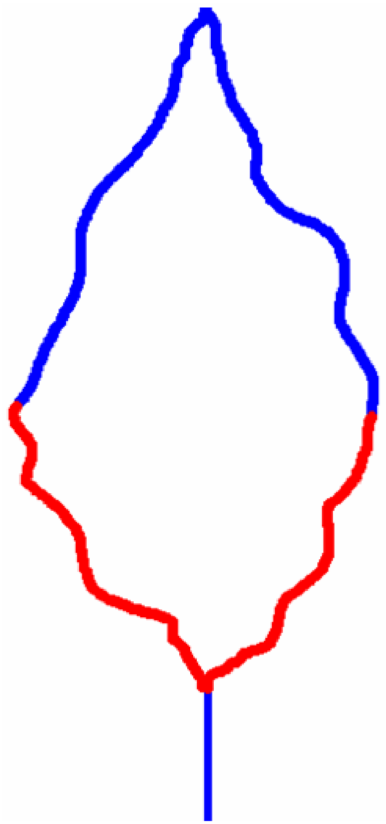
Meaningful Collisions: Challenges for MD4-style Hash Functions

1. One Commonly Chosen Prefix
2. **One Commonly Chosen Prefix + Partial Control over Colliding Blocks**
3. Two Arbitrary Different Chosen Prefixes
4. Two Arbitrary Different Chosen Prefixes + Partial Control over Colliding Blocks



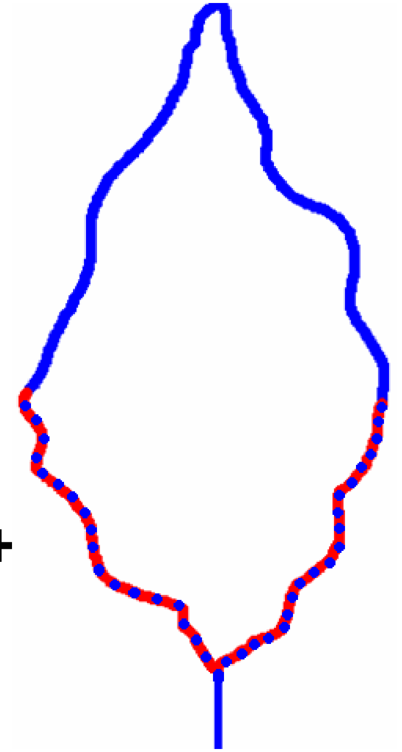
Meaningful Collisions:
Challenges for MD4-style Hash Functions

1. One Commonly Chosen Prefix
2. One Commonly Chosen Prefix +
   Partial Control over Colliding Blocks
3. **Two Arbitrary Different Chosen Prefixes**
4. Two Arbitrary Different Chosen Prefixes +
   Partial Control over Colliding Blocks



Meaningful Collisions: Challenges for MD4-style Hash Functions

1. One Commonly Chosen Prefix
2. One Commonly Chosen Prefix +
   Partial Control over Colliding Blocks
3. Two Arbitrary Different Chosen Prefixes
4. **Two Arbitrary Different Chosen Prefixes +**
   Partial Control over Colliding Blocks



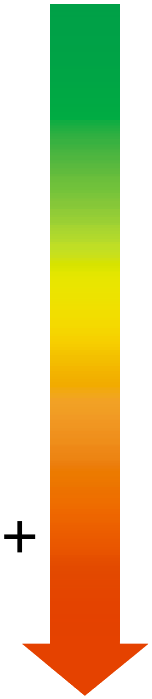
Meaningful Collisions:
Challenges for MD4-style Hash Functions

I. One Commonly Chosen Prefix

II. One Commonly Chosen Prefix +
Partial Control over Colliding Blocks

III. Two Arbitrary Different Chosen Prefixes

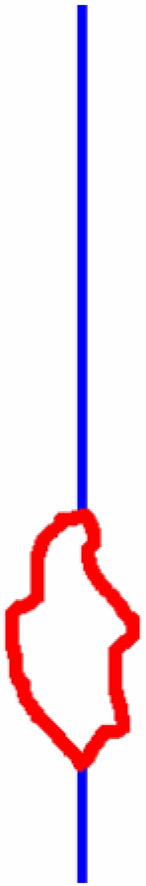
IV. Two Arbitrary Different Chosen Prefixes +
Partial Control over Colliding Blocks



