New Techniques for Cryptanalysis of Cryptographic Hash Functions

Rafi Chen

Department of Computer Science, Technion – Israel Institute of Technology

Joint work with Eli Biham

Talk Outline

- Definition and properties
- Applications
- Hash functions from the 90's till today
- Merkle-Damgård construction and its weaknesses
- Differential cryptanalysis of hash functions.
- The multi-block technique.
- The neutral-bits technique.
- Results.

A Cryptographic Hash Function

- A Cryptographic hash function H takes a message of arbitrary length and generates a short fingerprint.

$$H : \{0,1\}^* \mapsto \{0,1\}^m$$

A Cryptographic Hash Function

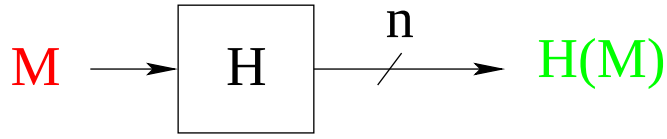
- A Cryptographic hash function H takes a message of arbitrary length and generates a short fingerprint.

$$H : \{0,1\}^* \mapsto \{0,1\}^m$$

- H has no secret key or hidden data. Cryptographic applications that use it rely on its properties.

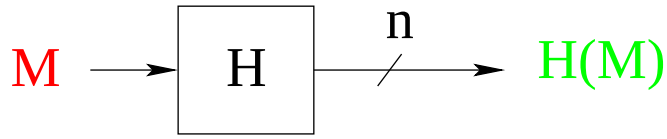
Required Properties

- Preimage resistance (2^n):

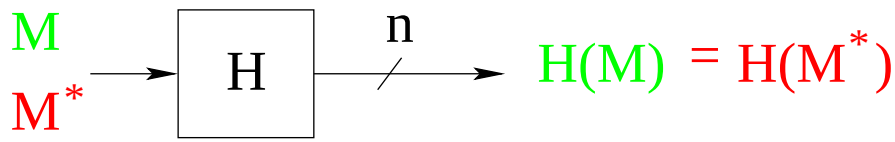


Required Properties

- Preimage resistance (2^n):

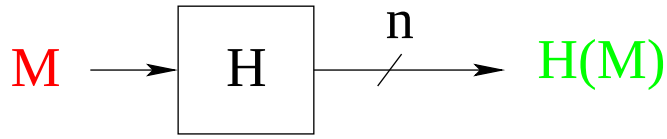


- 2nd Preimage resistance (2^n):

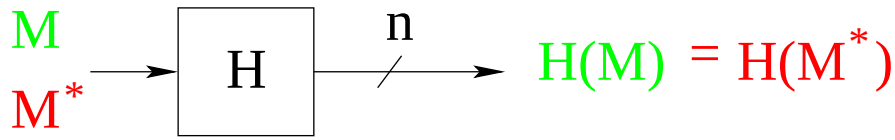


Required Properties

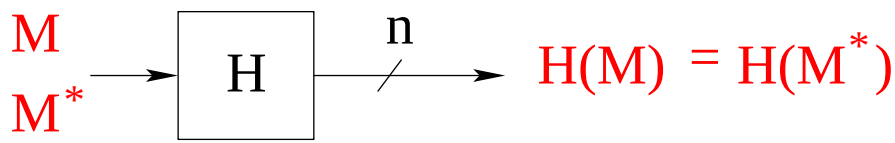
- Preimage resistance (2^n):



- 2nd Preimage resistance (2^n):

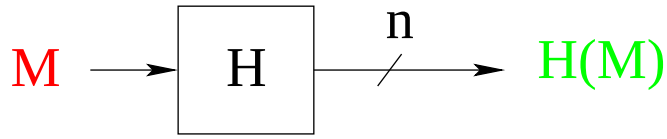


- Collision-resistance ($2^{n/2}$):

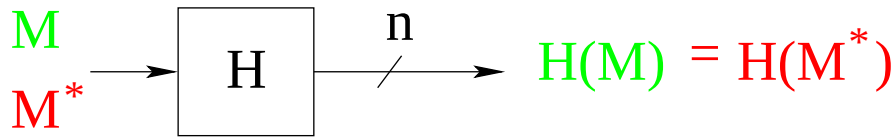


Required Properties

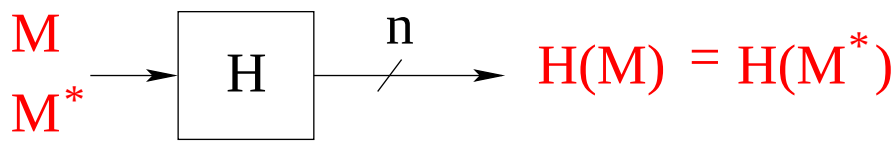
- Preimage resistance (2^n):



- 2nd Preimage resistance (2^n):



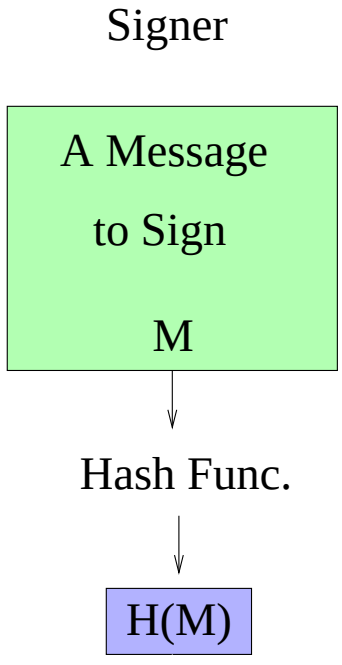
- Collision-resistance ($2^{n/2}$):



