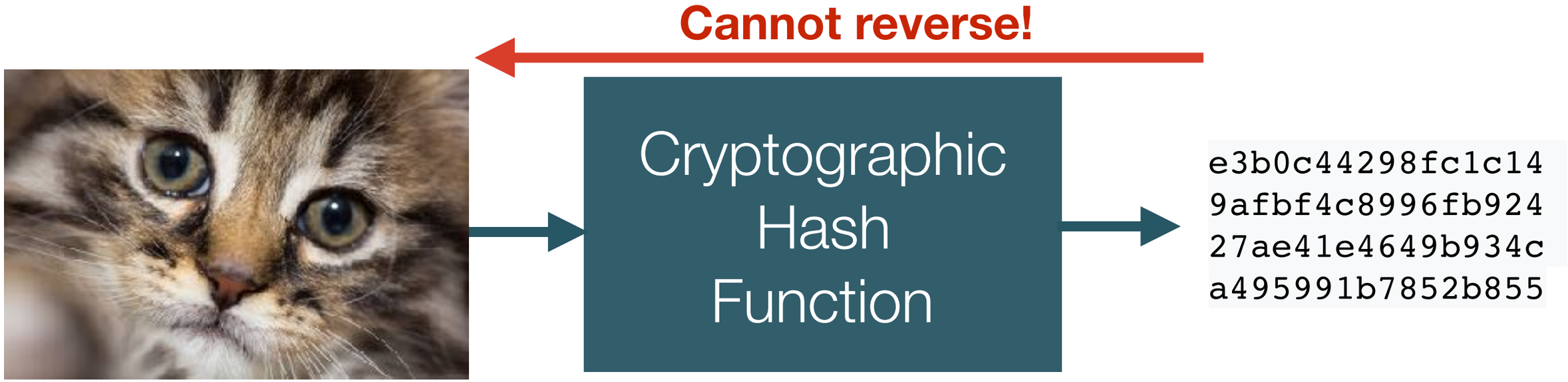
CS 465
Computer Security

Cryptographic Hash Functions

Last Updated: Sep 19, 2017

Message Digests



- Input to a hash function is called a pre-image
- Output of a hash function is a message digest *d*, also known as hash codes or hash values
- Cryptographic hash functions are ONE-way

Properties of a Hash Function (H)

1. H can be applied to a block of data of any size
2. H produces a fixed-length output
3. H(***x***) is relatively easy to compute for any given ***x***
4. For any given value ***h***, it is computationally infeasible to find ***x*** such that H(***x***) = ***h***

   (one-way)

5. For any given block ***x***, it is computationally infeasible to find ***y*** ≠ ***x*** with H(***y***) = H(***x***)

   (weak collision resistance)

6. It is computationally infeasible to find any pair (***x***, ***y***) such that H(***x***) = H(***y***)

   (strong collision resistance)

Attacks on Hash Functions

- First pre-image attack
  - Given a digest ***d***, find a message ***m*** such that H(***m***) = ***d***
    - Think property #4

- Second pre-image attack
  - Given a message m_1, find a different message m_2 such that H(m_2) = H(m_1)
    - Cost: 2^n where n = # of digest bits
    - Think property #5

- Collision attack
  - Find any two messages, m_1 and m_2, such that H(m_1) = H(m_2)
    - Birthday Attack
      - Given ***n***-bit digest, birthday attack says that we'll find a match after $2^{n/2}$ attempts
    - Think property #6