“easier”

“harder”

I. One Commonly Chosen Prefix



- Small number of colliding blocks
- Enough for colliding meaningful postscript files, etc… (see tomorrow)

Example: Collision for 64-step SHA-1

Same Stuff

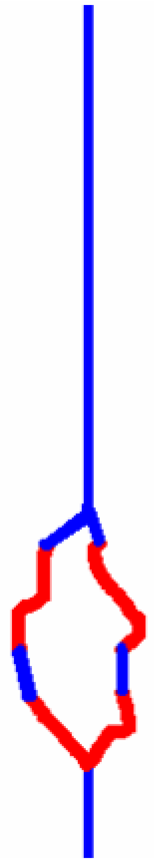
[random not meaningful garbage]

Same Hash

Same Stuff

[different random not meaningful Garbage]

II. One Commonly Chosen Prefix + Partial Control over Colliding Blocks



- Small number of colliding blocks
- Application in areas where format restrictions apply

Example: Collision for 64-step SHA-1

I hereby solemnly promise to finish my PhD thesis by the end of 2005

[Garbage]

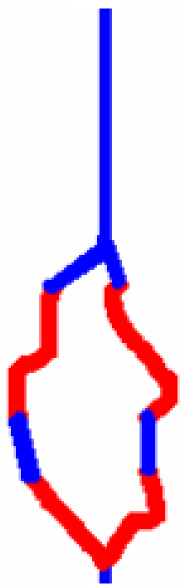
Same Hash

I hereby solemnly promise to finish my PhD thesis by the end of 2006'

[different Garbage]

Details (shown at Rump Session of Crypto 2006)

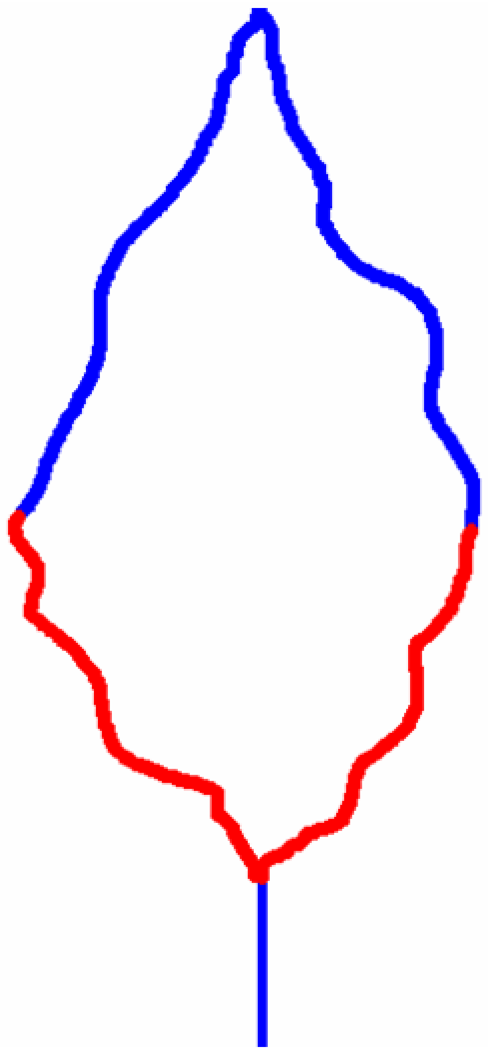
```
4920 6865 7265 6279 2073 6f6c 656d 6e6c  I hereby solemnl
7920 7072 6f6d 6973 6520 746f 2066 696e  y promise to fin
6973 6820 6d79 2050 6844 2074 6865 7369  ish my PhD thesi
7320 6279 2074 6865 2065 6e64 206f 6620  s by the end of
3230 3035 200a 0aea cbd7 029e 9f21 9821  2005 ........!.!
f0f0 ff92 13e4 3df4 07ca 4a69 0673 6850  ......=...Ji.shP
7f39 7c77 dddf 45c1 52ac 0ab0 9d15 11cf  .9|w..E.R.......
a15f dc78 9f4d 8621 5d1d 41f3 c2a7 3c6a  ._.x.M.!].A...<j
c2b5 d3a1 1ebb 7dee ffc2 7fb5 5c31 535c  ......}.....\1S\
8fb1 3dce c26a 4b89 0e82 d260 8ce7 31fb  ..=..jK....`..1.
383b 24d9 37fb eca9 f5e3 90b6 c123 15d5  8;$.7........#..
c1a4 8abe 9ad3 c1df f6d5 50c9 6bd9 572d  ..........P.k.W-
```



