Cryptography

Message Authentication Code (MAC)

M S Vilkhu

30 Nov 2024(C1/C3/C5)

Message Authentication Code (MAC)

Topic

12.1 Message Authentication Requirements

12.2 Message Authentication Functions

Message Encryption

Message Authentication Code

12.3 Requirements for Message Authentication Codes

12.4 Security of MACs

Brute-Force Attacks

Cryptanalysis

Message Authentication Code (MAC)

- One of the most **fascinating and complex areas** of cryptography is that **of message authentication** and the related area of **digital signatures**.
- It would be impossible, in anything less than book length, to exhaust all the cryptographic functions and protocols that have been proposed or implemented for message authentication and digital signatures.
- Instead, the purpose of this chapter and the next is to provide a broad overview of the subject and to develop a **systematic means of describing the various approaches**.

Message Authentication Code (MAC)

- Look at
- **Requirement** of Message authentication & Digital Signature
- Message authentication known as message authentication code (MAC).
- Security consideration of MAC

Message authentication Requirements

- In the context of **communications across a network**, the following attacks can be identified towards Message.

1. **Disclosure**: Release of message contents to any person or process not possessing the appropriate cryptographic key.

2. **Traffic analysis**: Discovery of the **pattern** of traffic between parties. In a connection-oriented application, the **frequency and duration of connections** could be determined. In either a connection-oriented or connectionless environment, the **number and length** of messages between parties could be determined.

3. **Masquerade**: **Insertion of messages** into the network from a **fraudulent source**. This includes the creation of messages by an opponent that are purported to come from an authorized entity. Also included are **fraudulent acknowledgments** of message receipt or nonreceipt by someone other than the message recipient.

Message authentication Requirements

- In the context of **communications across a network**, the following attacks can be identified.

1. **Disclosure**: Release of message contents to any person or process not possessing the appropriate cryptographic key.

2. **Traffic analysis**: Discovery of the **pattern** of traffic between parties. In a connection-oriented application, the **frequency and duration of connections** could be determined. In either a connection-oriented or connectionless environment, the **number and length** of messages between parties could be determined.

1 & 2 - in the realm of **message confidentiality.**

Message authentication Requirements

- In the context of **communications across a network**, the following attacks can be identified.

3. **Masquerade**: **Insertion of messages** into the network from a **fraudulent source**. This includes the creation of messages by an opponent that are purported to come from an authorized entity. Also included are **fraudulent acknowledgments** of message receipt or nonreceipt by someone other than the message recipient.

4**. Content modification**: Changes to the contents of a message, including inser- tion, deletion, transposition, and modification.

5. **Sequence modification**: Any modification to a sequence of messages between parties, including insertion, deletion, and reordering.

6. **Timing modification:** Delay or replay of messages.

In a **connection-oriented application**, an entire session or sequence of messages could be a **replay** of some previous valid session, or individual messages in the **sequence** could be delayed or replayed.

In a **connectionless application**, an individual message (e.g., datagram) could be **delayed** or **replayed**.

3 through 6 in the foregoing list are generally regarded as **message authentication**

Message authentication Requirements

- In the context of **communications across a network**, the following attacks can be identified.

7. **Source repudiation**: **Denial** of **transmission** of message by source.

8. **Destination repudiation**: **Denial** of **receipt** of message by destination.

7 come under the heading of **digital signatures**.

8 may require a combination of the use of **digital signatures and a protocol designed** to counter this attack.

Message authentication Requirements

Summary

1 & 2 - in the realm of **message confidentiality.**

3 through 6 in the foregoing list are generally regarded as **message authentication**.

7 come under the heading of **digital signatures**.

Generally, a **digital signature** technique will also **counter** some or all of the attacks listed under items **(3) through (6).**

8 may require a combination of the use of **digital signatures and a protocol designed** to counter this attack.

Message authentication Requirements

- **message authentication**
- is a procedure to **verify that received messages** come from the alleged source and have **not been altered**.
- **may also verify sequencing and timeliness**.

- A **digital signature** is an **authentication technique** that also includes **measures to counter repudiation** by the source.

Message authentication Function

- Any **message authentication or digital signature** mechanism has **two levels** of functionality.

1. At the **lower level,** there must be some sort of function that **produces an authenticator**: a value to be **used to authenticate a message**.
2. This lower-level function is then used as a **primitive** in a **higher-level authentication** protocol that enables a **receiver to verify the authenticity of a message**.