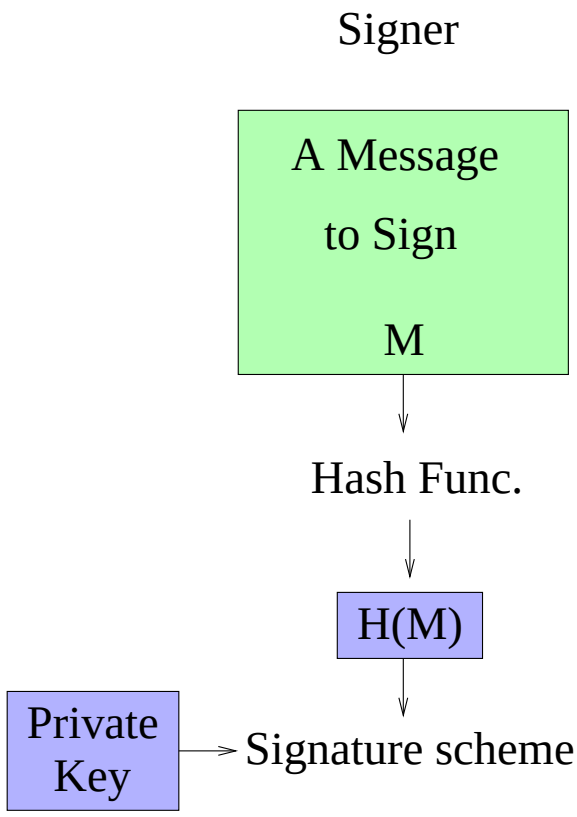
- Easy to compute

Applications - Digital Signature

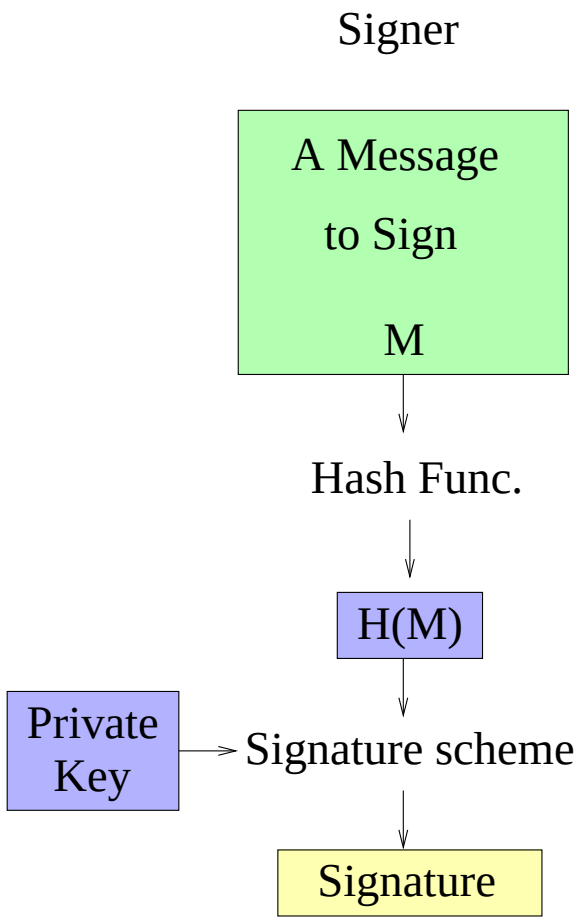
Applications - Digital Signature



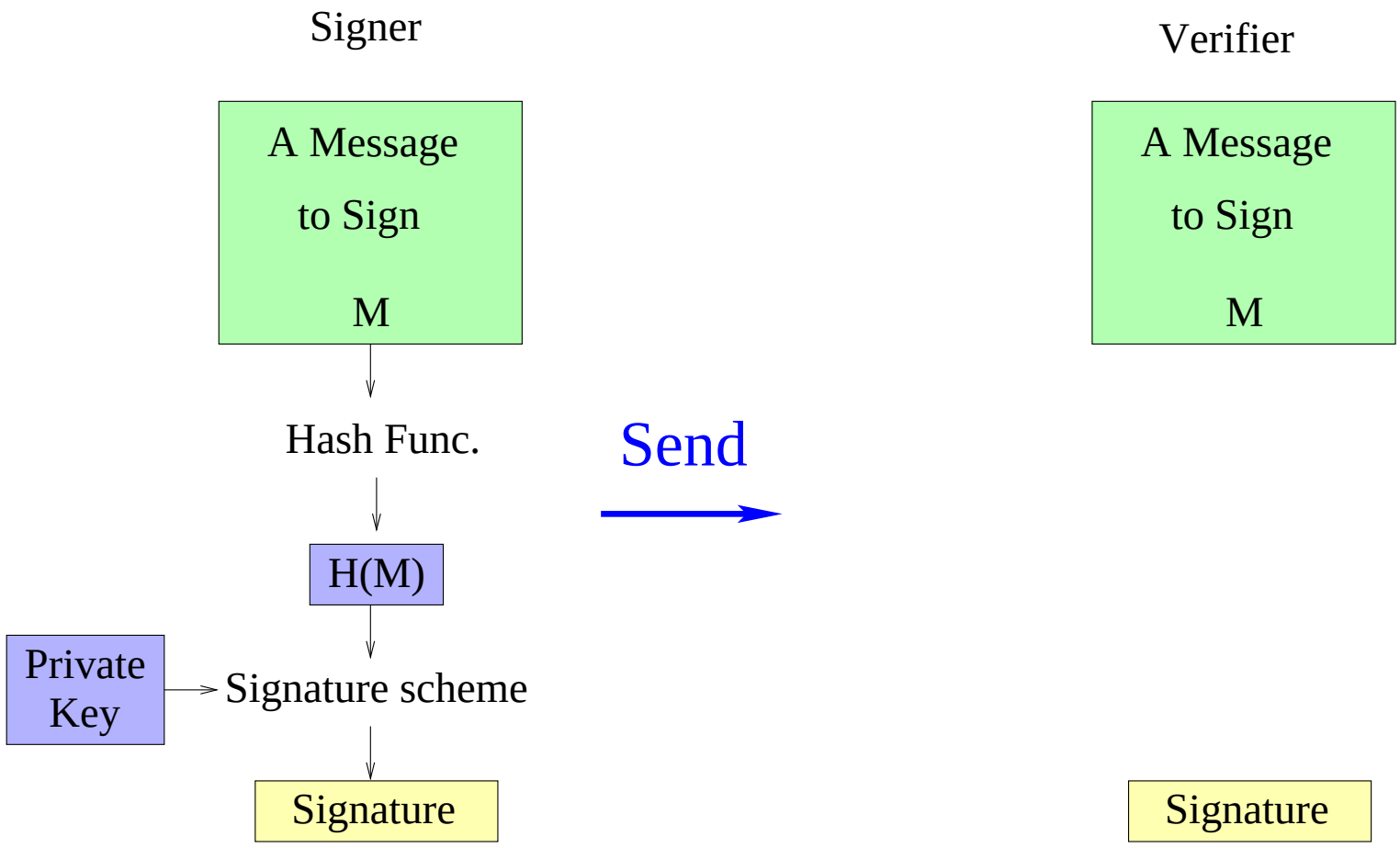
Applications - Digital Signature



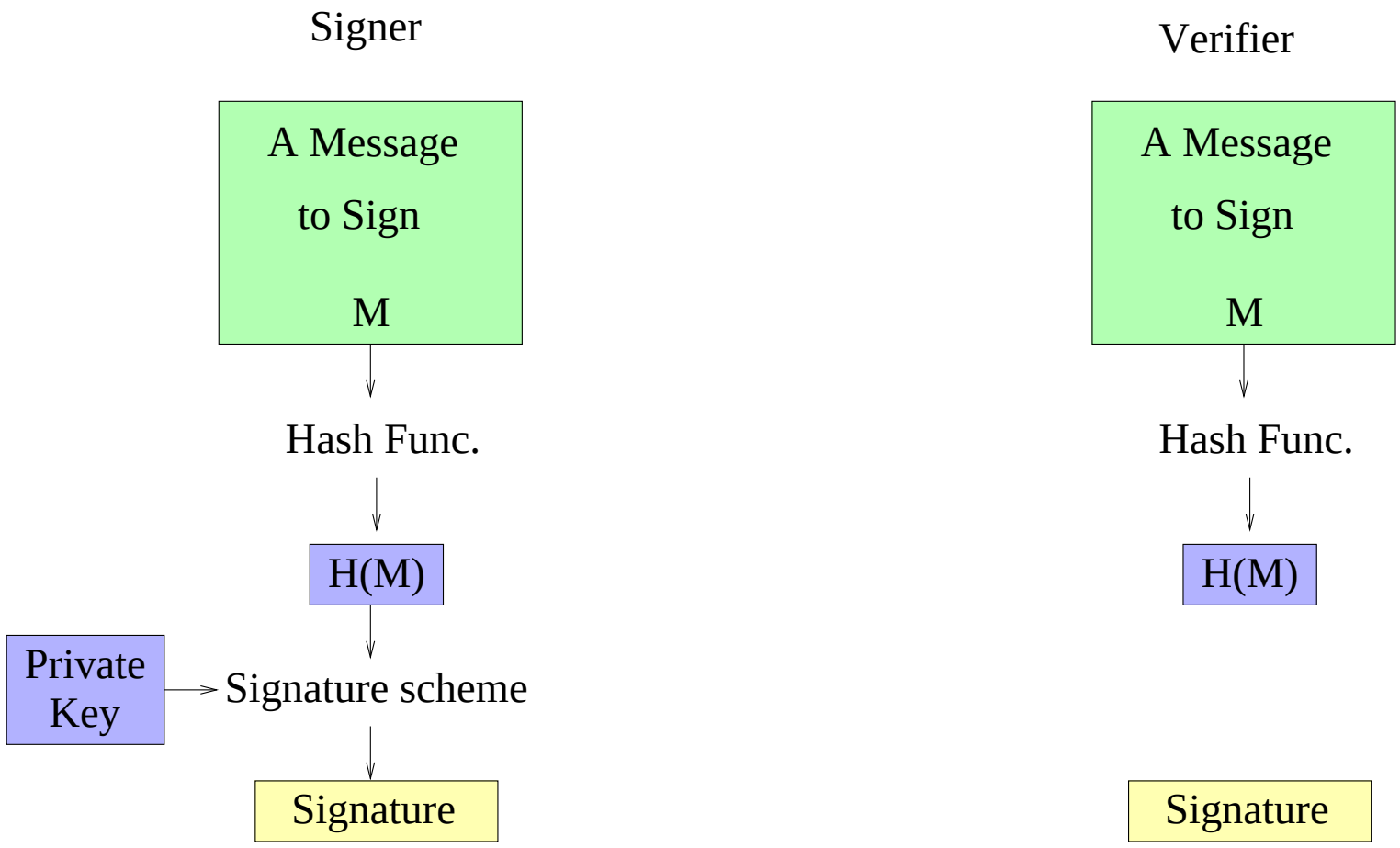
Applications - Digital Signature



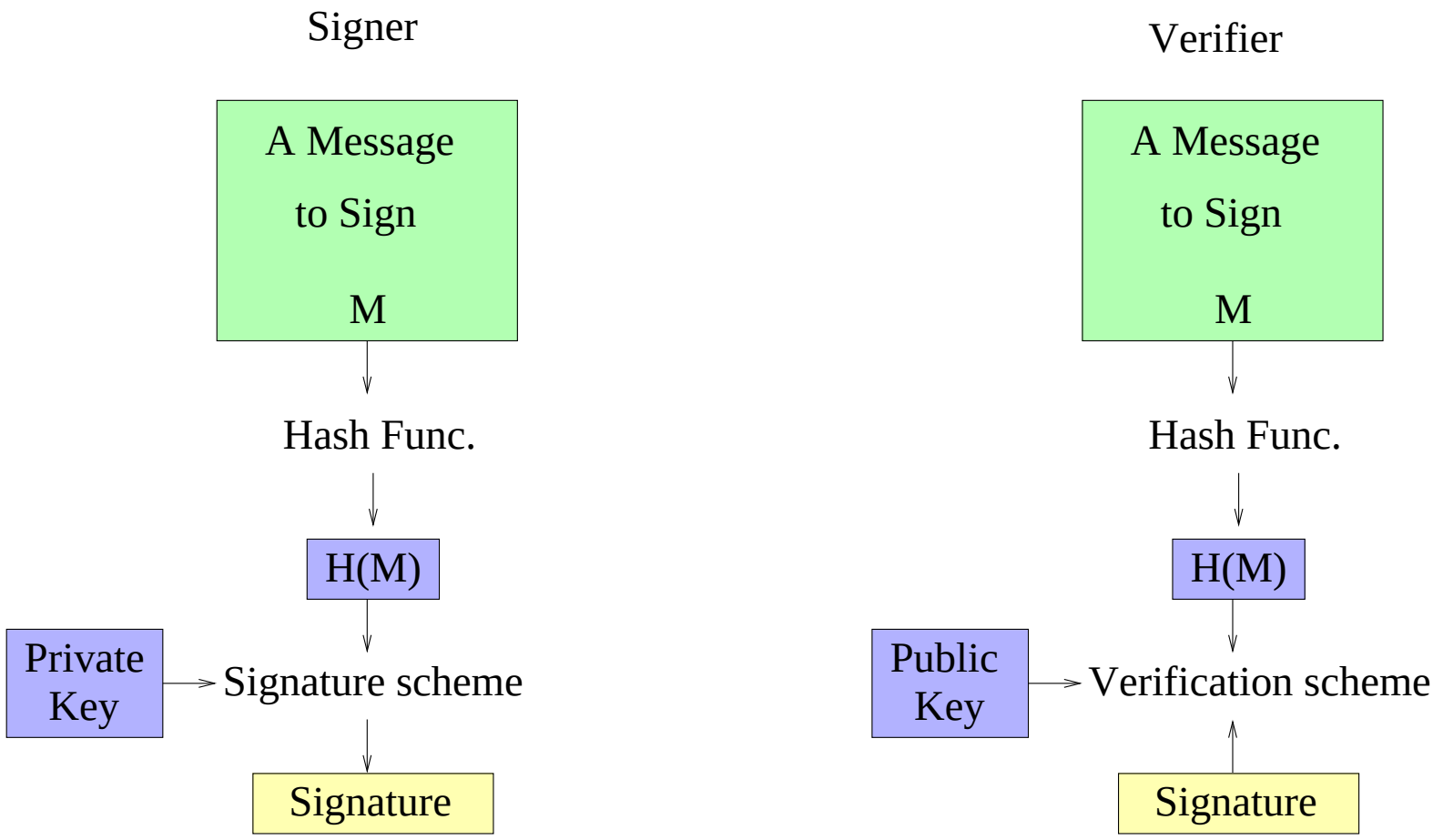
Applications - Digital Signature



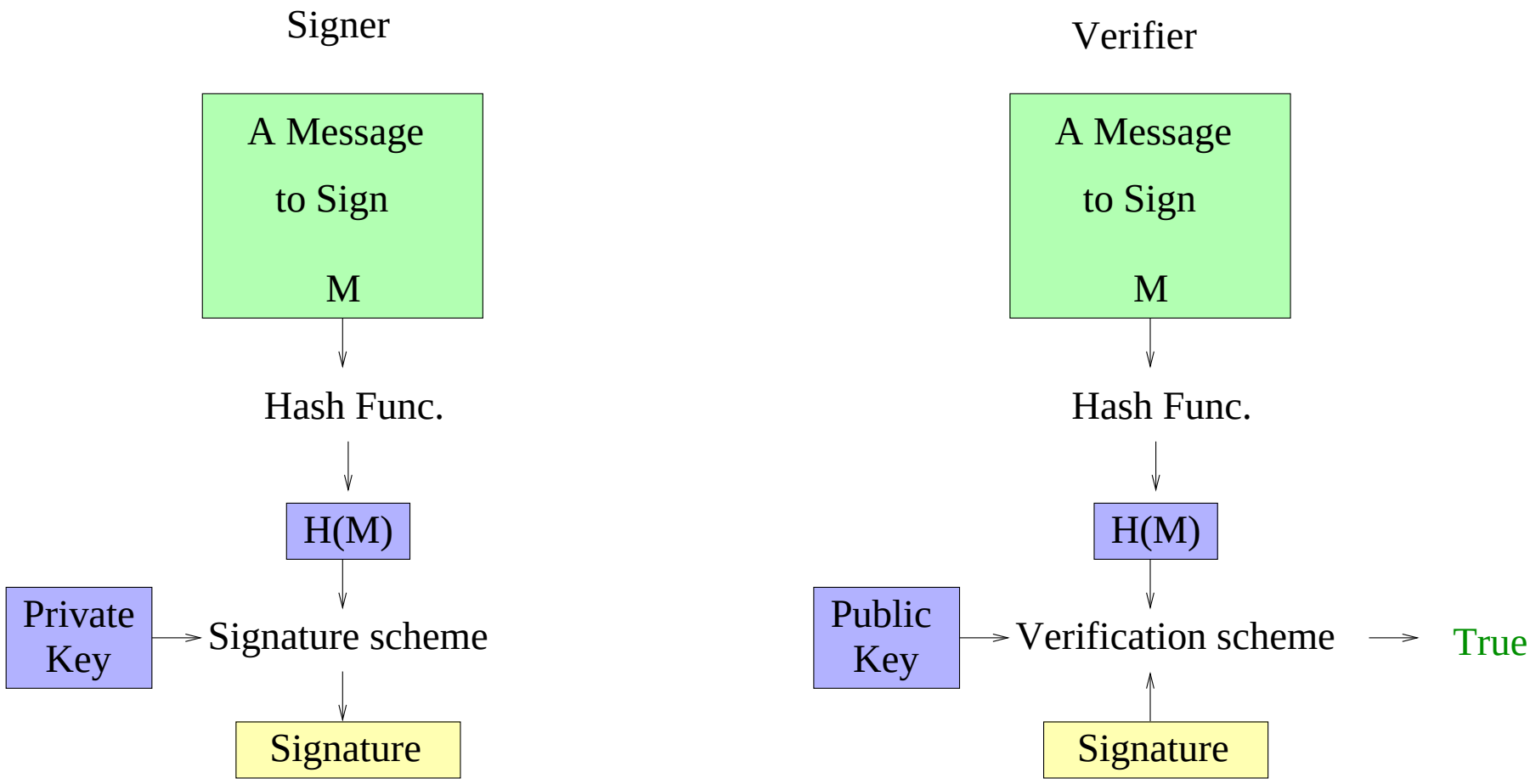
Applications - Digital Signature



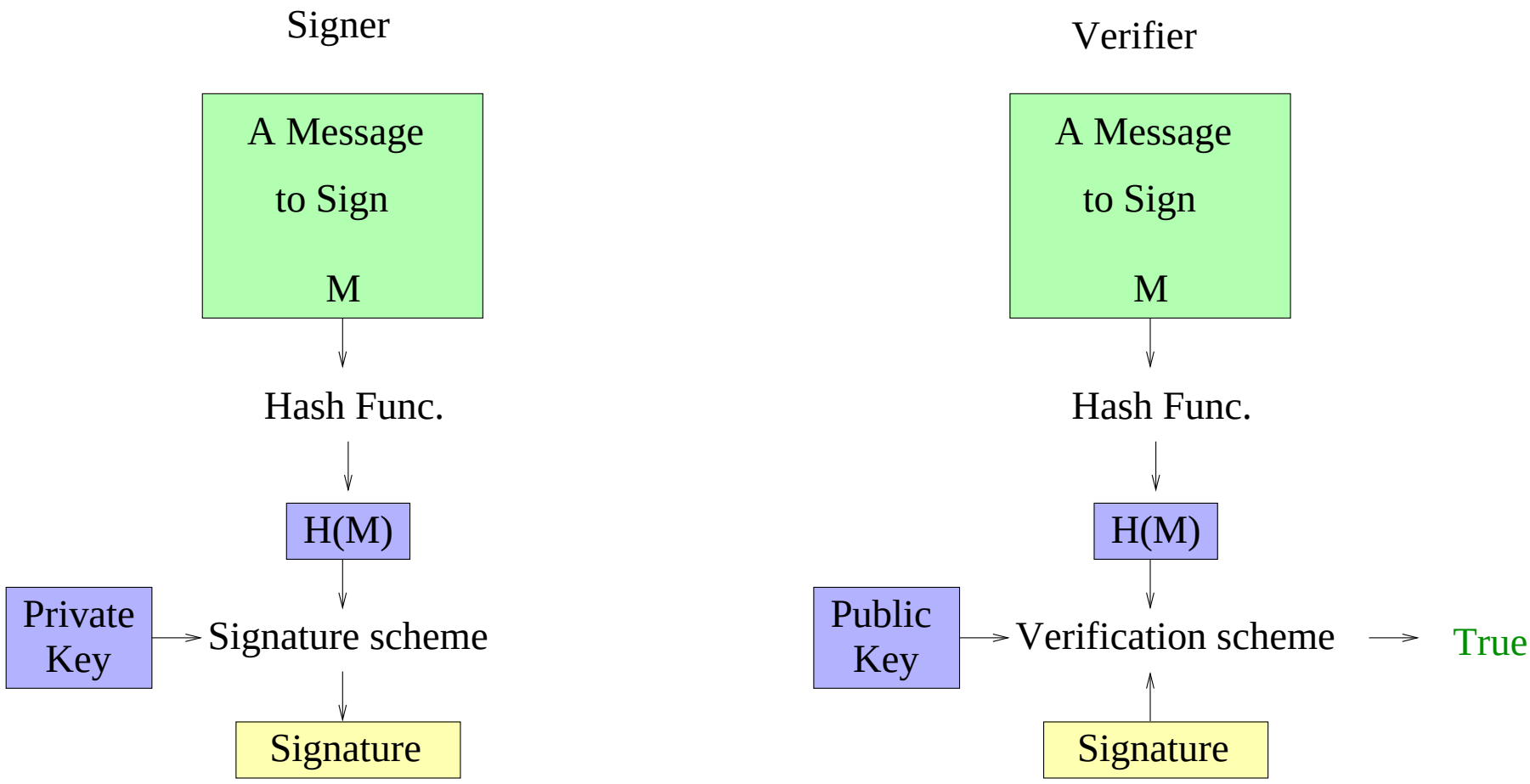
Applications - Digital Signature



Applications - Digital Signature



Applications - Digital Signature



If $H(M) = H(M^*)$ then M and M^* have the same signature.

Applications

- Message Integrity:
  - Instead of protecting the whole data, protect the hash of the data.
  - Second preimage resistance is required.

Applications

- Message Integrity:
  - Instead of protecting the whole data, protect the hash of the data.
  - Second preimage resistance is required.
- Password protection.
  - A password file holds:
    (User name, $salt, H(password||salt)$).
  - Passwords are protected in case an attacker accesses the password file.
  - Preimage resistance is required.

Applications

- Commitment
  - A who commit to M sends $H(M||salt)$ to B.
    - At the time A reveals his commitment he publishes M and the $salt$. B verifies the commitment by hashing and comparing.
    - Collision resistance, preimage resistance and second preimage resistance are required.

Applications

- Message Authentication Code - MAC.
  - Preimage resistance is required.

Applications

- Message Authentication Code - MAC.
  - Preimage resistance is required.
- and there are many more...

Hash Functions from the 90's till Today

1990-2000 (partial list)

- The hash functions use Merkle-Damgård construction.
- Hash size 128-192 bits.
- Optimized for 32-bit machines (except for Tiger).

Function	Dig. size	Designed	Broken	Complexity
Snefru	128-224	1990	1990	$2^{12.5} - 2^{56.5}$
MD4	128	1990	1995,2004	$2^{20}, 2^{8}$
MD5	128	1992	2004,2008	$2^{39}, 2^{16}$
SHA-0	160	1993	1998,2004	$2^{61}, 2^{51}, 2^{39}$
SHA-1	160	1995	2005,2011	$2^{63}, 2^{58}$
Tiger	≤ 192	1995		
RIPEMD-160	160	1996		

2000-2003

- Whirlpool, Nessie(2000) and SHA-2, NIST (2002)
- The hash functions still use Merkle-Damgård construction.
- Whirlpool is based on the Square block cipher.
- SHA-224, SHA-256, SHA-384, SHA-512 are based on the MD/SHA concept with more complex operations.
- Hash size 224-512.
- No real motivation to upgrade till the first attacks on SHA-1 in 2005.

SHA-3 Competition (2007)

- The break of SHA-1 motivated NIST to establish a public competition to choose the next generation of hash functions.
- 64 proposals were submitted.
- 51 passed Round 1, 14 passed Round 2, five passed Round 3, and the final decision will be given in 2012.

Recommendations

- Do not use broken hash functions, not SHA-1 and certainly not MD5.
- Midterm solution - Upgrade to Whirlpool or SHA-2.
- Upgrade to SHA-3 when it is available.

