CSE 484 / CSE M 584: Finish Hash Functions + MACs

Fall 2023

Franziska (Franzi) Roesner
franzi@cs

UW Instruction Team: David Kohlbrenner, Yoshi Kohno, Franziska Roesner. Thanks to Dan Boneh, Dieter Gollmann, Dan Halperin, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials ...

Announcements

- Today:
  - Finish MACs + hash functions
  - Guest lecture!

Hash Functions Review

- Map large domain to small range (e.g., range of all 160- or 256-bit values)
- Properties of cryptographically secure hash functions:
  - **One-wayness:** Given a point in the range (that was computed as the hash of a random domain element), hard to find a preimage
  - **Collision Resistance:** Hard to find two distinct inputs that map to same output
  - **Weak Collision Resistance:** Given a point in the domain and its hash in the range, hard to find a new domain element that maps to the same range element

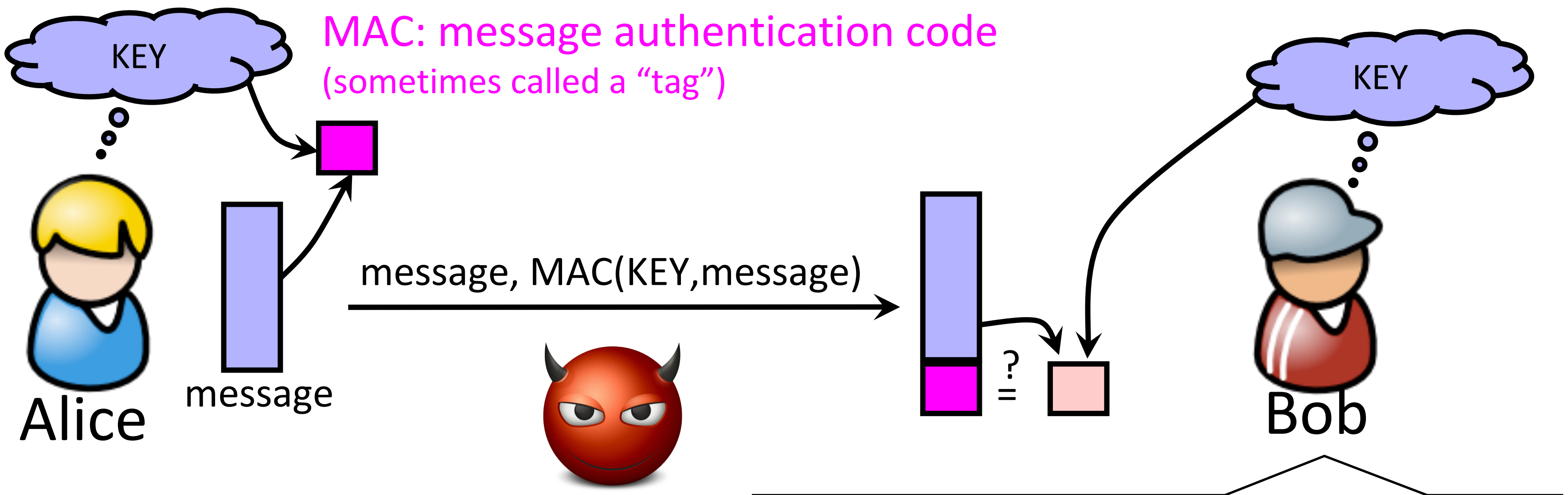
Which Property Do We Need?

One-wayness, Collision Resistance, Weak CR?

- UNIX passwords stored as hash(password)
  - **One-wayness:** hard to recover the/a valid password
- Integrity of software distribution
  - **Weak collision resistance**
  - But software images are not really random… may need **full collision resistance** if considering malicious developers

Recall: Achieving Integrity

Message authentication schemes: A tool for protecting integrity.


KEY
MAC: message authentication code
(sometimes called a "tag")
KEY
Alice
message
message, MAC(KEY,message)
?
=
Bob

Recomputes MAC and verifies whether it is equal to the MAC attached to the message

Integrity and authentication: only someone who knows KEY can compute correct MAC for a given message.

