CS915/435 Advanced Computer Security - Elementary Cryptography

Hash

Roadmap

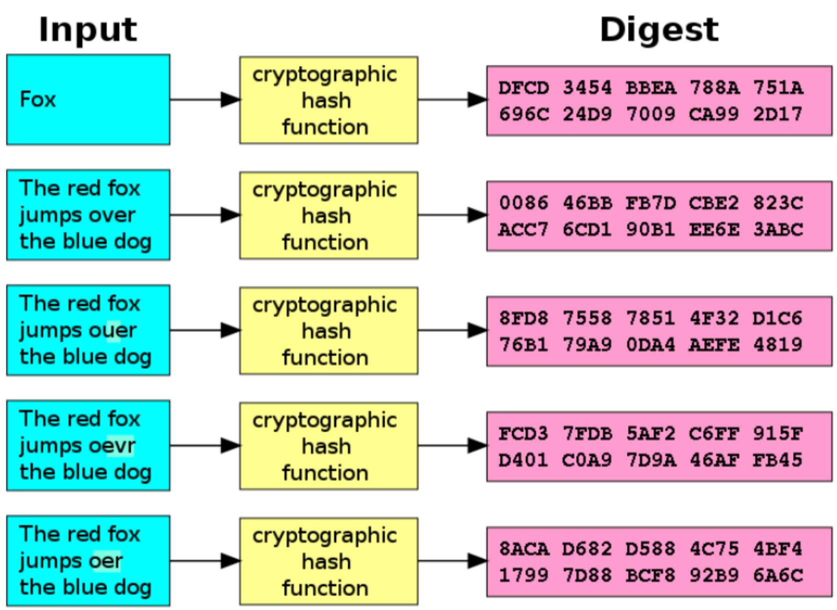
- Symmetric cryptography
  - Classical cryptographic
  - Stream cipher
  - Block cipher I, II
  - **Hash**
  - MAC
- Asymmetric cryptography
  - Key agreement
  - Public key encryption
  - Digital signature

Hash function

- Compress an arbitrary message into an output of fixed length (checksum)
  - Being used since 1950s
  - To facilitate detecting errors or data comparison
- Checksum is primarily used for error detection
- Hash functions (e.g., SHA256) is more complex
- Designed for broader applications
  - Data integrity, digital signatures

Cryptographic hash function

- Invented for digital signature
  - Provide assurance of data integrity
  - Avalanche effect



(Wikipedia)

Abstraction: random oracle model

Black box

Query

Response

If the query is **new**

She (i.e., oracle) gives you a fixed-length random string and marks a record on her book

Else

She looks up the book and gives you the same previous answer.

But ideal random oracle is impossible

- How to ensure that each output represents only one input message?
- That’s theoretically impossible
  - The message space size is much larger than the size of the output space
- Practical solution
  - Ensure that it is computationally infeasible to find two messages with the same output

Security requirements of hash

1. Pre-image resistance
   - Given $H(m)$, can't find m
2. Second pre-image resistance
   - Given m_1, can't find a different message m_2 such that $H(m_1) = H(m_2)$
3. **Collision resistance**
   - Can't find two different messages m_1 and m_2 such that $H(m_1) = H(m_2)$

Background: birthday paradox

- To guarantee we find two people with the same birthday, we will need 366 people.
- But in practice:
  - It’s almost guaranteed that there are at least two students with the same birthday in this classroom.