Merkle-Damgård Construction and Its Weaknesses

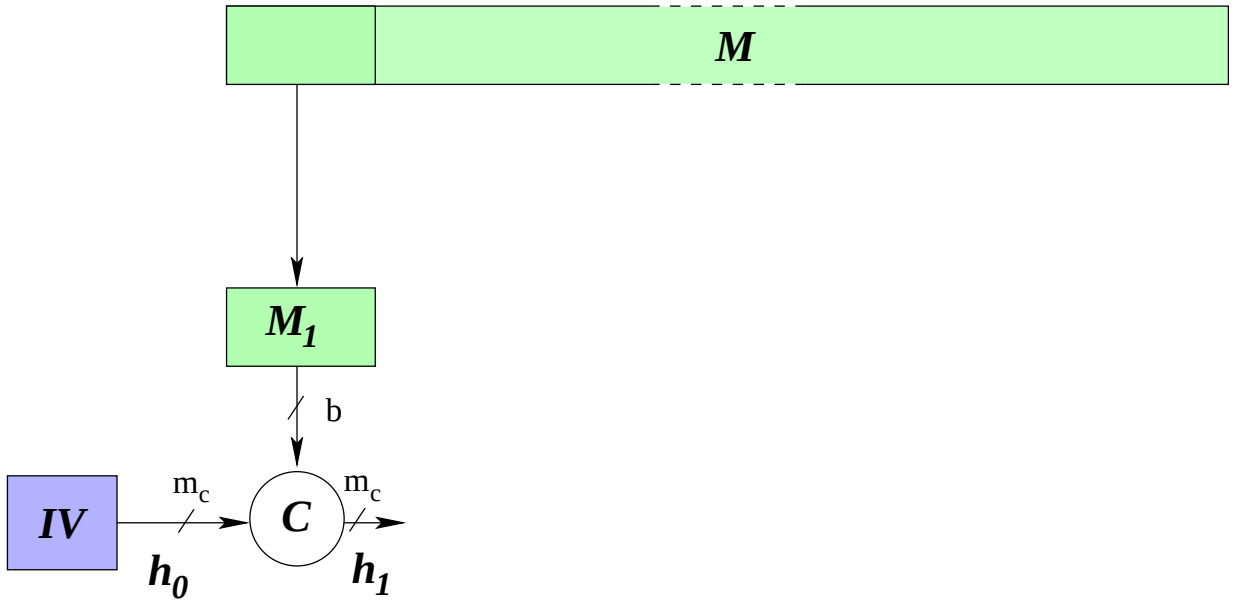
Merkle-Damgård Construction (1989)

- The hash function iterates a compression function C

$$C : \{0,1\}^{m_c+b} \mapsto \{0,1\}^{m_c},$$

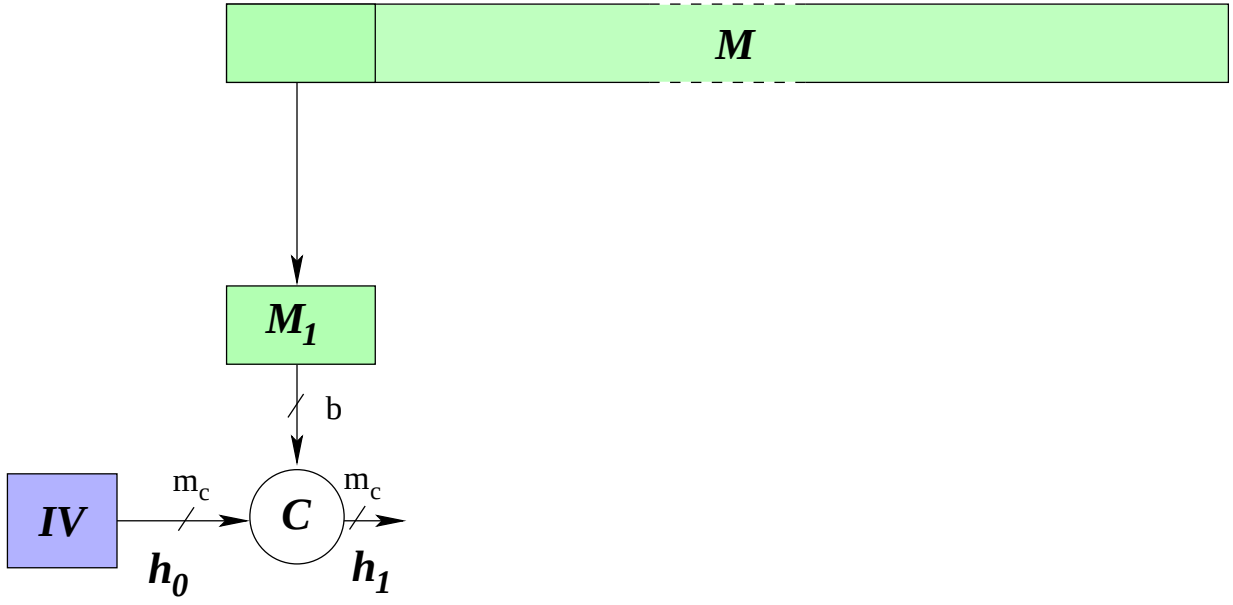
on a chaining value h_{k-1} and a message block M_k.

Merkle-Damgård Construction (1989)



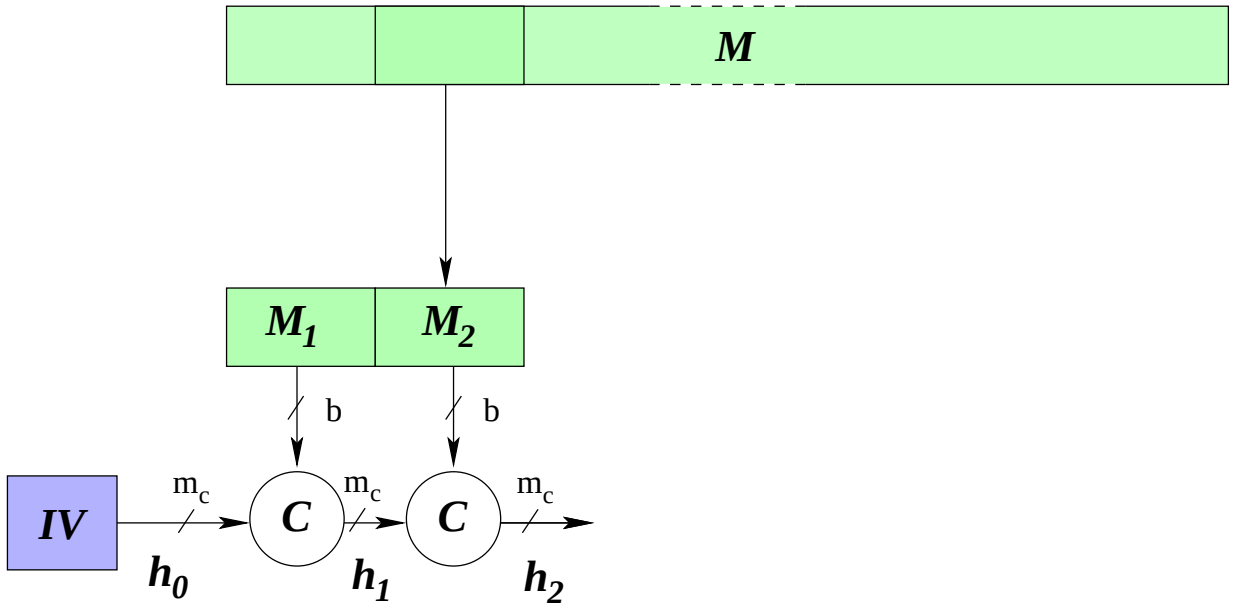
- The first chaining value is initialized to $h_0 = IV$.
- For each M_k and h_{k-1} compute: $h_k = \mathsf{C}(M_k, h_{k-1})$.

Merkle-Damgård Construction (1989)



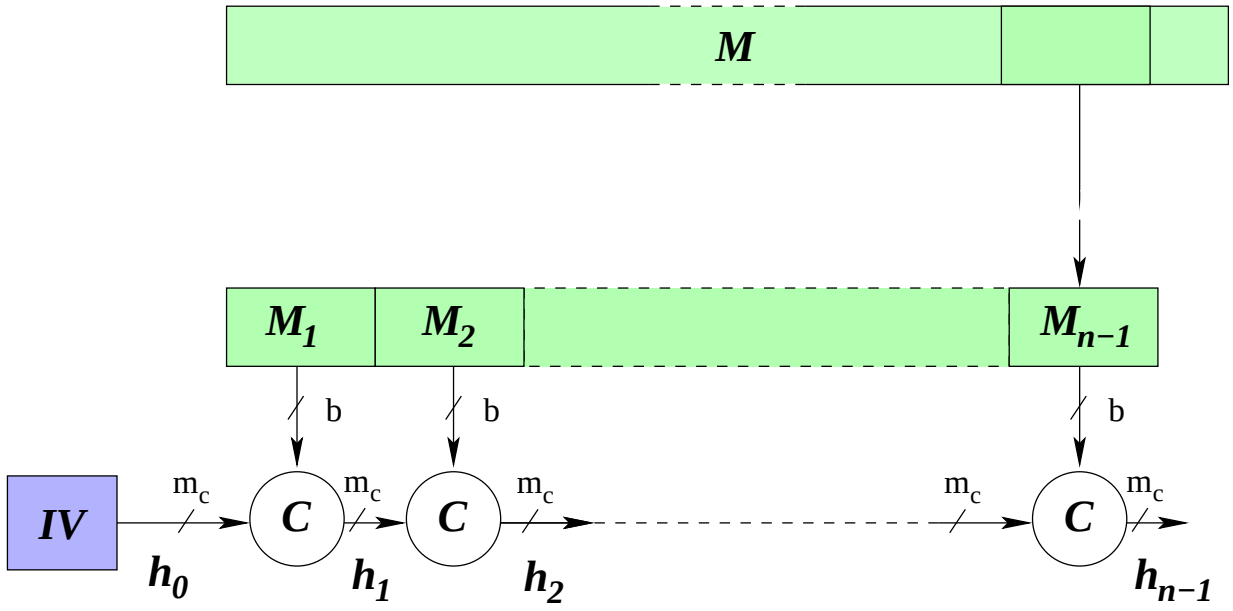
$$h_1 = \mathsf{C}(M_1, h_0)$$

Merkle-Damgård Construction (1989)



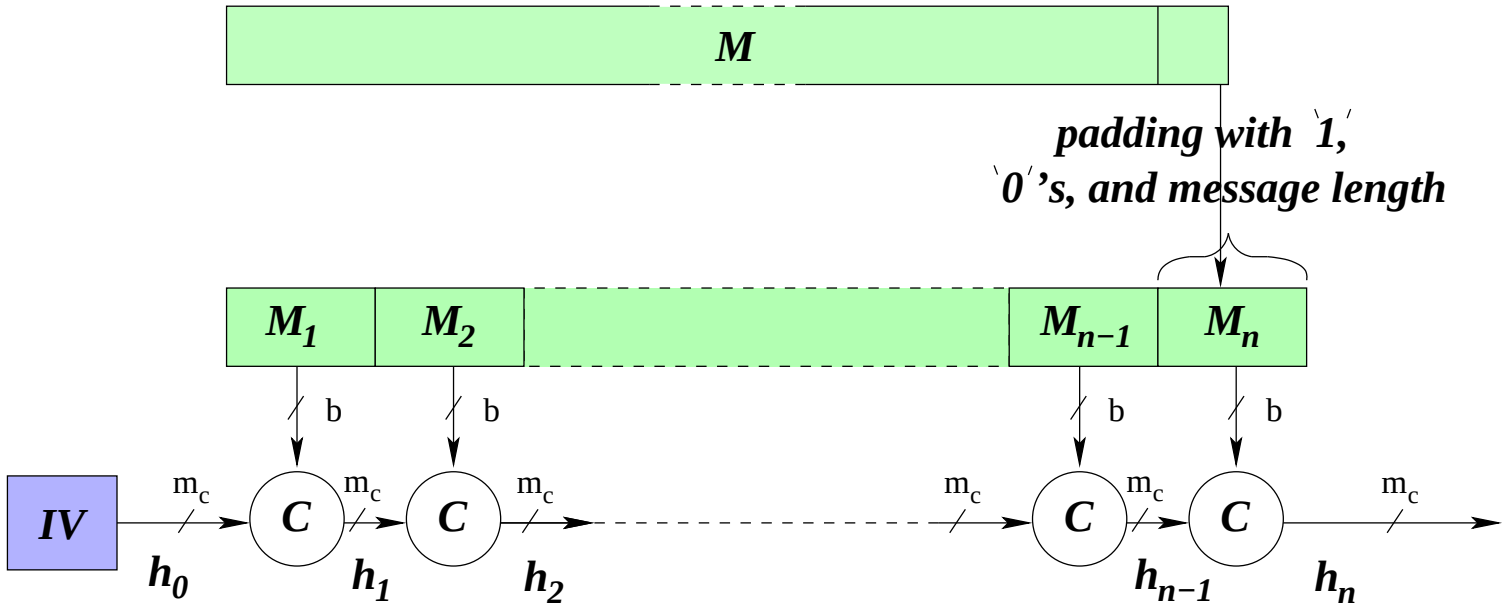
$$h_2 = \mathsf{C}(M_2, h_1)$$

Merkle-Damgård Construction (1989)



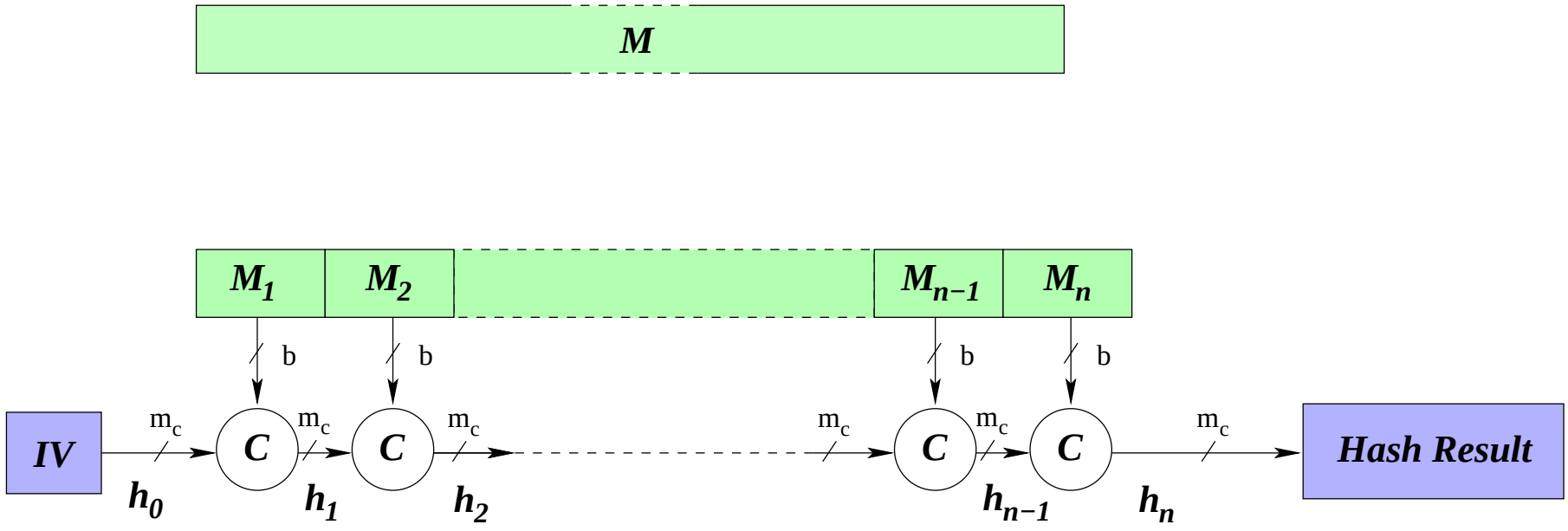
$$h_{n-1} = \mathsf{C}(M_{n-1}, h_{n-2})$$

Merkle-Damgård Construction (1989)



$$h_n = \mathsf{C}(M_n, h_{n-1})$$

Merkle-Damgård Construction (1989)



- h_n is the output of the hash function.

$$H(M) = h_n$$

Merkle-Damgård construction is the de-facto standard for hash functions.