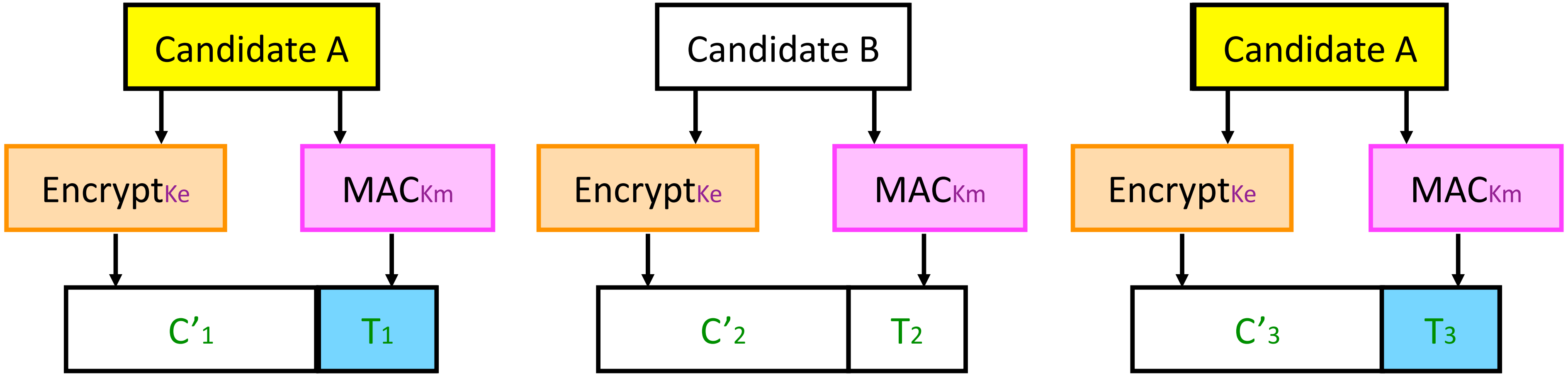
MAC with SHA3

- SHA3(Key || Message)
- Nice and simple ☺
- Previous hash functions couldn’t quite be used in this way (see: length extension attack)
  - HMAC construction (FYI)

- Why not encryption? (Historical reasons)
  - Hashing is faster than block ciphers in software
  - Can easily replace one hash function with another
  - There used to be US export restrictions on encryption

Authenticated Encryption

- What if we want both privacy and integrity?
- Natural approach: combine encryption scheme and a MAC.
- But be careful!
  - Obvious approach: Encrypt-and-MAC
  - Problem: MAC is deterministic! same plaintext → same MAC

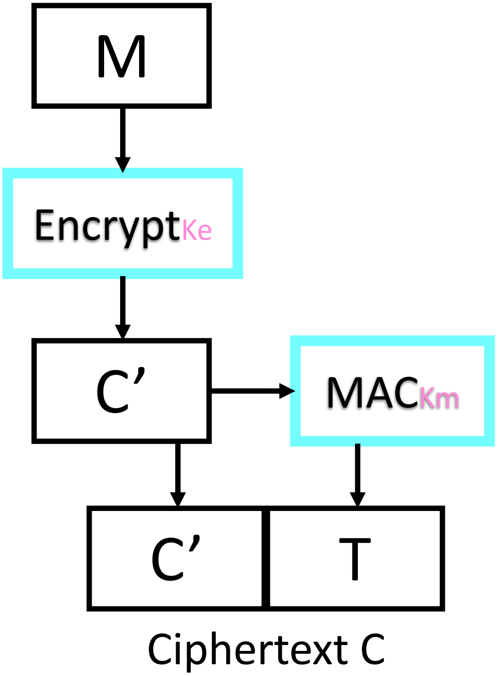


Authenticated Encryption

- Instead:

  Encrypt *then* MAC.

- (Not as good: MAC-then-Encrypt)



Encrypt-then-MAC