2nd (colliding) message:

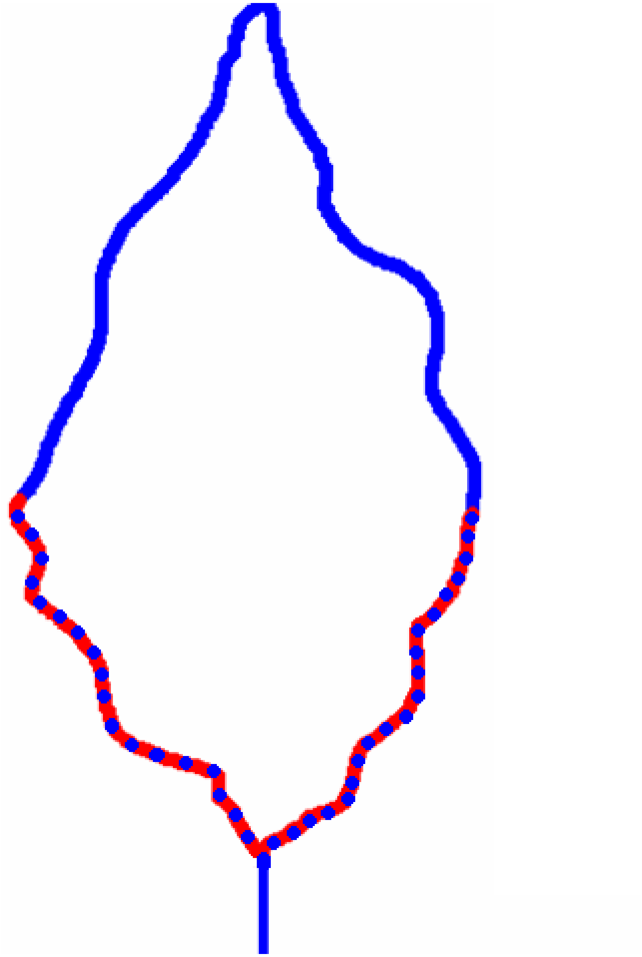
```
4920 6865 7265 6279 2073 6f6c 656d 6e6c  I hereby solemnl
7920 7072 6f6d 6973 6520 746f 2066 696e  y promise to fin
6973 6820 6d79 2050 6844 2074 6865 7369  ish my PhD thesi
7320 6279 2074 6865 2065 6e64 206f 6620  s by the end of
3230 3036 600a 0ab8 8bd7 02de 7f21 9873  2006`........!.s
50f0 ff92 93e4 3db4 27ca 4a68 2673 6830  P.....=.'.Jh&sh0
ff39 7c76 9ddf 4583 92ac 0af3 dd15 11ed  .9|v..E.........
a15f dc7b df4d 8663 9d1d 41b0 02a7 3c48  ._.{.M.c..A...<H
c2b5 d3a2 5ebb 7dbc bfc2 7ff5 bc31 530e  ....^.}......1S.
2fb1 3dce 426a 4bc9 2e82 d261 ace7 319b  /.=.BjK....a..1.
b83b 24d8 77fb eceb 35e3 90f5 8123 15f7  .;$.w...5....#..
c1a4 8abd dad3 c19d 36d5 508a abd9 570f  ........6.P...W.
```