253 people in a room

- Odds are good that one of them shares a birthday with you

23 people in a room

- Odds are good that two people share a birthday

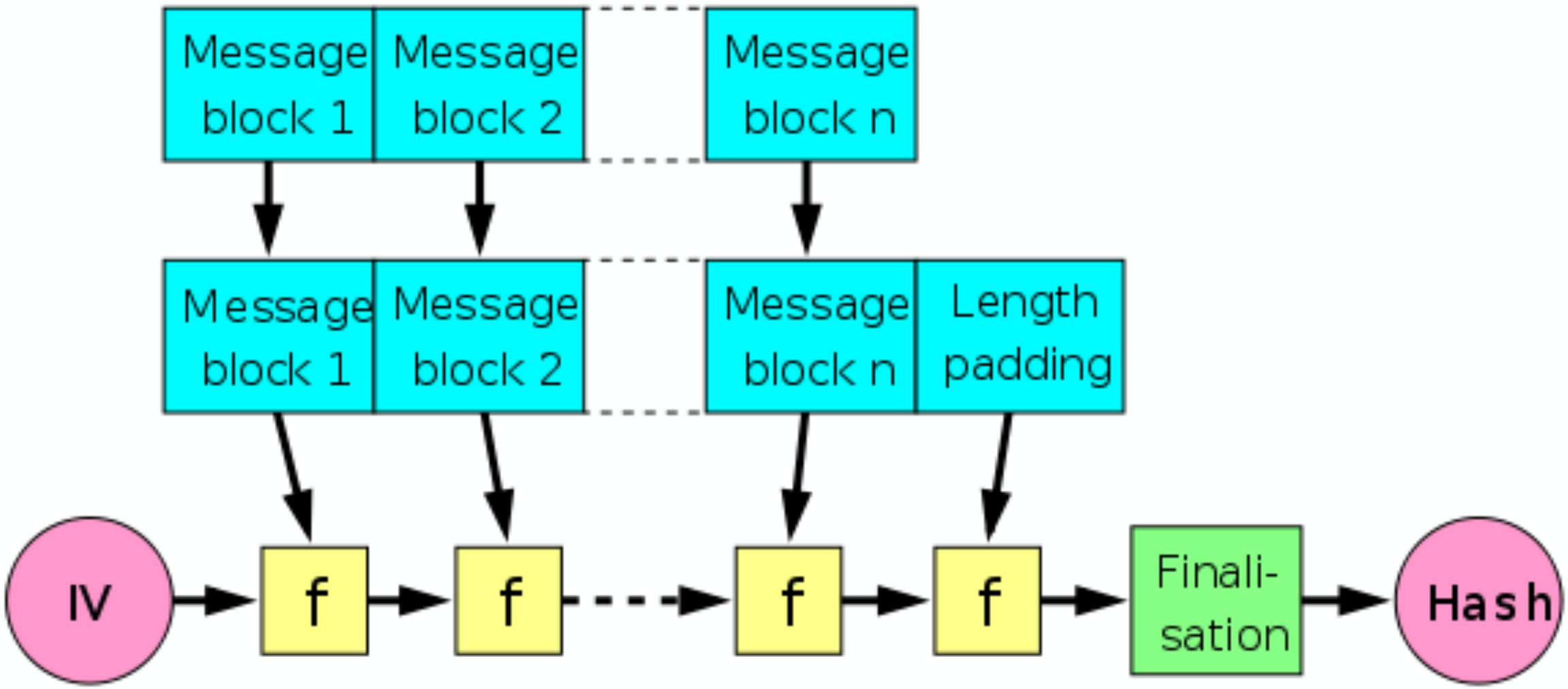
Useful Applications of Hashes

- Human-readable method to compare/verify data
  - File downloads
  - Before Learning Suite, we used to have students email in a hash of their projects
- Chaining events together – blockchain (think Bitcoin)
- Digital signatures and message authentication codes
- Fundamental building block of many secure protocols
  - Schneier (Secrets and Lies) – “They are probably the single most useful tool in a cryptographer’s toolbox”

SHA-1 Hash Function

- The following diagrams illustrates how SHA-1 is implemented
- SHA-1 is implemented using the Merkle-Damgård construction
  - See http://en.wikipedia.org/wiki/Merkle–Damgård_construction
- It is important to understand the details of this implementation in order to understand the MAC attack discussed later and the MAC attack lab