Calculating the probability

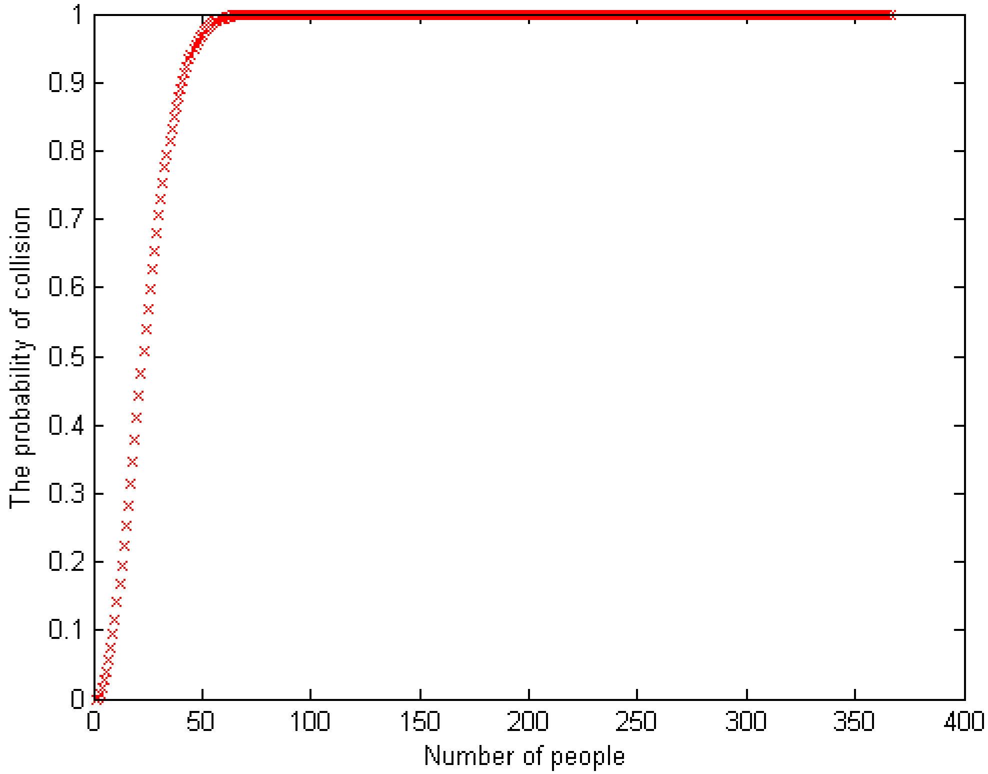
- Let P(n) be the probability of finding two same birthdays given n students in the class.
- To compute P(n), it is easier to reason with P'(n), the probability that n students are **not** born the same day
  - Obviously, $P(n) = 1 - P'(n)$

