Week 6: Hashing

Dr. Qublai K. Ali Mirza

University of Gloucestershire

qalimirza@glos.ac.uk

Overview

Confidentiality, Integrity, Availability

- One of the fundamental tenets of cybersecurity
- Confidentiality: Ensuring only authorised personal has access to resource
- Integrity: Assurance that data has not been tampered with
- Availability: Proper functioning of systems after attack

CIA Triangle

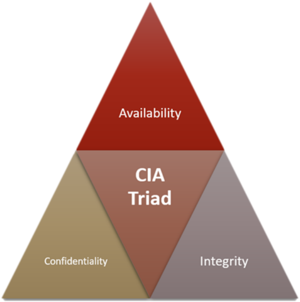

Figure: CIA Triangle, from [3]

Active attacks

- Masquerade
  - Attack through false pretences
- Replay
  - Retransmission of previously captured data
- Message modification
  - Illegal alteration of some or all of a legitimate message
- Denial of Service
  - Rendering some or all of the communication infrastructure needed for data transmission

Authentication mechanisms

- When looking at *any* authentication mechanism, it needs to consist of :
  - A function that produces an *authenticator*
  - A higher-level protocol that uses it to verify message authenticity
- Different authentication approaches can be grouped into:
  - Message Authentication Code (*MAC*)
  - Hash functions

Overview

- Uses a secret key K to generate an fixed-length authenticator
- Calculated as a function of the message M and K using:

$$MAC = C_K(M)$$

- Sent along with the original message M to the receiver
- M deemed authentic if the receiver obtains the same MAC value

UNIVERSITY OF GLOUCESTERSHIRE

Operation overview

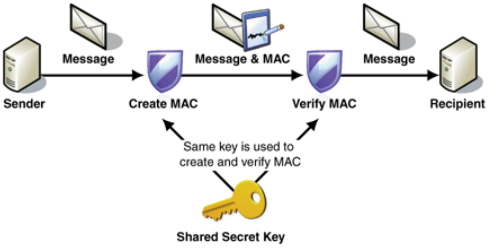

Figure: MAC operation overview, from [5]

Characteristics

- One direction in nature
- K is known by both the sender and receiver
- Provide assurance that message content has not been altered
- Less vulnerable against attack compared to encryption
- Can be used to authenticate *both* text and binary data

UNIVERSITY OF GLOUCESTERSHIRE

Data Authentication Algorithm (DAA)

- Features the use of *DES*
- Cipher Block Chaining (CBC) mode used
- M broken into 64-bit continuous blocks before applying DES
- Either the entire calculation result or the leftmost N bits $(16 \leq N \leq 64)$ used as MAC

Data Authentication Algorithm (DAA)

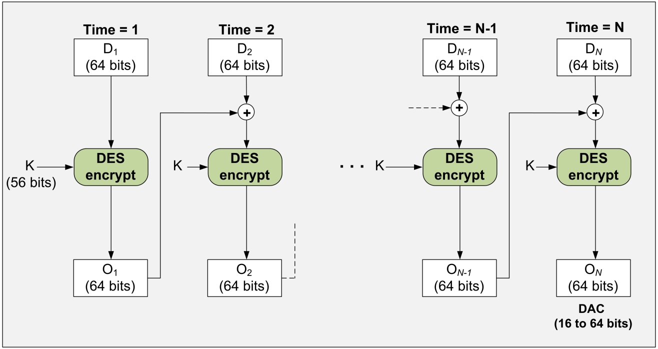

Figure: DAC operation, adapted from [6]

Overview

- Similar to MAC
- Hash code calculated as a function of the *bits* in the message M using:
$$h = H(M)$$
- Change in either a single bit or bits results in a complete change in h
- Used when creating a digital signature

Keyed-Hash MAC (HMAC)

- FIPS standard for creating MAC from hash function h
- Different hash functions can be used to calculate h
- It is calculated using [4]:

$$HMAC(K, m) = H((K' \oplus opad)||H((K' \oplus ipad)||m))$$

HMAC

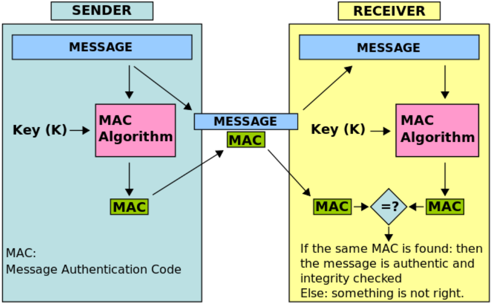

Figure: HMAC operation, from [1]

Characteristics

- Can be used on message M of any size
- Produces a fixed-length output h
- One-way in nature
- Strong collision resistance, meaning it is impossible to find x and y such that:
$$H(y) = H(x)$$
- Computationally inexpensive

Hash calculation

- The input (e.g., file, executable) is converted into a sequence of blocks of n-bits
- The most basic hash calculation involves performing bitwise `XOR` operation for every block using:

$$C_i = b_{i1} \oplus b_{i2} \oplus \ldots \oplus b_{im}$$

- Not really secure since the message can be manipulated without changing the resulting hash (aka. *birthday attacks*)

Birthday Attacks

- Based on *Birthday Paradox*, which states that
  - In a randomly chosen n number of people, there exists at least *two* people with the same birthday
- Used to break hash by:
  - opponent generates $2^{m/2}$ variations of a valid message all with essentially the same meaning
  - opponent also generates $2^{m/2}$ variations of a desired fraudulent message
  - two sets of messages are compared to find pair with same hash (probability > 0.5 by birthday paradox)
  - have user sign the valid message, then substitute the forgery which will have a valid signature

Brute force

- Different approaches taken for hash and MAC
- Hash
  - Depends on the hash code *length*
  - For a hash code of length n, it requires $2^{n/2}$ bit combinations to find a hash collision
- MAC (Message Authentication Codes)
  - Requires knowledge of message-MAC pairs
  - For a key length of k bits and MAC length of n bits, the amount of effort required: $\min(2^k, 2^n)$

Cryptanalysis

- Exploit structure of the hashing algorithm used
- There exists a number of analytic attacks on iterated hash functions
- Main goal: To find a collision that matches the target hash/MAC values

Bringing it together

- Today we looked at MAC and Hashes
- We also looked at the inner workings of both as well
- Next week: *Asymmetric encryption*

Post-sessional work

- Using the article by [2] (available on *Moodle*) as a starting point, write a critical review on how asymmetric encryption is used to protect sensitive data
- Upload your completed work to *Moodle* before next *Monday*.

References I

- CN Blogs. http://pic002.cnblogs.com/images/2012/427162/2012072911095143.png. Accessed: 20-10-2017. 2012.
- P Fanfara, E Danková, and M Dufala. "Usage of asymmetric encryption algorithms to enhance the security of sensitive data in secure communication". In: *Applied Machine Intelligence and Informatics (SAMI), 2012 IEEE 10th International Symposium on*. IEEE. 2012, pp. 213–217.
- Infosec Institute. *CIA Triad*. http://resources.infosecinstitute.com/cia-triad/. Accessed: 20-10-2017. 2017.

UNIVERSITY OF GLOUCESTERSHIRE

References II

Hugo Krawczyk, Ran Canetti, and Mihir Bellare. “HMAC: Keyed-hashing for message authentication”. In: (1997).

Microsoft. *Data Origin Authentication*. https://msdn.microsoft.com/en-us/library/ff648434.aspx. Accessed: 20-10-2017. 2005.

William Stallings. *Cryptography and network security: principles and practices*. Pearson Education India, 2006.

Q & A