Merkle-Damgård Construction

- The hash size should be long enough to prevent Yuval's type attacks.
- The padding of the length prevents some long messages second preimage attacks.
- The compression function is not invertible to prevent meet-in the middle attacks.
- $H(M)$ is collision free if $C(M_k, h_{k-1})$ is collision free.

Wang's MD5 Collision

- In 2005 Wang found a collision of MD5 with a complexity 2^{39}.

Wang's MD5 Collision

- In 2005 Wang found a collision of MD5 with a complexity 2^{39}.
- Wang's novel technique was exciting. However, was it more than academic achievement?

Wang's MD5 Collision

In particular, is this collision a security risk?

M

```
02DD31D1 C4EEE6C5 069A3D69 5CF9AF98 87B5CA2F AB7E4612 3E580440 897FFBB8
0634AD55 02B3F409 8388E483 5A417125 E8255108 9FC9CDF7 F2BD1DD9 5B3C3780
D11D0B96 9C7B41DC F497D8E4 D555655A C79A7335 0CFDEBF0 66F12930 8FB109D1
797F2775 EB5CD530 BAADE822 5C15CC79 DDCB74ED 6DD3C55F D80A9BB1 E3A7CC35
```

M^*

```
02DD31D1 C4EEE6C5 069A3D69 5CF9AF98 07B5CA2F AB7E4612 3E580440 897FFBB8
0634AD55 02B3F409 8388E483 5A41F125 E8255108 9FC9CDF7 72BD1DD9 5B3C3780
D11D0B96 9C7B41DC F497D8E4 D555655A 479A7335 0CFDEBF0 66F12930 8FB109D1
797F2775 EB5CD530 BAADE822 5C154C79 DDCB74ED 6DD3C55F 580A9BB1 E3A7CC35
```

- Notice that given a single collision of the hash function the number of colliding pairs is practically unlimited.

- Notice that given a single collision of the hash function the number of colliding pairs is practically unlimited.
  - E.g., If

$$H(m) = H(m^*)$$

    then

$$H(m||M) = H(m^*||M)$$

    for all M's.

- Notice that given a single collision of the hash function the number of colliding pairs is practically unlimited.
  - E.g., If

    $$H(m) = H(m^*)$$

    then

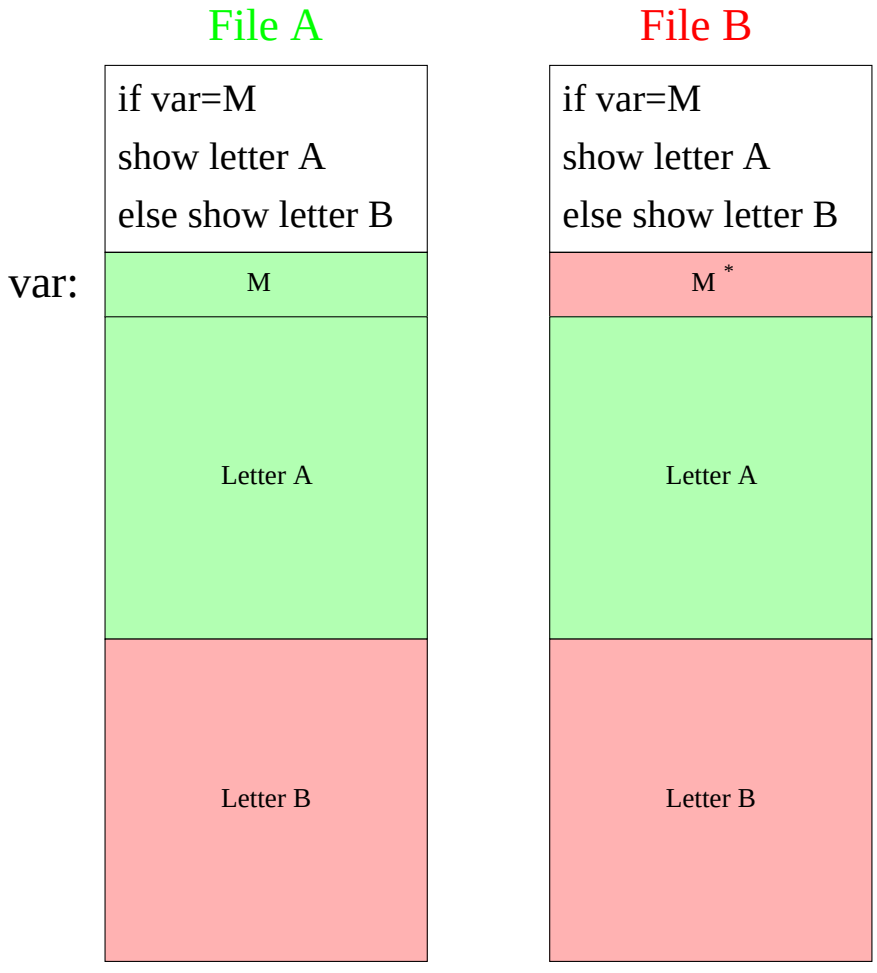
    $$H(m||M) = H(m^*||M)$$

    for all M's.
- But m and m^* are meaningless and can not be used in a real message.

Should a collision of a random and meaningless pair of messages worry us?

“The Story of Alice and Her Boss”, Lucks and Daum (2005)

Two postscript documents: File A shows Letter A, File B shows Letter B



$$C(h_i, M) = C(h_i, M^*) \longrightarrow H(\text{File A}) = H(\text{File B})$$

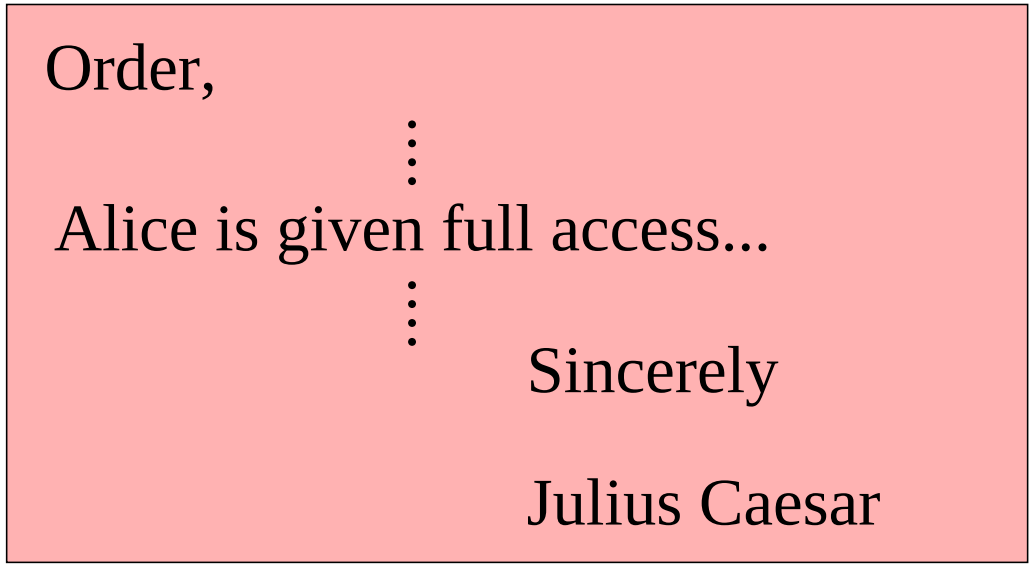
- Alice prepares file A and file B, sends file A to her boss, and asks him to sign.

- Alice prepares file A and file B, sends file A to her boss, and asks him to sign.
- Alice's boss is satisfied with what he sees (Letter A)

> To whom it may concern,
>
> ⋮
>
> I highly recommend hiring Alice...
>
> ⋮
>
> Sincerely
>
> Julius Caesar

and he signs.

- Alice prepares file A and file B, sends file A to her boss, and asks him to sign.
- Alice uses file B and shows Letter B signed by her boss.



- With this trick the same collision may be reused with as many letters Alice likes.
- The same trick is applicable to pdf and doc documents.
- This trick is applicable to any executable that is based on programming language...

Conclusion: Do not use a broken hash function

Conclusion: Do not use a broken hash function

Do software manufactures aware of the risk?

- from openSUSE 11.4 download page (2011):

“ Verify your download (optional, for experts)
Many applications can verify the checksum of a download. To verify your download can be important as it verifies you really have got the ISO file you wanted to download and not some broken version. You could verify the file in the process of downloading. For example a checksum (SHA256) will be used automatically if you choose Metalink in the field above and use the add-on DownThemAll! in Firefox. We offer three different checksums:
* gpg signature offers the most security as you can verify who signed it. It should be 79C1 79B2 E1C8 20C1 890F 9994 A84E DAE8 9C80 0ACA.
* md5 checksum is still the most commonly used checksum. Many ISO burners display it right before burning.
* sha1 checksum is the less known but more secure checksum than md5.”

- More than six years after SHA-1 was broken and MD5 collisions were published, they are still used in real applications.

- 2005: 800 calls of MD5 in Microsoft Windows. (Preneel’s talk, ICICS 2010)
- Android applications use RSA and MD5 for signature.

What if finding collisions is trivial, e.g, MD5 or MD4?

Rouge CA, Sotirov et al. (2008)

- A Certificate Authority (CA) is a trusted third party who issues and revokes certificates associating public encryption keys with the identity of their owners.
- Digital signatures are used by Certificate Authorities to sign certificates.
- An attacker who can forge certificates may impersonate any website on the Internet.
- In particular an attacker who can forge a certificate of a CA may impersonate any website on the Internet, including banking and e-commerce sites secured by the HTTPS protocol.

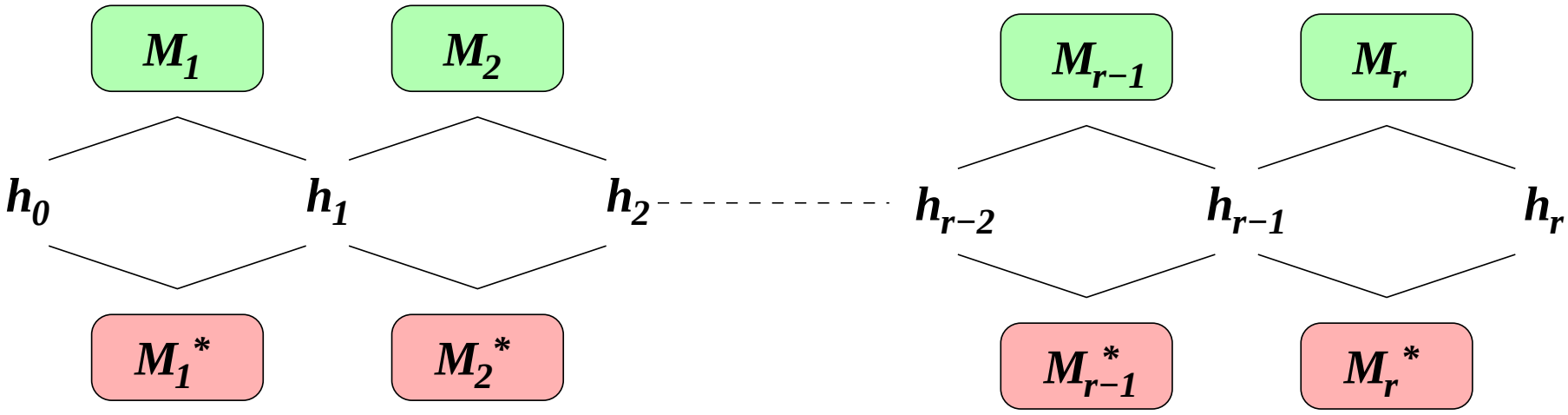
- Sotirov et al. demonstrated how collisions of MD5 are used to create a rogue CA certificate, which in turn allows the creation of valid certificates of arbitrary web sites.

- Microsoft response:
  "This new disclosure does not increase risk to customers significantly, as the researchers have not published the cryptographic background to the attack, and the attack is not repeatable without this information," ...

- 24/3/2011:
  Comodo a trusted internet security provider whose mission is to ‘create trust online’ gets a crucial hack attack issuing a fake digital SSL certificates. It is roaming on seven different domains including those of Live, Google, Yahoo, Skype, Mozilla and more.

Is $H(M)$ as strong as $C(M_k, h_{k-1})$?

Multi-Collision, Joux (2004)

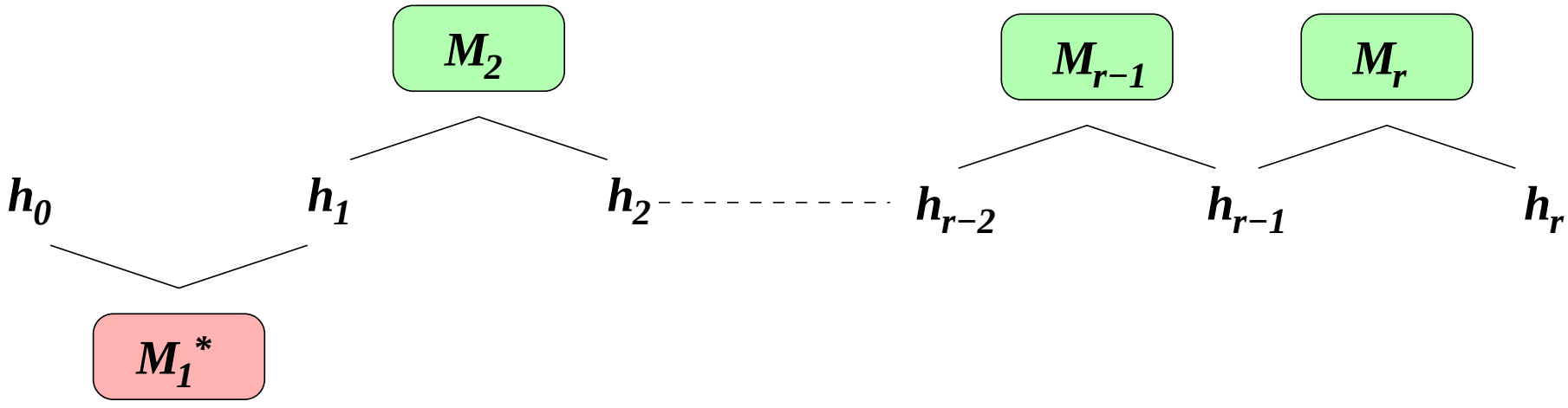


- r collisions of C() $\rightarrow 2^r$-collisions of H(), i.e., 2^r messages have the same hash value.