- $P(1) = 0$

- $P(2) = 1 - 364/365$

- $P(3) = 1 - (364/365) \times (363/365)$

The probability of finding the same birthday



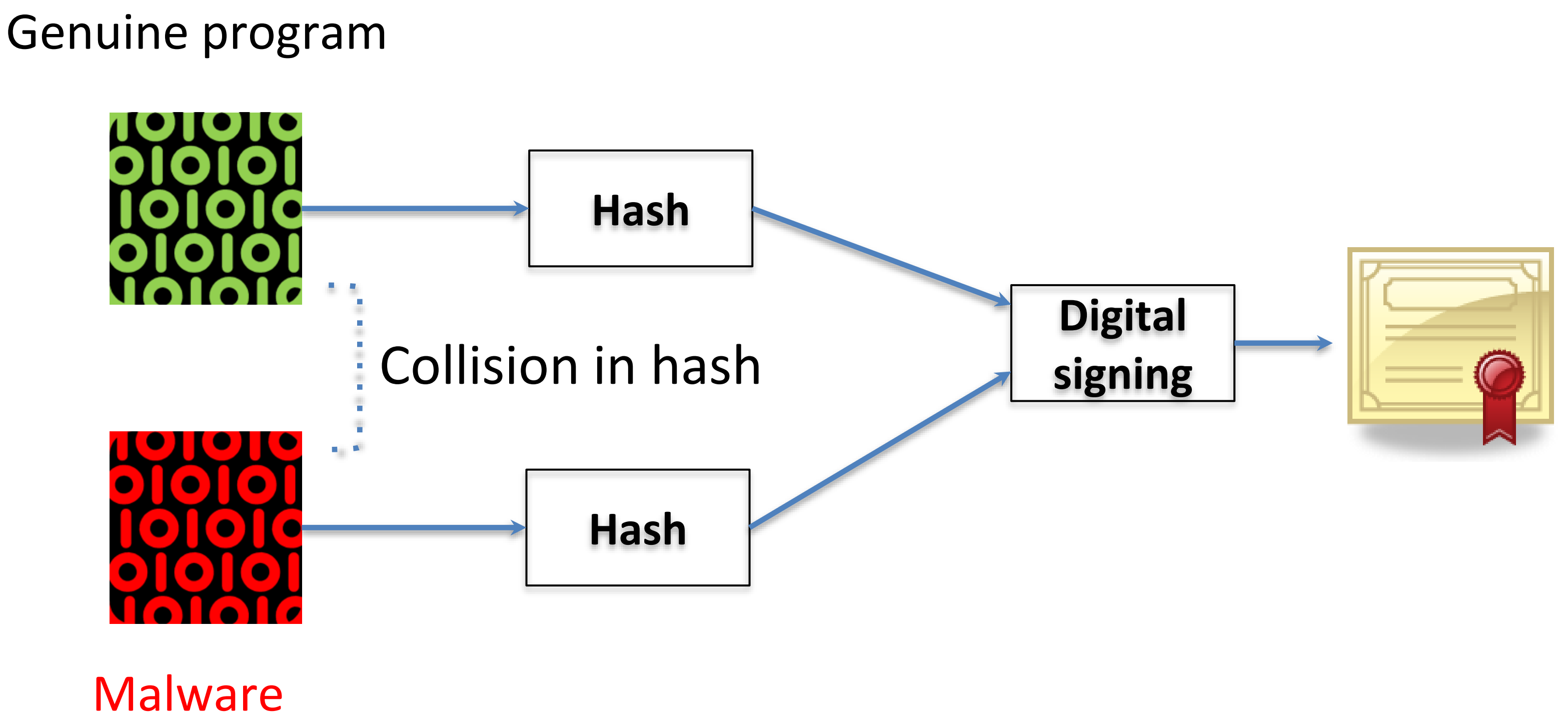
Birthday attack on collision resistance

- Assume a hash function with n bits output
- Birthday attack algorithm
  1. Select $2^{n/2}$ random input messages
  2. Compute the hash of each input message
  3. Look for a collision among the output. (If not found, go back to step 1)
- Implication from the attack
  – For n-bit security, the output of hash must be at least 2n bits long.

Hash family

- MD5 (1991)
  - 128-bit message digest
  - Broken by Wang Xiaoyun et al in 2005
- NIST standard: Secure Hash Algorithm
- SHA-1 (1995)
  - 160-bit message digest
  - Insecure (2^{69}, Wang Xiaoyun et al 2005)
- SHA-2 (2001)
  - SHA-256
  - SHA-512
- SHA-3 (2015)

Collision in hash is dangerous



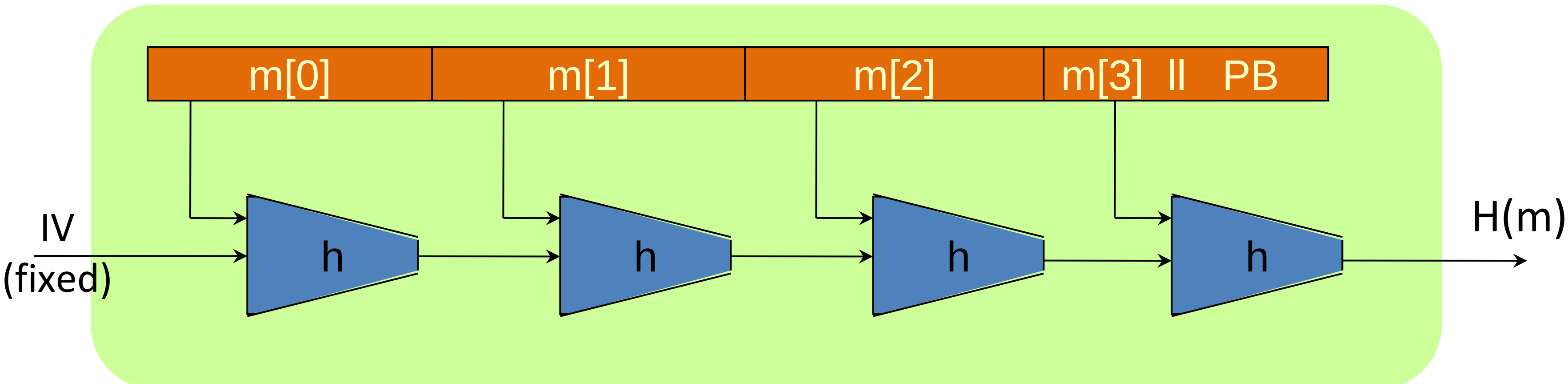
Real-world example: Flame malware

- Detected by CERT in May 2012
- Advanced espionage malware
- It exploits MD5 collision
  - Microsoft Terminal Server Licensing Service certificate still uses MD5
  - Produced a counterfeit digital signature that appears to have originated from Microsoft