III. Two Arbitrary Different Chosen Prefixes



- Using feed-forward operation, iteratively cancel out chaining differences with selected near-collision paths
- Usually much more than two message blocks needed
- Speedup: birthday phase before
- Example: see tomorrow

IV. Two Arbitrary Different Chosen Prefixes + Partial Control over Colliding Blocks



- Using feed-forward operation, iteratively cancel out chaining differences with selected near-collision paths
- Combination of methods

Example Characteristic for type IV

i	∇A_i	∇W_i	F_W	$P_u(i)$	$P_c(i)$
-4:	0000uuu1nu001n1u100nn111u1nn00u1				
-3:	01000n0n110nunnun1nu00n1u1n1un00				
-2:	0uu0nn10uu1nunu10111n01uu1u11n1n				
-1:	1u1n1111110n11n110101n1u1n001001				
0:	01u0nu1unu0n01010n10001un0n0n00u	n0n000000000000000000000000000nn	0	0.00	0.00
1:	u001uuu1u011n1nn1001un0nu0u10n1u	10n01000100011000000110110un0011	0	0.00	0.00
2:	uuu0nu0uuu0nu1uuu11u0001n0u0001u	0uu11111010100-----------------0	17	-14.00	0.00
3:	101nn110n1un00nn1uu-0un1uu0-10-1	nnn11010101011-10------1--u1n1u1	9	-8.00	0.00
4:	1011u01u00n11111n000u0-n0100011n	00u00101100110------100--n1001un	8	-7.00	0.00
5:	00n111uu101111nn1u0u10u0-1n00010	x1un010010-110--011--0101u-11-10	8	-3.00	0.00
6:	0nn01n0nn0-1uu--01n1-11u0--u0n0n	xu-n0-11------0-00-0-----x-u--uu	20	-14.61	-0.19
7:	n0-nu-0110n0--1101-0u10-00-011nu	xu1u010-------1-------1---x---u0	22	-19.00	-0.68
8:	00n10001n0u10u101--u0n01u1n--0u1	-1n0------0----1---------1-0----	25	-13.00	0.00
9:	-11111n---100n-10----0n0u0001---	-nn-------0---------------uu--u-	26	-17.00	-2.00
10:	0--n1-1-0-010n-0--u--1-01u1n0---	-nu----------------------n----uu	27	-15.00	-1.00
11:	1---110-0--11n------1---0-01nu0-	--n--------------------1-n------	29	-18.00	-0.39
12:	--nu-1n-n-n--uun--n-1--nu-u1u010	xnu----------------------u----u-	28	-13.00	-1.00
13:	u---01--0-0-0--100--1-------0-10	-nn---------------------------n-	29	-17.00	-4.74
14:	x-0-11--1-1-1--011--1-----0-1-1-	x------------------------------u	31	-2.00	-2.00
15:	--------------------------------	--------------------------x-----	32	-2.00	0.00

Agenda

- Authentication using hash functions –
  Attacks on NMAC/HMAC-SHA-1

- New view on the problem of collision search in SHA-1
- New automated method – Results and Examples
- Extensions to (partly) meaningful collisions

- Conclusions

Conclusions / Future Work

- Collision for full (80-step) SHA-1 is getting closer
- Optimization is ongoing
  - 2005: 2^69
  - 2006: 2^62 – 2^63
  - Advanced techniques as used for partial meaningful collisions can also be turned into **faster** collision search
  - 2007: ?

- Apply to other hash functions like RIPEMD-160, SHA-2?
- More powerful attacks on NMAC/HMAC?

Most Recent Results on SHA-1

Q&A

Christian.Rechberger@iaik.tugraz.at

www.iaik.tugraz.at/aboutus/people/rechberger

Hash&Stream, Salzburg, 2007/02/01

Institute for Applied Information Processing and Communications (IAIK) - <u>Krypto Group</u>

Faculty of Computer Science
Graz University of Technology