Merkle-Damgård Construction



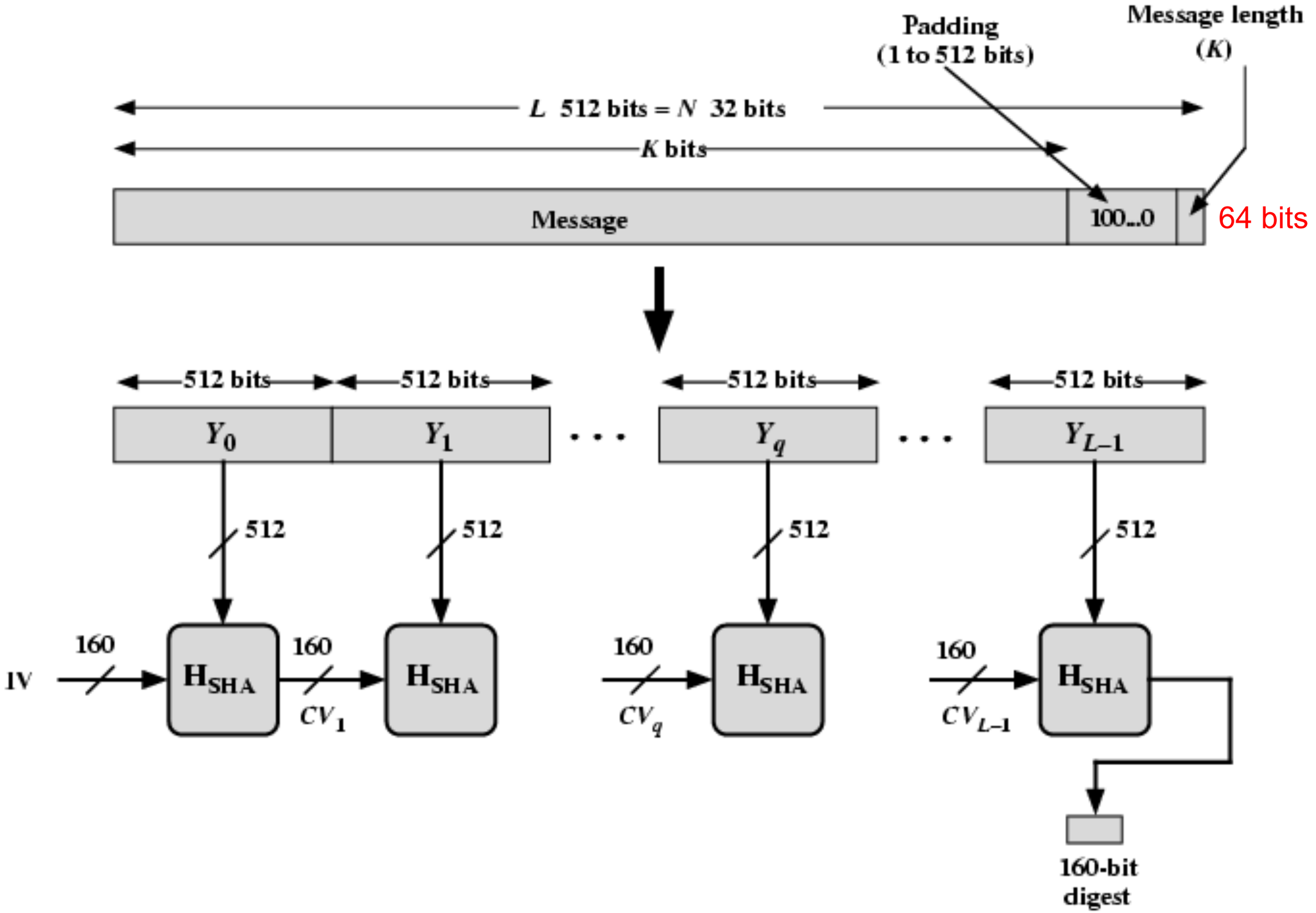
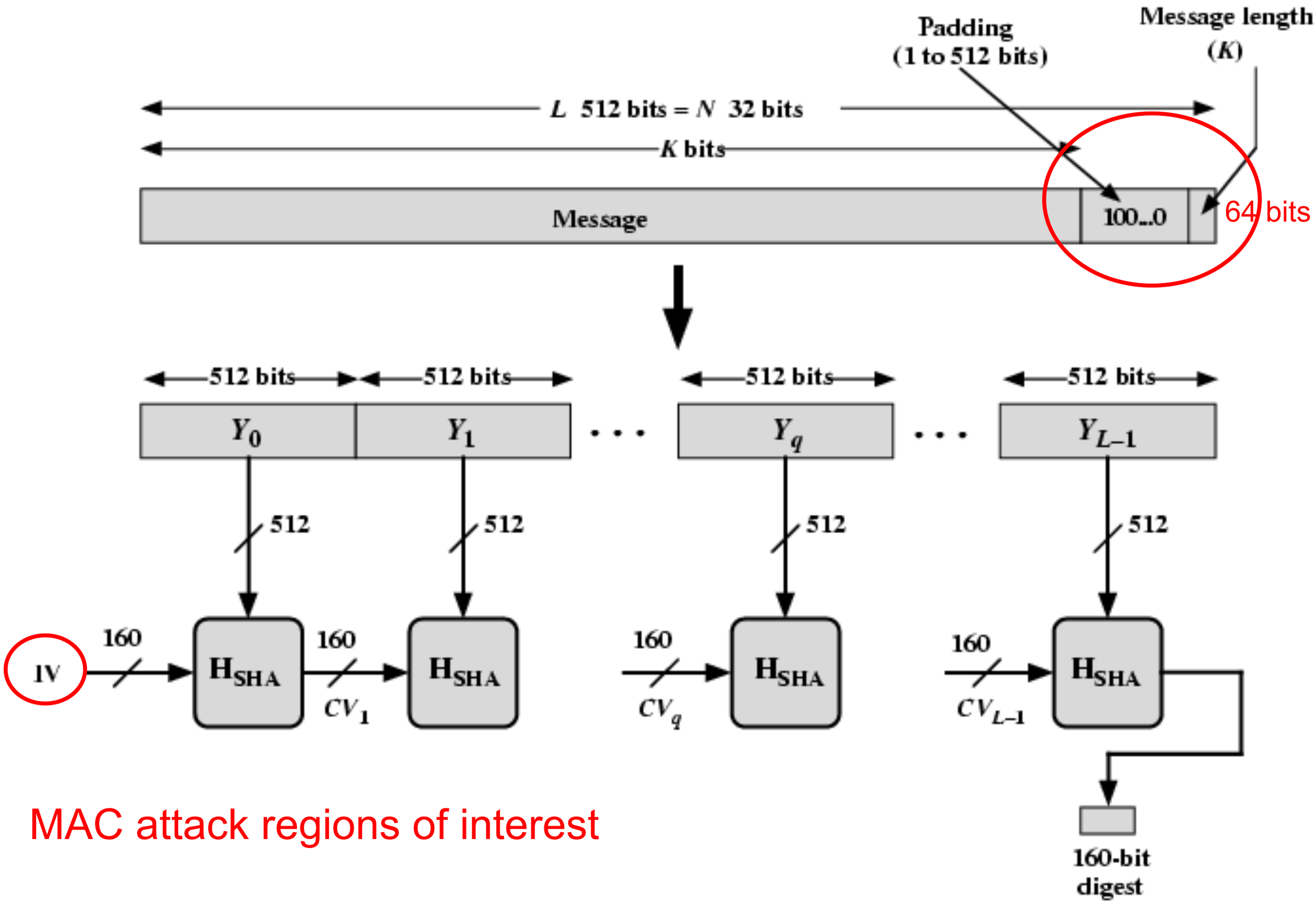


Figure 3.4 Message Digest Generation Using SHA-1

Computing SHA-1 padding

- See 5.1.1 of the SHA-1 spec
  - https://cs465.internet.byu.edu/static/pubs/fips180-3_final.pdf



MAC attack regions of interest

Figure 3.4 Message Digest Generation Using SHA-1

MD5 and SHA

- MD5
  - Completely broken – collisions found
- SHA-0
  - Completely broken – collisions found
- SHA-1
  - Weaknesses discovered, not recommended for new apps after 2010
- SHA-2
  - ok to use, no known flaws

MD5 and SHA

- SHA-3
  - Subset of Keccak (pronounce ketchak)
  - New government standard in 2015 based on a competition (like AES)
  - Completely different construction than prior SHA variants
  - Does not use Merkle-Damgard construction – uses a sponge construction
  - A defense in the event that SHA2 collapses
- See https://en.wikipedia.org/wiki/SHA-3 comparison chart

Other resources

- See articles about SHA-1 collisions
  - 2005 – Chinese researchers discovered first SHA-1 collisions faster than brute force (2^69)
  - 2017 – Google can generate PDF files with the same SHA-1 hash
    - 100,000 times faster than brute force