Hash function design basics

- A typical hash function involves three components in the design
  - Operation mode
  - Compression function structure
  - Confusion-diffusion operations

Merkle-Damgard construction



- PB: padding block

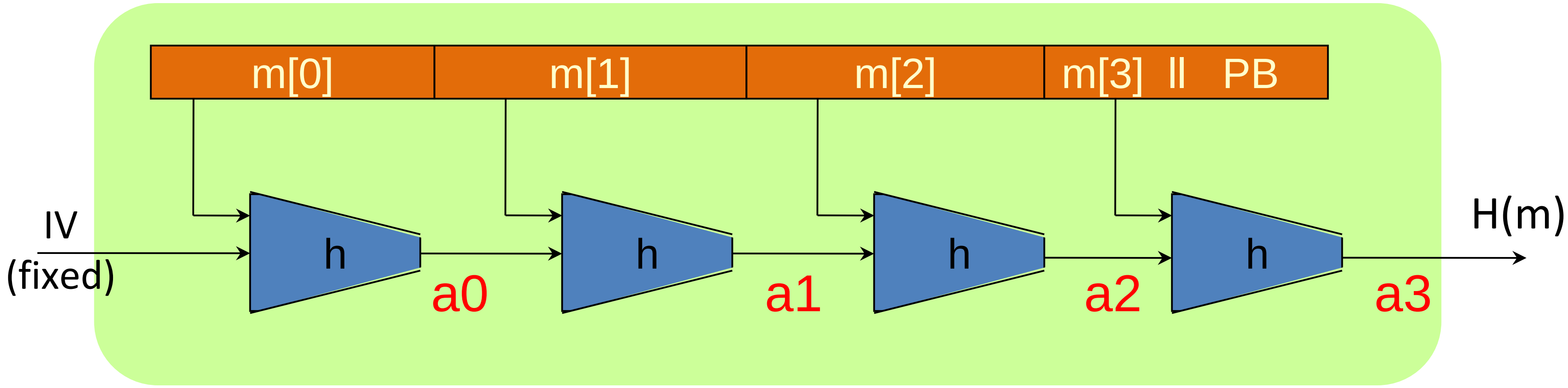
1000…0 ll msg len

64 bits

- **Theorem**: If the compression is collision-resistant, then the hash function is collision resistant

Assume m ≠ m' and H(m) = H(m')

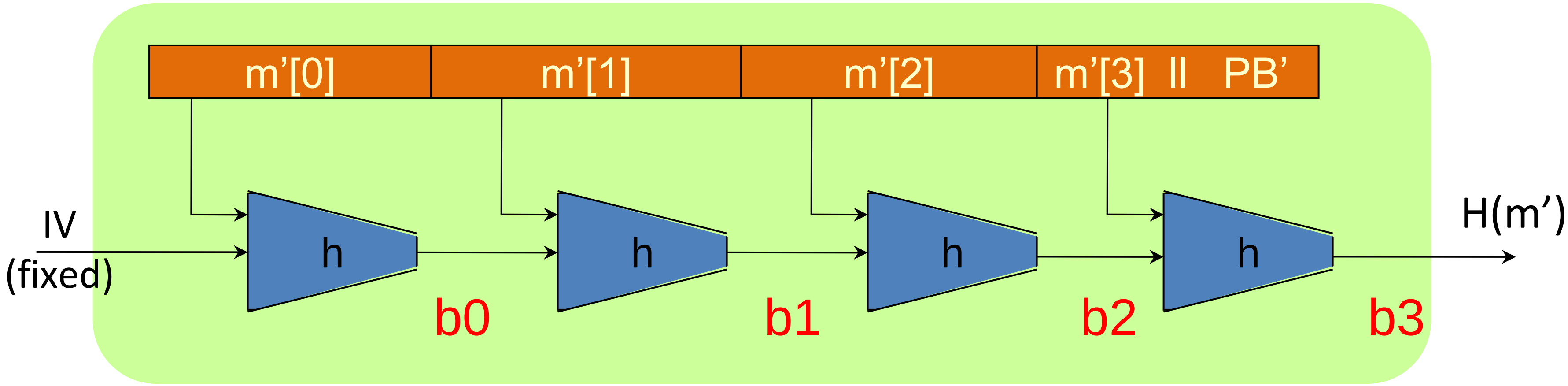
Proof of Merkle-Damgard theorem



Proof (sketch)