Multi-Collision, Joux (2004)

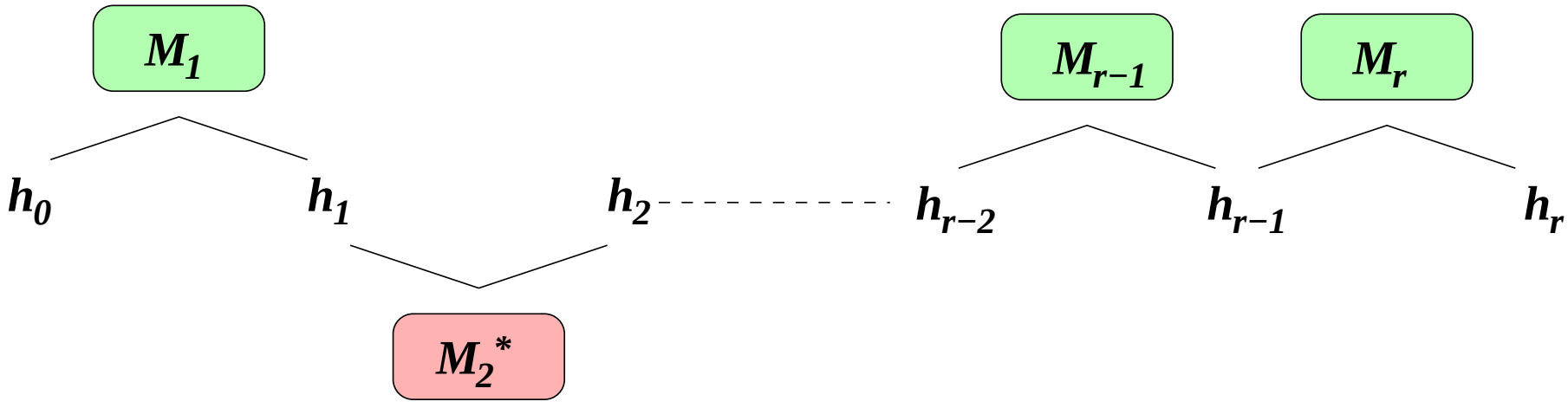
- r collisions of C() $\rightarrow 2^r$-collisions of H(), i.e., 2^r messages have the same hash value.

Multi-Collision, Joux (2004)



- r collisions of C() $\rightarrow$ 2^r-collisions of H(), i.e., 2^r messages have the same hash value.

Multi-Collision, Joux (2004)

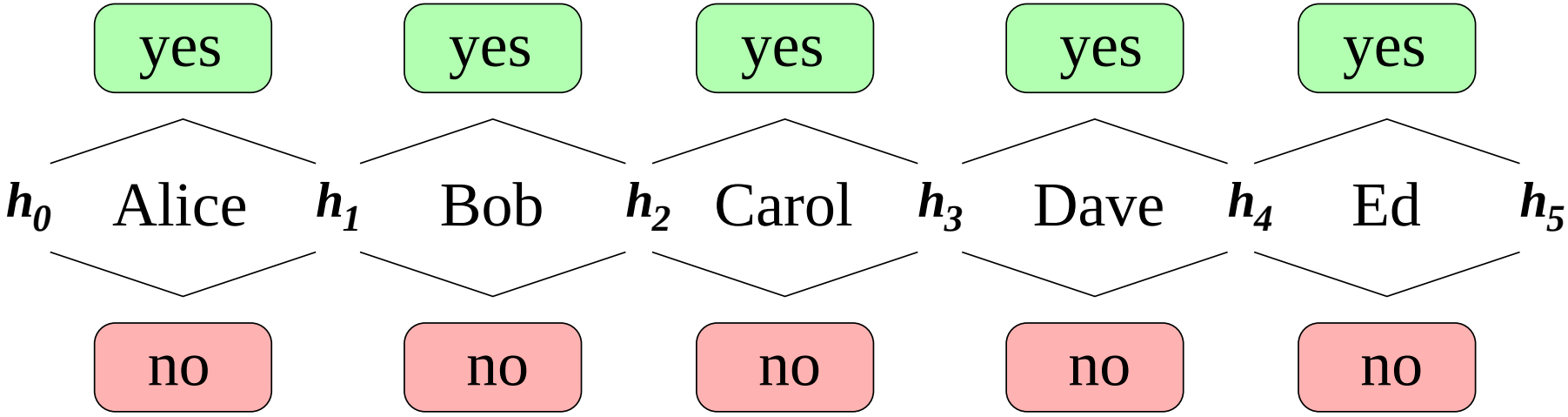


- r collisions of C() $\rightarrow$ 2^r-collisions of H(), i.e., 2^r messages have the same hash value.

- Multi-collisions are used to show that cascading two hash functions is not much stronger than the strongest of the two (in respect to collision resistance and preimage resistance).

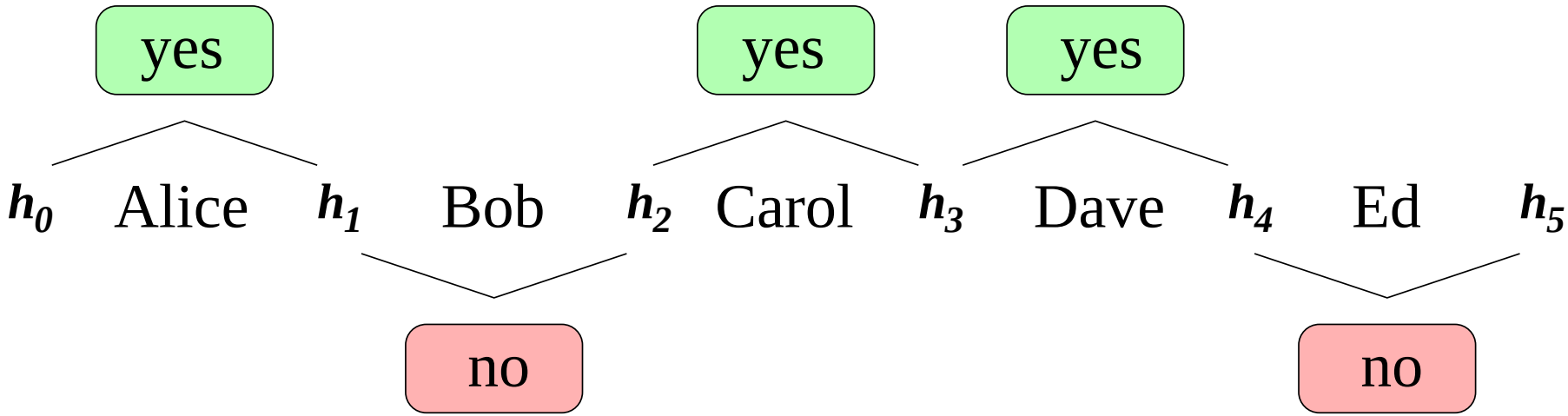
Nostradamus Attack, Kelsey and Kohno(2005)

- Nostradamus commitment to “which celebrities will marry this year” is $H(M) = h_5$:



Nostradamus Attack, Kelsey and Kohno(2005)

- Nostradamus commitment to “which celebrities will marry this year” is $H(M) = h_5$:
- At the end of the year he reveals...



Differential Cryptanalysis of Hash Functions

Differential Cryptanalysis of H.F.'s

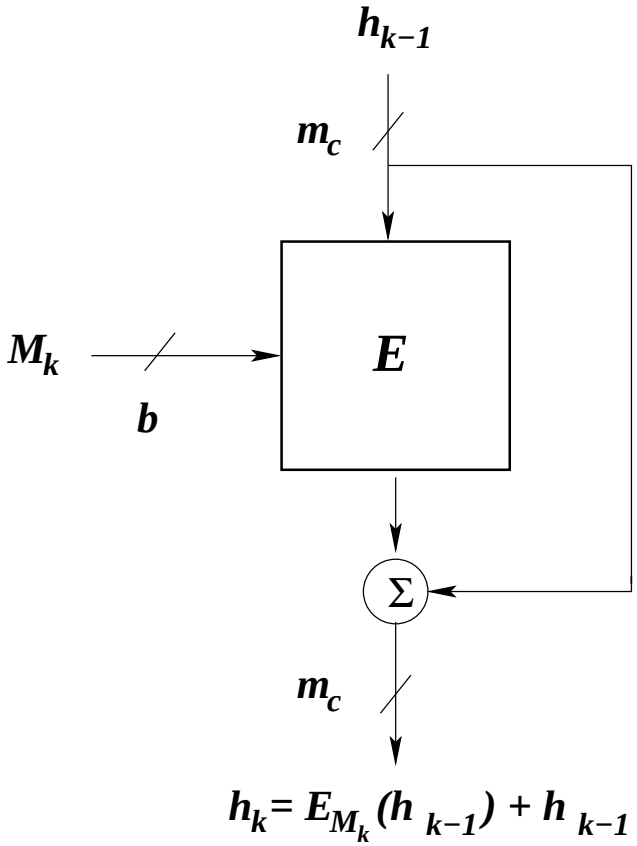
- Our research is focused on attacking the collision resistance property.
- The most general, efficient and widely used technique to attack the collision resistance property is differential cryptanalysis that was introduced by Biham and Shamir in 1990.
- In our research we use and enhance the differential cryptanalysis technique.

Differential Based Attacks

- In 1998 Chabaud and Joux published an attack on SHA-0.
- In 2004 we published our neutral-bits technique with application to SHA-0.
- In 2005 we published the multi-block technique and the first attacks on SHA-1.
- Joux used our techniques added an improvement and found a collision of SHA-0.
- Wang used some of our techniques, introduced substential improvements of her own and broke SHA-1,

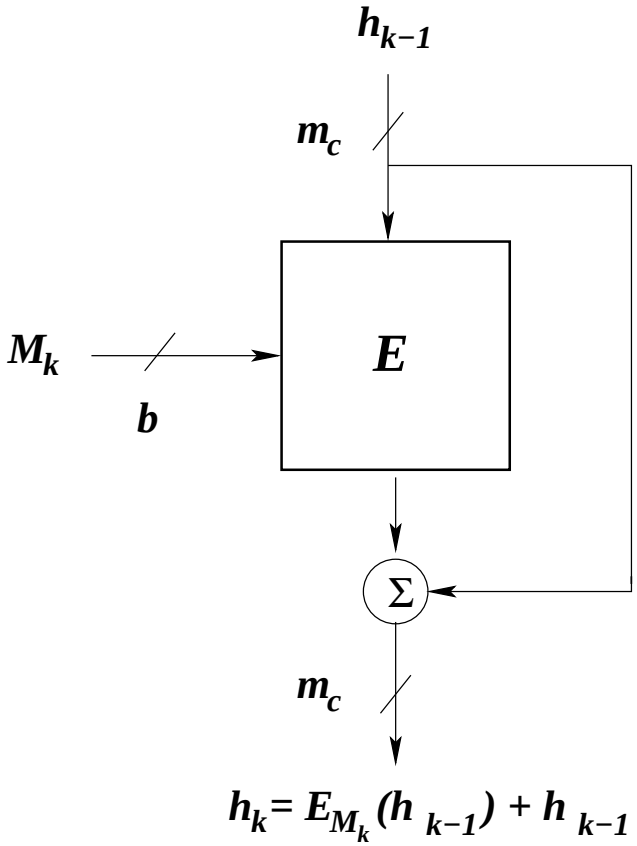
- In 2005 Wang published her modular differential and message modification techniques to attack MD4, MD5, HAVAL, RIPEMD-128, SHA-0 and SHA-1.
- Recently we have developed the second order differential technique.

Compression-Function Design



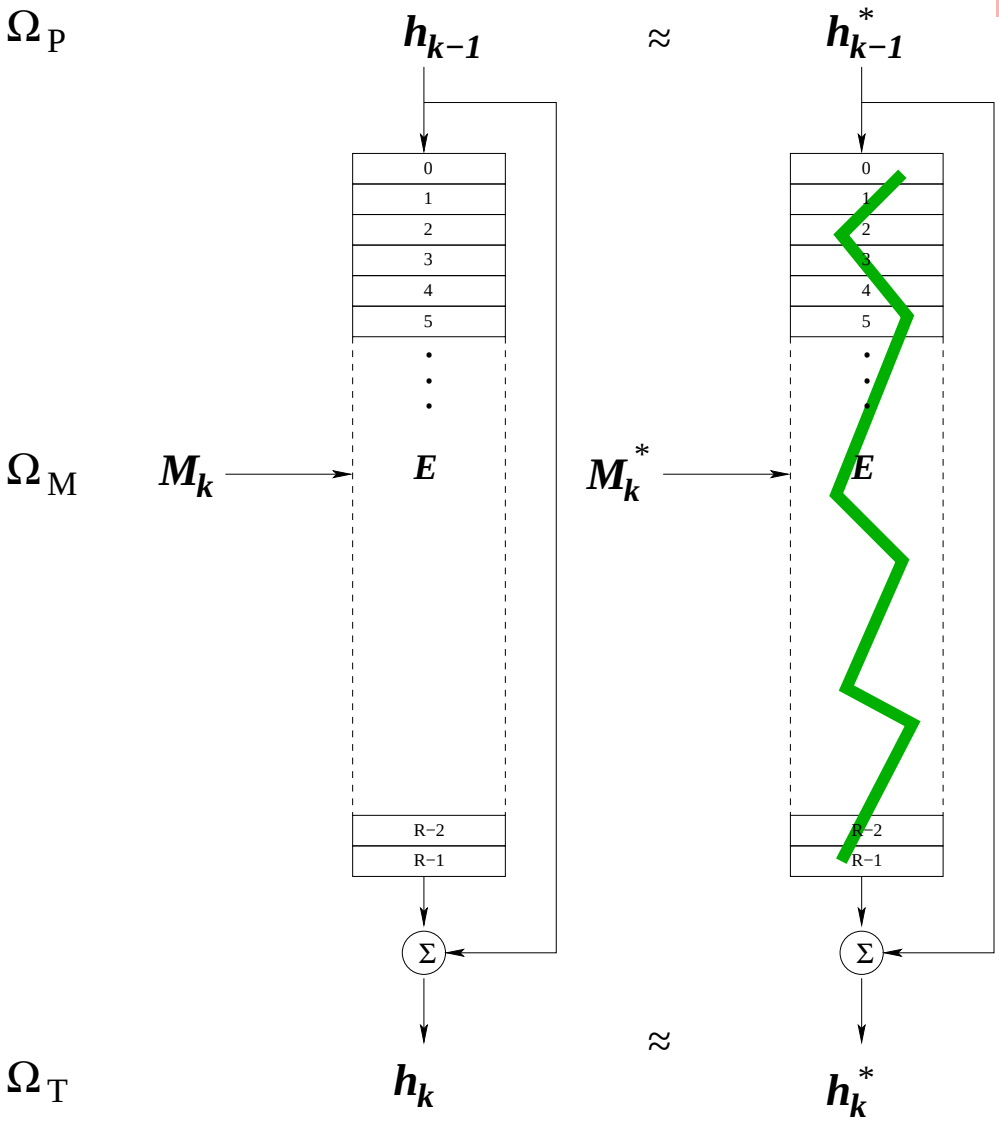
- Based on an encryption function surrounded by a feed-forward that cancels the ability to decrypt.