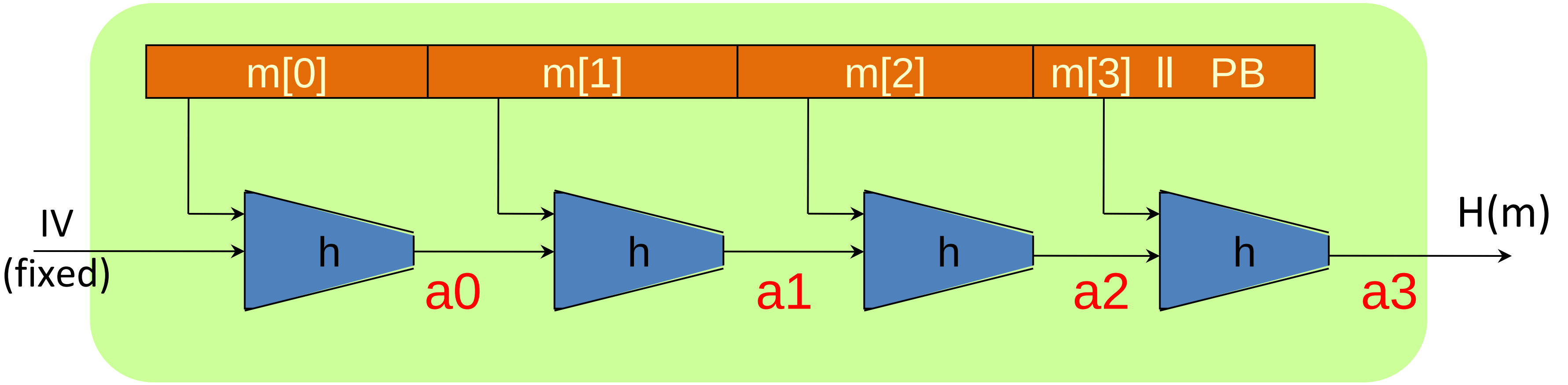
Case 1: m[3] ≠ m'[3]

Then h(a2,m[3]) ≠ h(b2,m'[3])



Assume m ≠ m’ and a3 = b3

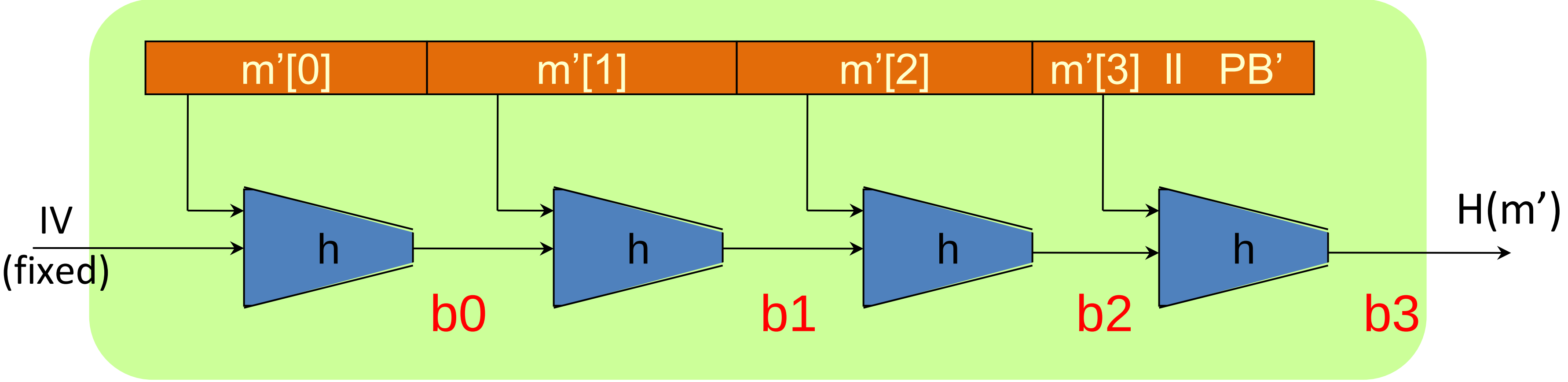
Proof of Merkle-Damgard theorem



Proof (sketch)

Case 2: m[3] = m’[3]