Compression-Function Design



- Based on an encryption function surrounded by a feed-forward that cancels the ability to decrypt.
- The message is used as the “key” and the chaining value as the “plaintext”.

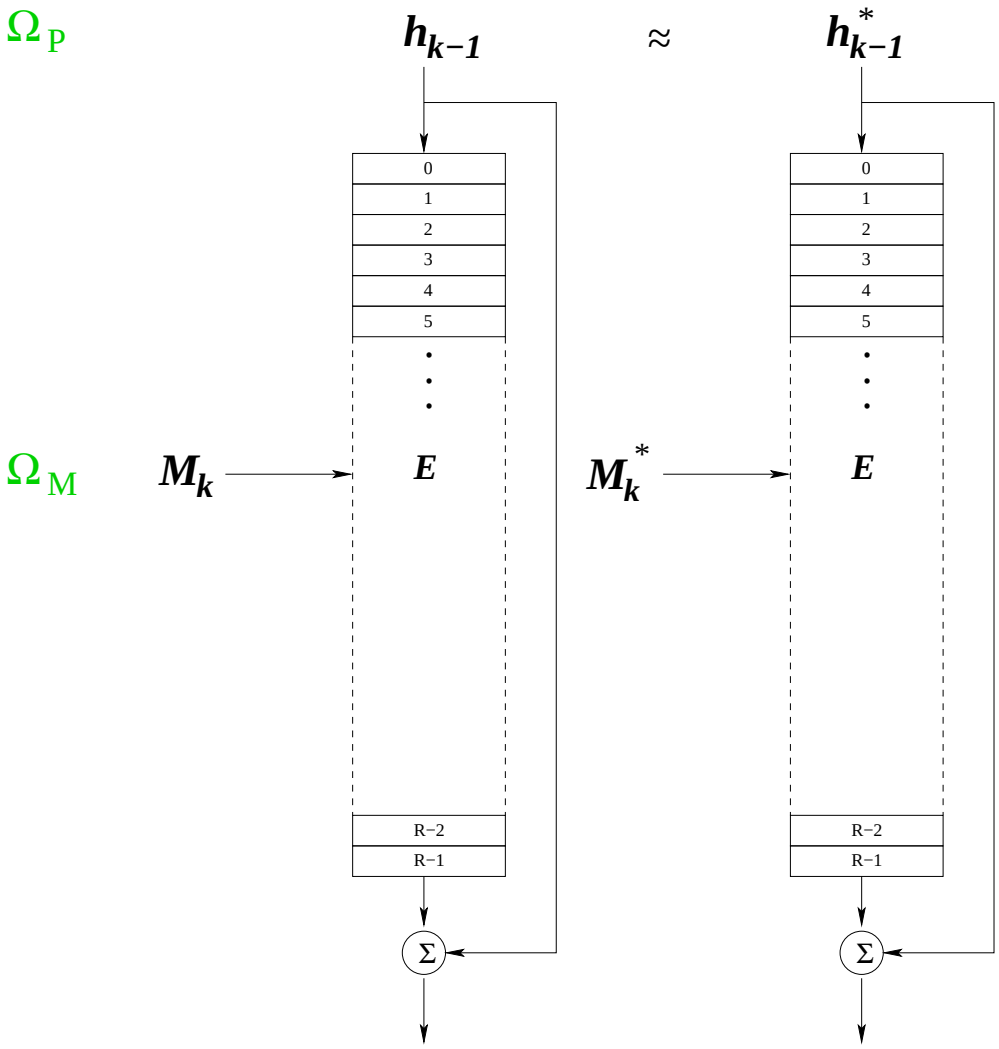
Differential Cryptanalysis of H.F.'s



The idea:

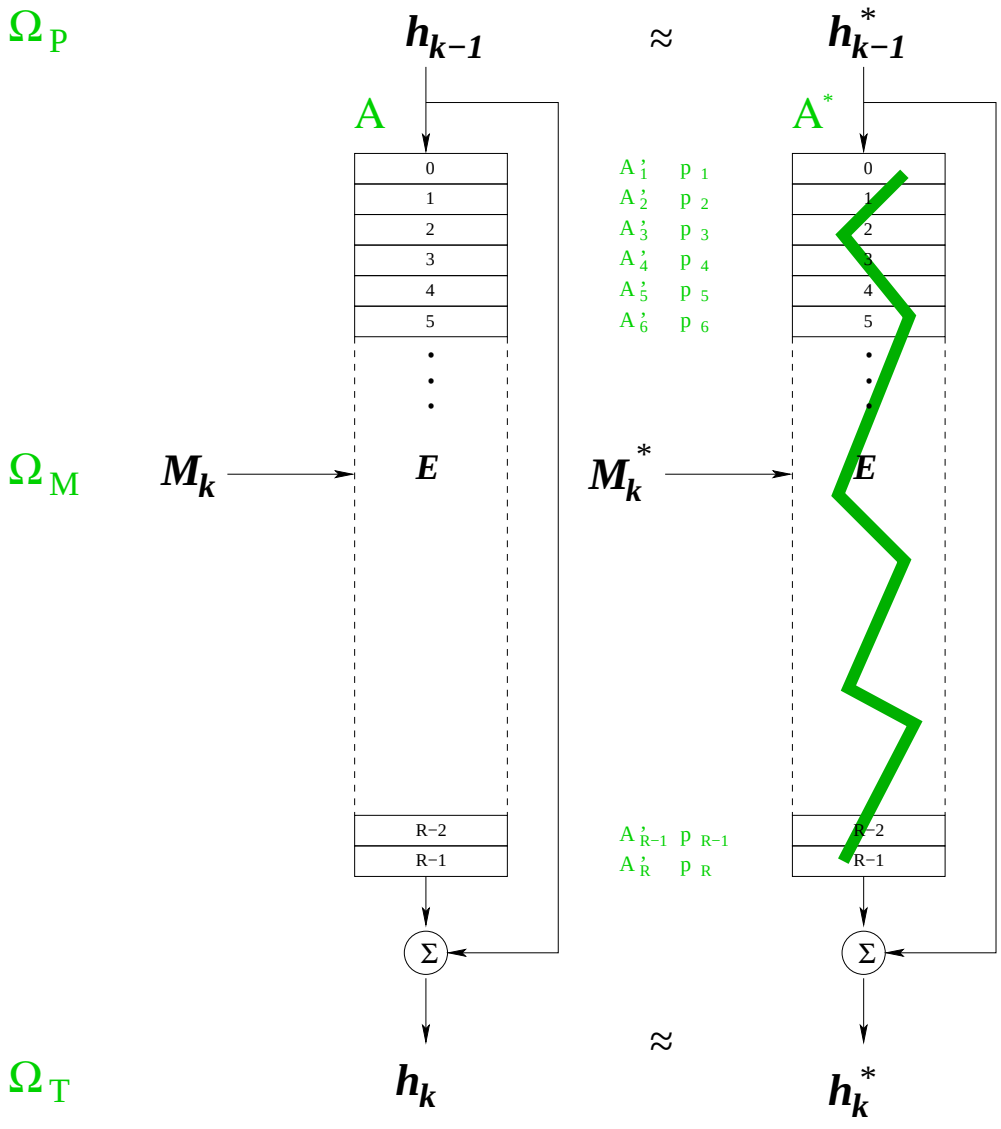
- Differences are easier to predict than values.

Differential Cryptanalysis of H.F.'s



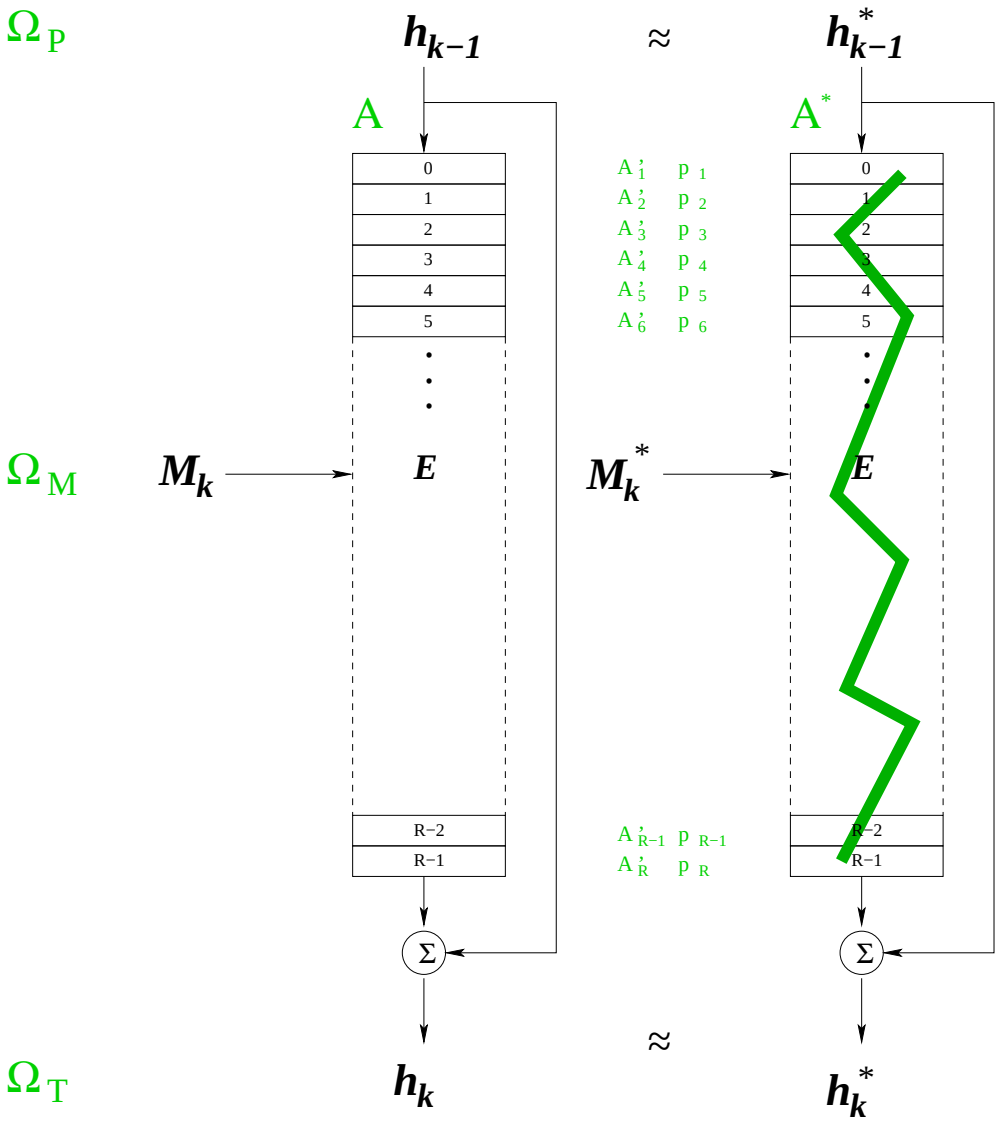
- An attacker selects input differences Ω_p and Ω_M,

Differential Cryptanalysis of H.F.'s



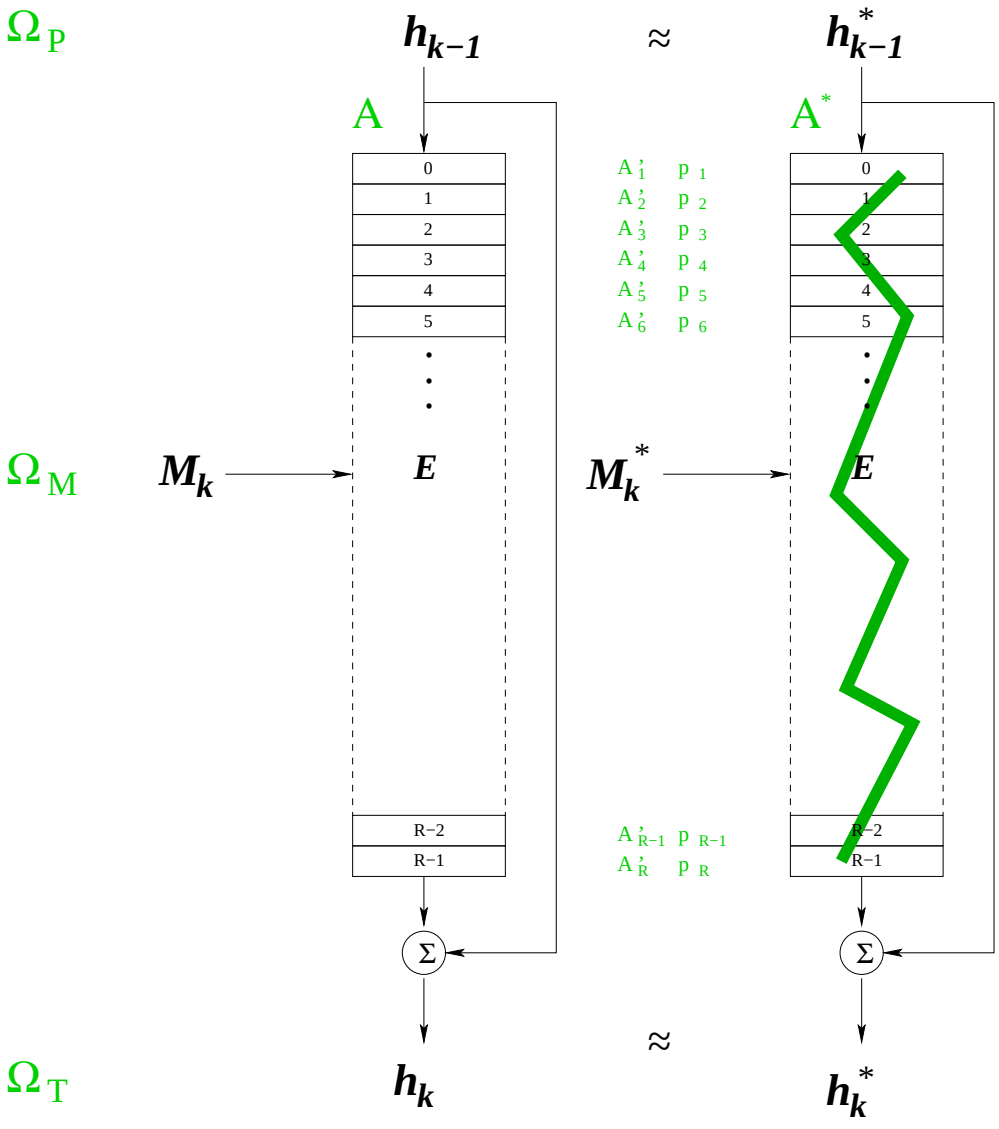
An attacker selects input differences Ω_p and Ω_M, and analyzes the predicted difference of the internal state A'_i and the probability p_i that it occurs, in each round up to the output difference Ω_T.

Differential Cryptanalysis of H.F.'s



- The list of predicted differences and probabilities forms a characteristic.

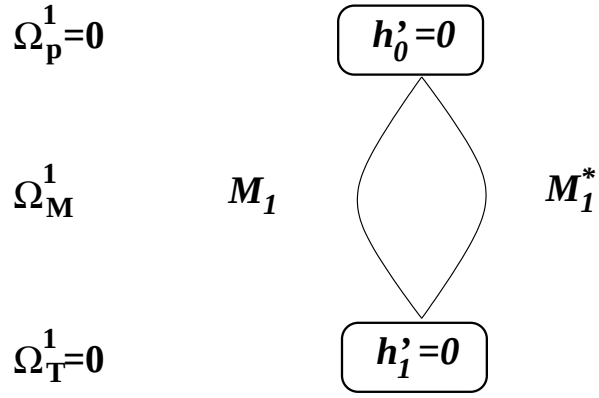
Differential Cryptanalysis of H.F.'s



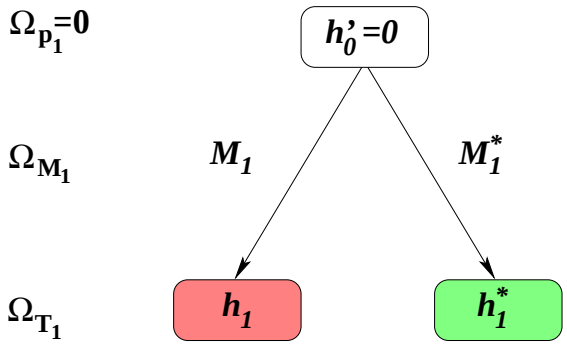
- The list of predicted differences and probabilities forms a characteristic.
- The probability of the characteristic is $\prod_{i=1}^{r} p_i$.

The Multi-Block Technique

A Characteristics of One-Block Attack



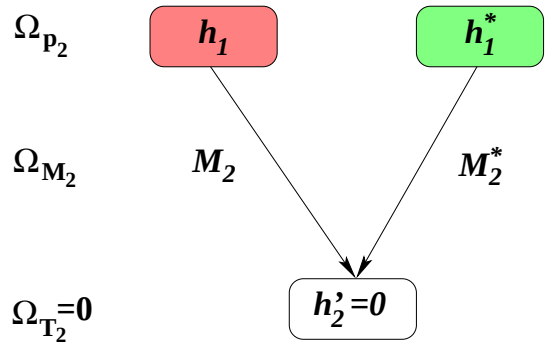
A Characteristics for Near-Collision



$$C(h_0, M_1) \approx C(h_0, M_1^*)$$

Near−Collision

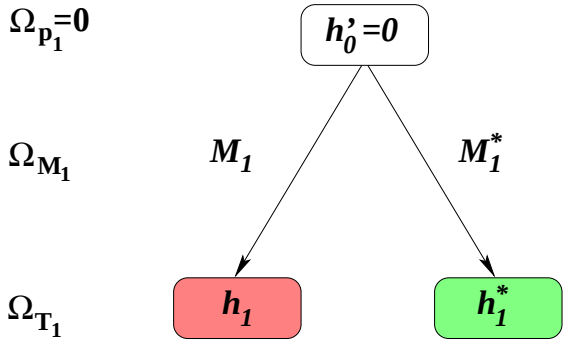
A Characteristic for Pseudo-Collision



$$C(h_1, M_2) = C(h_1^*, M_2^*)$$

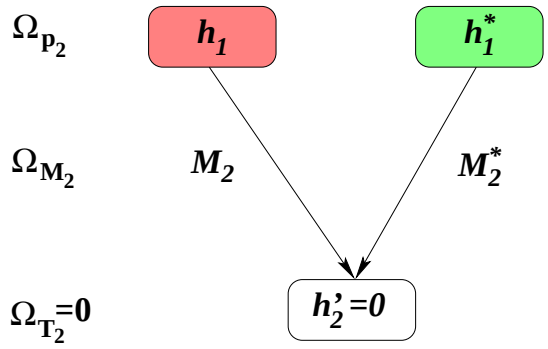
Pseudo−Collision

Characteristics for Two-Block Attack



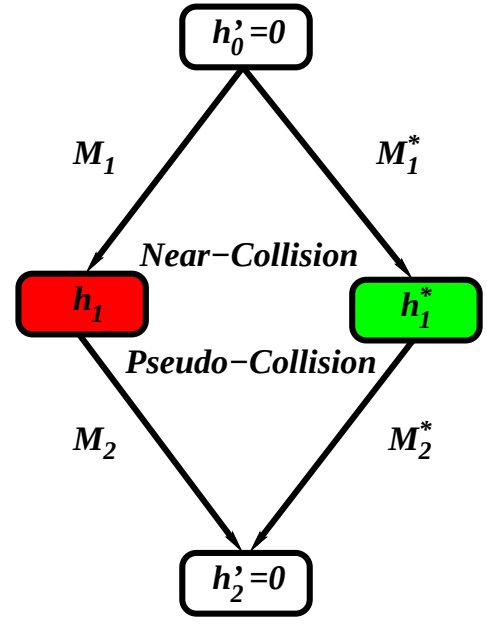
$C(h_0,M_1) \approx C(h_0,M_1^*)$

Near−Collision



$C(h_1,M_2) = C(h_1^*,M_2^*)$

Pseudo−Collision



Two−Block Collision

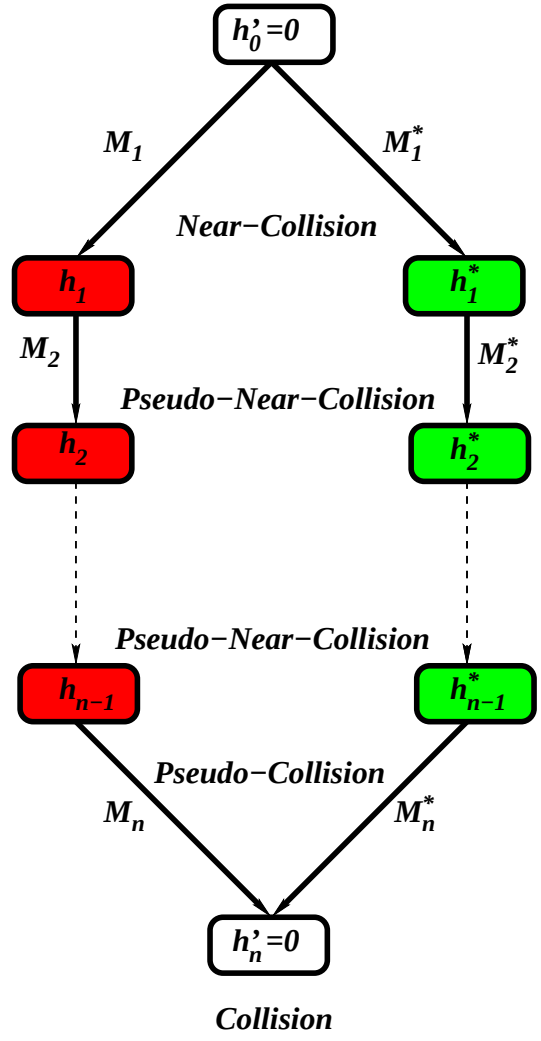
Multi-Block Attack

The first pair creates a near−collision.

The second pair starts with a small difference in the initial value and ends with a near−collision.

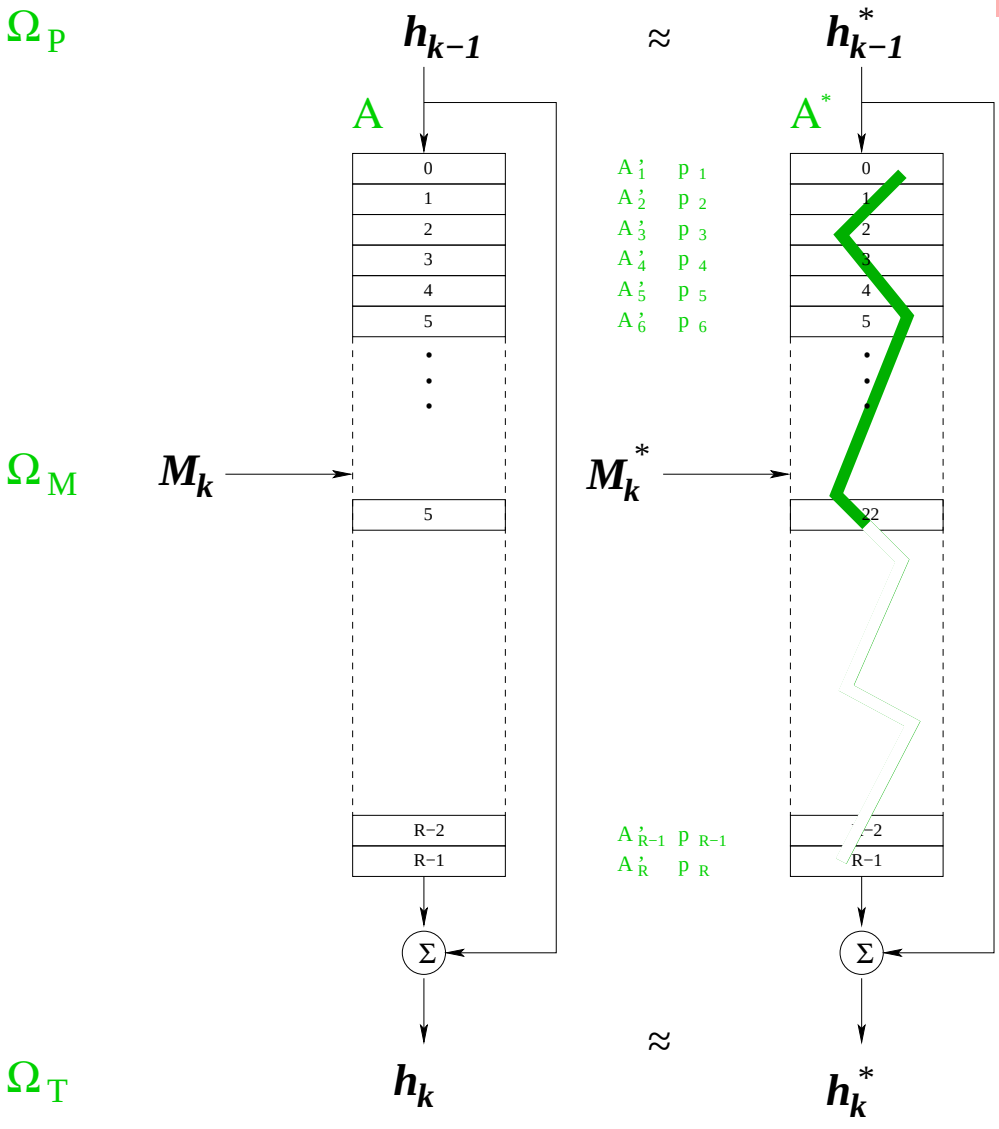
Additional pairs are added as necessary to reduce the search complexity.

The last pair is a pseudo−collision.



The Neutral-Bits Technique

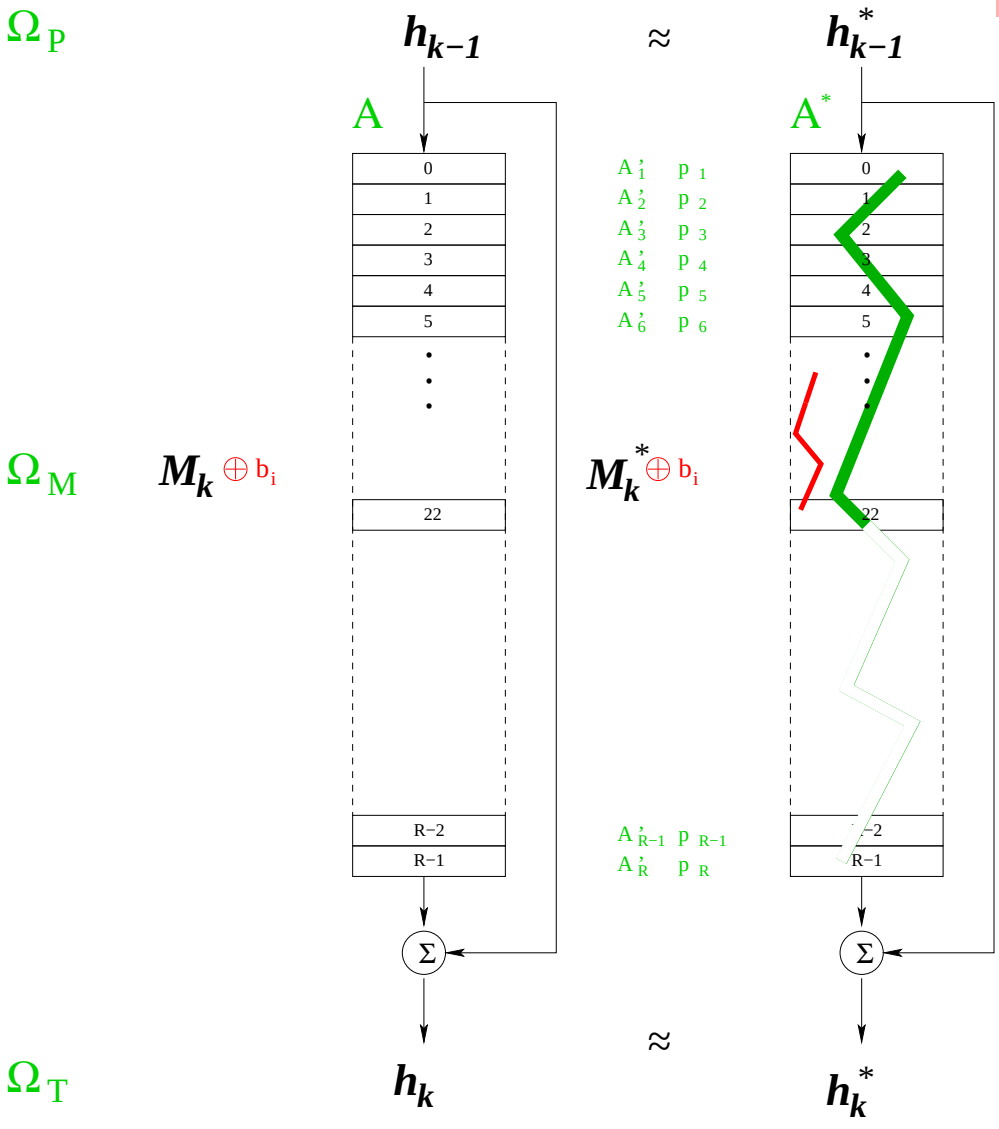
Neutral Bits



The idea:

- Let a pair M_k, M_k^* conforms to the characteristic at least up to Round 22.