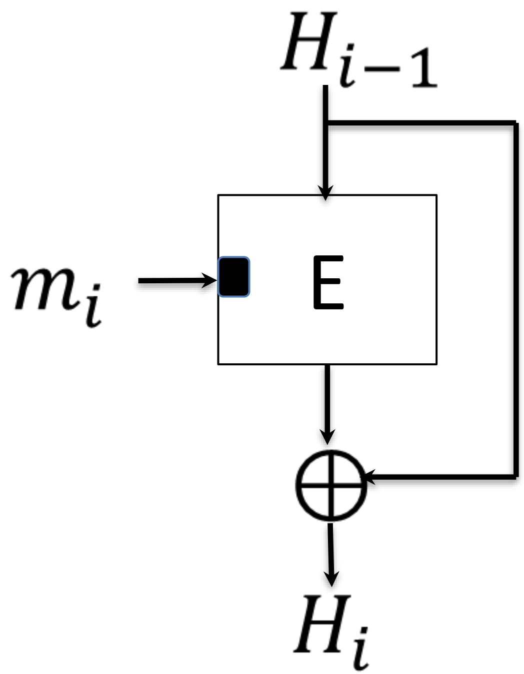
Then a2 = b2



a0 = b0 a1 = b1 a2 = b2 a3 = b3

Compression functions

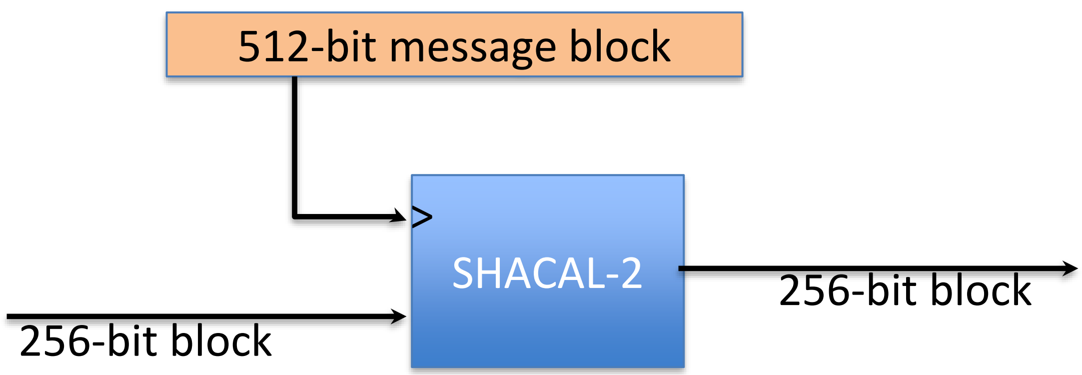
Davies-Meyer (used in MD5, SHA-1, SHA-2)



- E is a block cipher
- Use message as key
- **h(H,m)**=E(m,H) ⊕H
- Compression
  - Input size: key size + block size
  - Output size: block size

An example: SHA-256

- Merkle-Damgard function
- Davies-Meyer compression function
- Block cipher: SHACAL-2



Applications of Hash

- Digital signature
- Data integrity (collision or pre-image resistance)
  - Example: Checksum for downloading software
- Random number generator
- Data privacy
  - Protect plain password
- Commitment scheme
- Mining cryptocurrency!

Password authentication

Bob

Login: bob/1234

Login success

Compare password against Database

Username: password
Alice: 19890102
Bob: 1234
Charlie: cryfield204

Bob

Login: bob/1234

Login success

Compute hash of the password and compare it against Database

Username: H(pw, salt)	**Salt**
Alice: 8ced834745…	8FA
Bob: ff4ed0bd13d2…	E9A
Charlie: 32cbba4a1…	48C

Sadly, real-world security

- 2012-09-25: IEEE suffered a data breach
  - 100,000 Plaintext passwords were leaked.
- 2012-07-12: Yahoo Voices database breached
  - Half a million plaintext passwords leaked
- 2012-06-07: LinkedIn user database breached.
  - 6.5 million Unsalted hashes dumped online.
- 2010-12-22: Gawker Media accounts hacked
  - 1.3 million plaintext passwords were leaked

Caveat

- Hashing plaintext password with salt is only a best practice
  - It makes it difficult for the attacker to recover the password, but not impossible
- Dictionary attack
  - Given H(pw, salt) and salt
  - An attacker can exhaustively try out all passwords
  - The attack is feasible because passwords have low entropy