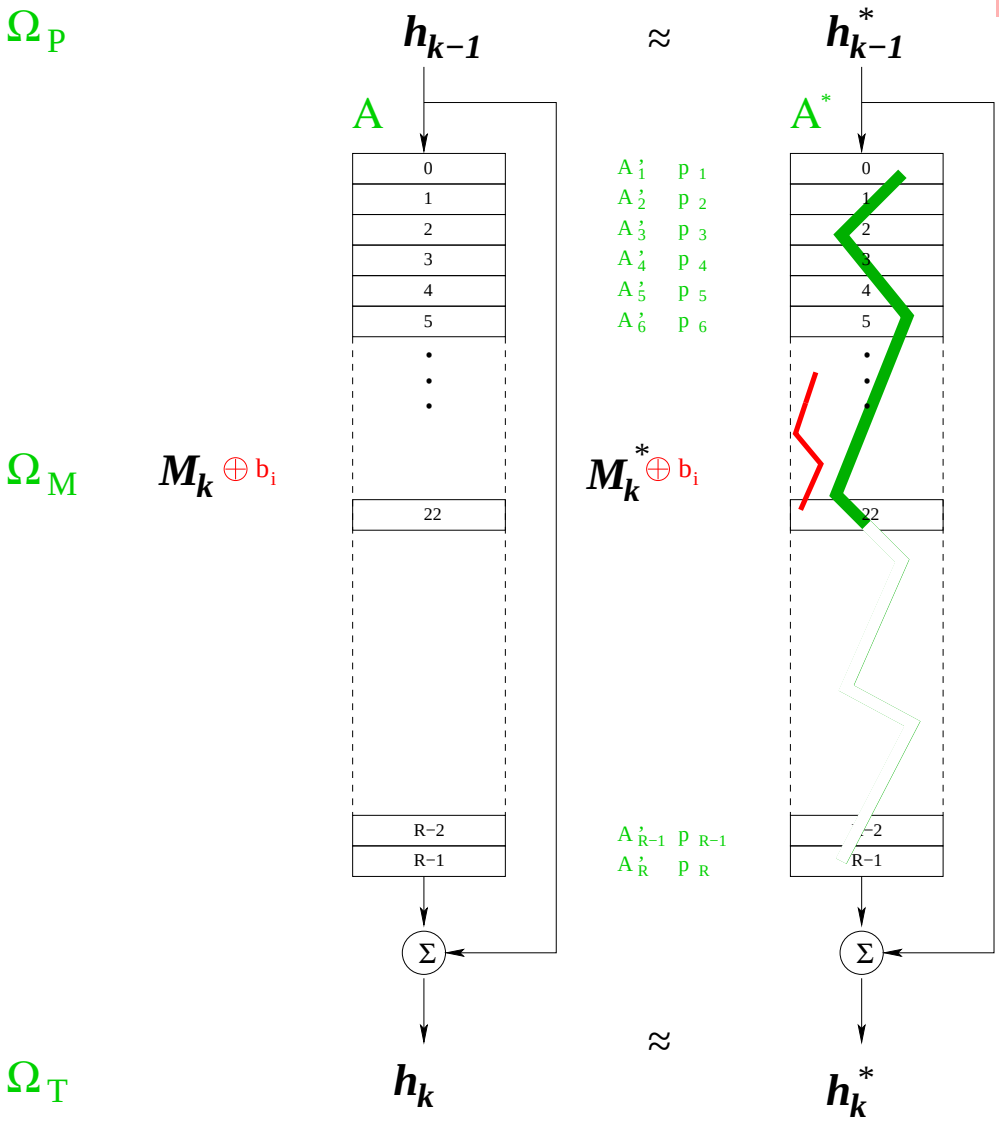
Neutral Bits



The idea:

- Let a pair M_k, M_k^* conforms to the characteristic at least up to Round 22.
- Complement bit i in both messages.

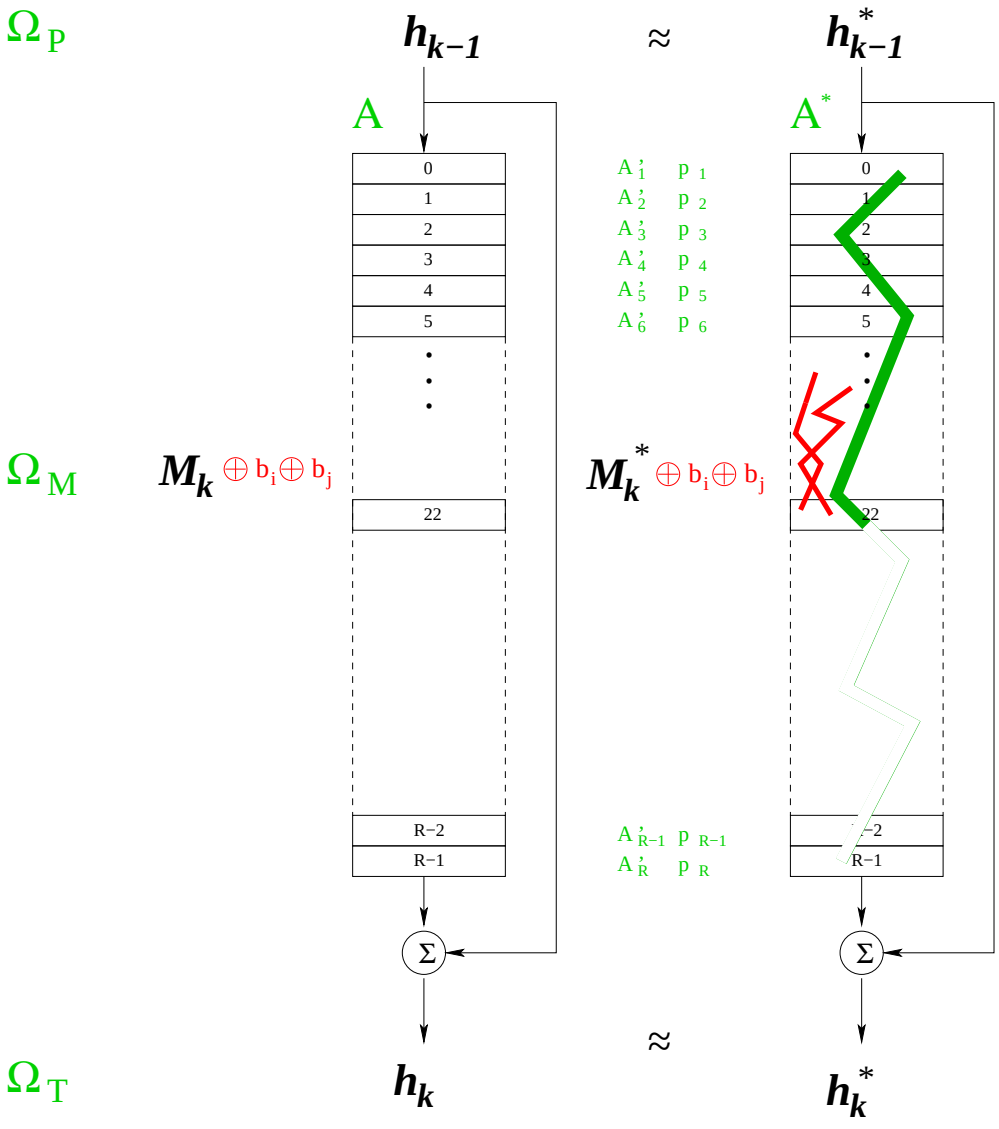
Neutral Bits



The idea:

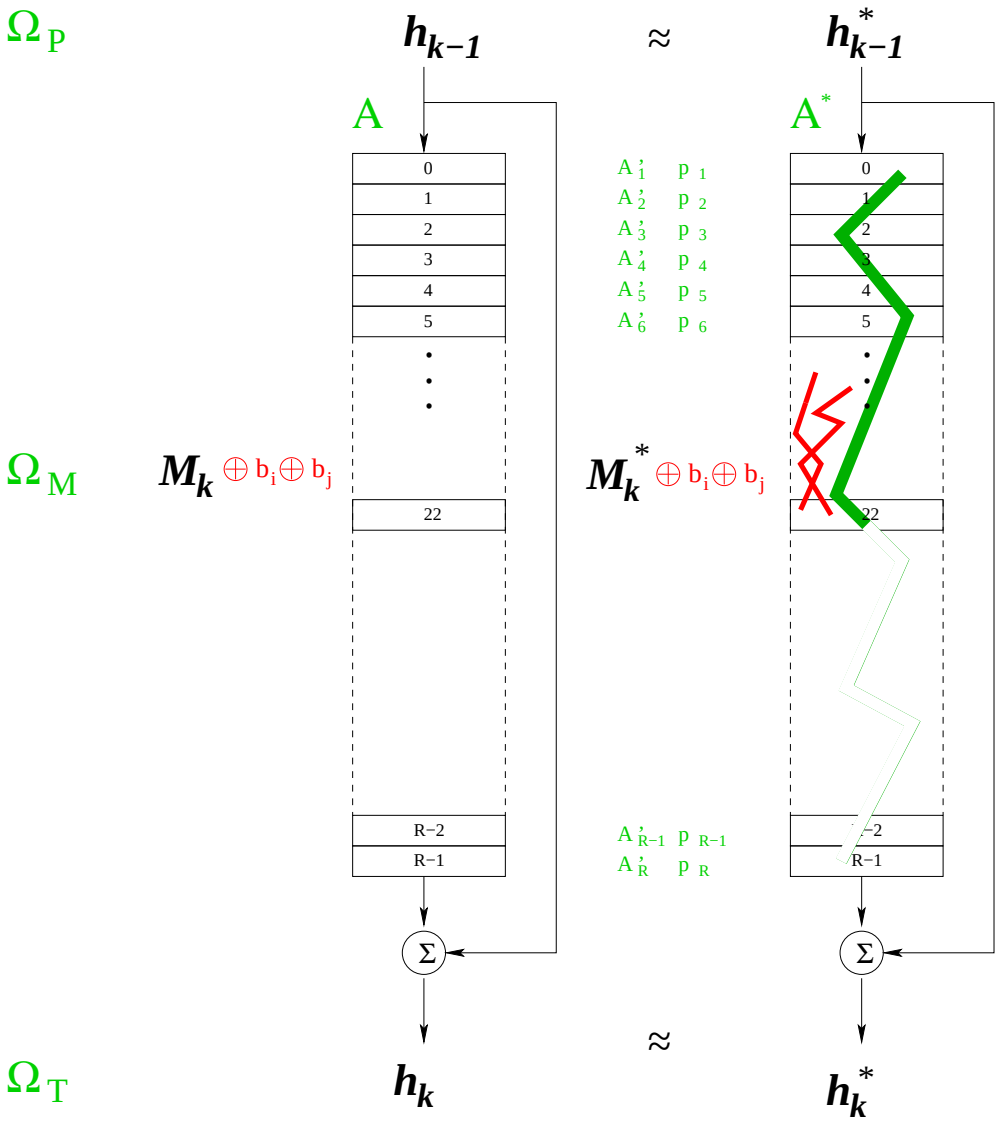
- Let a pair M_k, M_k^* conforms to the characteristic at least up to Round 22.
- Complement bit i in both messages.
- If the conformance of the new pair is not affected up to Round 22, then b_i is a neutral bit.

Neutral Bits



- Now Complement bit j in both messages.

Neutral Bits



- Now Complement bit j in both messages.
- If the conformance of the new pair is not affected up to Round 22, then b_i and b_j are mutually independent neutral bits.

Neutral Bits

- In SHA-0 it is easy to find sets of more than 40 mutually independent neutral bits.
- By complementing the 2^{40} different combinations of neutral bits we receive 2^{40} new messages, from which about 2^{37} conforms at least to Round 22.
- Using this technique the probability of the characteristic is effectively $\prod_{i=22}^{R} p_i$.

Example

- The following pair conforms to 22 rounds and has about 40 neutral bits from which about 2^{37} pairs that conforms to 22 rounds may be constructed.

M_1	19EF75A8	D2F24D9A	8F179A7D	1A295690
	2E84C143	D74B9DDC	18C10577	8107056E
	5B1A47ED	6212C3F2	3B2D04F8	F5581AB0
	26D8CDBC	AB3A3248	F347E871	46278F39
M_1^*	19EF75A8	D2F24D9A	8F179A7D	1A295692
	2E84C103	D74B9DDE	98C10577	0107056E
	DB1A47EF	6212C3B2	3B2D04F8	75581AF0
	A6D8CDBE	AB3A324A	7347E831	C6278F3B

Example (cont.)

Singles:	$W_{12}^4, W_{14}^9, W_{14}^{10}, W_{14}^{11}, W_{14}^{16}, W_{15}^4, W_{15}^5, W_{15}^9, W_{15}^{10}, W_{15}^{11}, W_{15}^{14}, W_{15}^{15}, W_{15}^{16}, W_{15}^{19}, W_{15}^{21}, W_{15}^{26}, W_{15}^{27}$
Pairs:	$(W_9^{13}, W_8^8), (W_{14}^{13}, W_{13}^8), (W_{15}^{13}, W_{14}^8), (W_{15}^{17}, W_{14}^{12}), (W_{15}^{20}, W_{14}^{15}), (W_{15}^{22}, W_{13}^{12})$
Triplets:	$(W_9^8, W_5^{15}, W_4^{10}), (W_{10}^{21}, W_6^{28}, W_5^{23}), (W_{11}^{24}, W_7^{31}, W_6^{26}), (W_{12}^2, W_8^9, W_7^4), (W_{12}^7, W_8^{14}, W_7^9), (W_{14}^{14}, W_{13}^{10}, W_{13}^9), (W_{14}^{18}, W_{13}^{13}, W_{12}^9), (W_{15}^8, W_{15}^3, W_{14}^{30}), (W_{15}^{12}, W_{10}^{14}, W_9^9)$
Quadruplets:	$(W_7^5, W_4^9, W_3^{12}, W_2^7), (W_{10}^{11}, W_6^{18}, W_3^{20}, W_2^{15}), (W_{11}^{12}, W_{10}^{18}, W_{10}^{17}, W_9^{12})$ $(W_{14}^7, W_{13}^{19}, W_{13}^{18}, W_{12}^{16}), (W_{15}^{25}, W_{13}^{21}, W_{13}^{15}, W_{12}^{16})$
Quintuplets:	$(W_{14}^{23}, W_{14}^{22}, W_{14}^{21}, W_{13}^{17}, W_{12}^{11}), (W_{15}^7, W_{14}^{17}, W_{10}^{24}, W_{10}^{23}, W_9^{18}),$ $(W_{15}^{24}, W_{15}^0, W_{14}^3, W_{13}^{22}, W_{13}^4), (W_{15}^{24}, W_{15}^0, W_{14}^3, W_{13}^{22}, W_{13}^4)$

Results Using Our Techniques

H.F.	Round	Blocks	Complexity		Found
			pairs	SHA calls	
SHA-0	50	2	2^{19}	2^{16}	+
	80	4	2^{51}	2^{46}	+
	82	1	2^{44}	2^{39}	+
SHA-1	34	1	2^{7}	2^{4}	+
	36	2	2^{24}	2^{21}	+
	40	2	2^{19}	2^{16}	+
	53	1	2^{49}	2^{46}	
	58	2	2^{53}	2^{50}	
	80	3		2^{58}	

Summary

- The research of hash functions in the last seven years received a lot of attention but we still do not have a recommended solution.
- SHA-2 is safer than SHA-1 but it suffers from Merkle-Damgaård weaknesses.
- The announcement on SHA-3 recommended algorithm is planned for 2012.
- Though the threats of using a broken hash function are clear and real, broken hash functions are still in use.

- According to the Israeli low, SHA-1 is not allowed anymore.
  RIPEMD-160 may be used till the end of 2012.
  SHA-2 and Whirlpool are allowed with no limitations.

- Low and Reality: According to the Ministry of Justice "COMSIGN ltd" is the only authorized CA in Israel. However, their certificate is signed by PKCS #1 SHA-1 With RSA Encryption (the signer is veriSign). For the fingerprint they use MD5 and SHA-1...