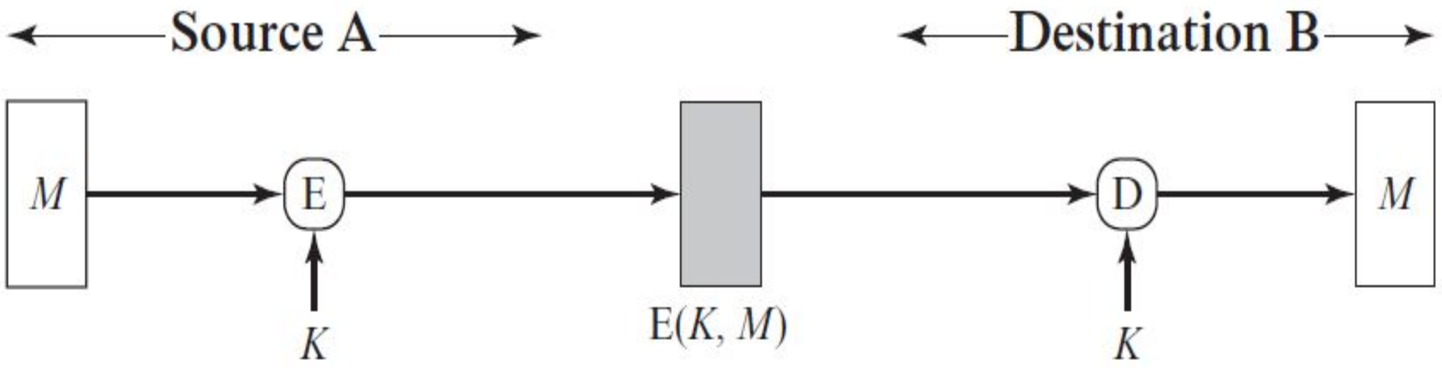
Message authentication Function

- **Functions that produce** the authenticator – 3 type as follows

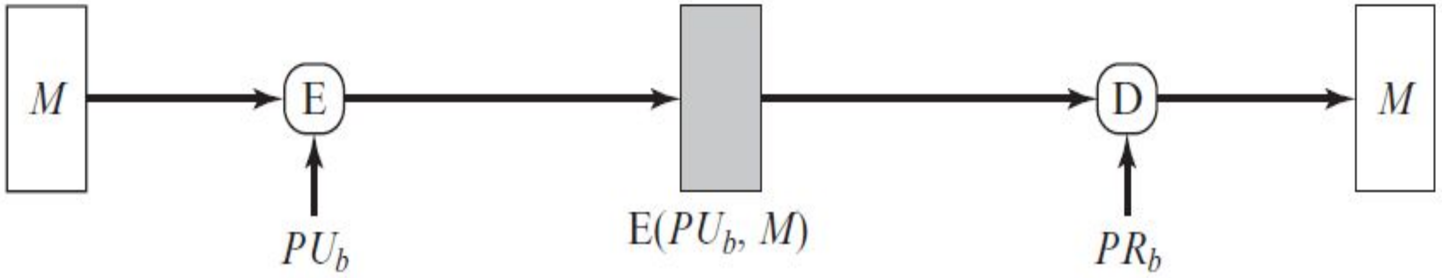
1. **Hash function**: A function that maps a message of any length into a fixed-length hash value, which **serves as the authenticator**
2. **Message encryption**: The ciphertext of the **entire message** serves as its **authenticator**
    1. Symmetric key and Asymmetric key encryption – both provide measure of authentication.
3. **Message authentication code (MAC):** A **function** of the message and a **secret key** that produces a fixed-length value that serves as the authenticator

Message authentication Function

- Fig (a) If no other party knows the key, then confidentiality is provided: No other party can recover the plaintext of the message
- B knows A possess the Key K and can generate the message
- Fig (b) public-key encryption provides **confidentiality but not authentication**



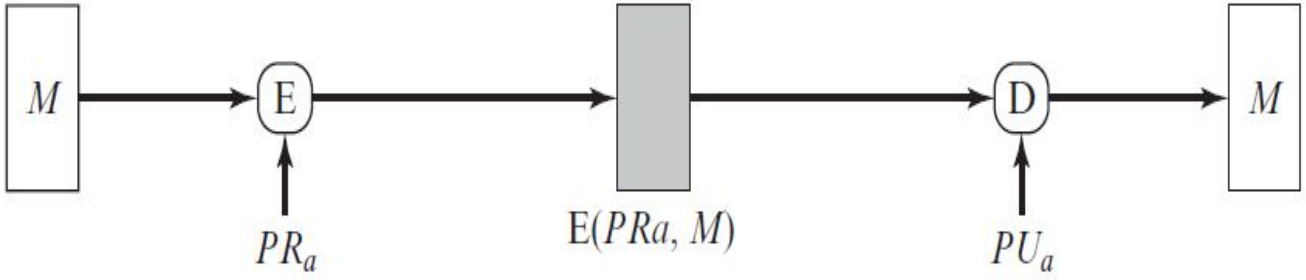
(a) Symmetric encryption: confidentiality and authentication



(b) Public-key encryption: confidentiality

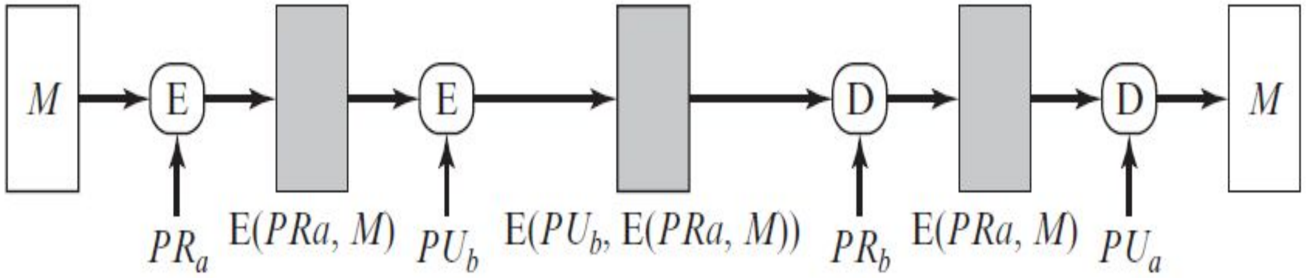
Message authentication Function

- Fig (c) To provide authentication, A uses its **private key to encrypt the message, and B uses A's public key to decrypt**



(c) Public-key encryption: authentication and signature

- Fig (d) To provide both confidentiality and authentication,



(d) Public-key encryption: confidentiality, authentication, and signature

Message authentication Function

- we may say that **symmetric encryption provides authentication** as well as **confidentiality**. Is this true?
- However, this flat statement **needs to be qualified**.
- Consider exactly what is happening at B. Given a decryption function D and a secret key K, the destination will accept any input X and produce output Y = D(K, X).
- If X is the ciphertext of a **legitimate message M** produced by the corresponding encryption function, then **Y is some plaintext** message M. Otherwise, **Y will likely be a meaningless sequence** of bits.

Message authentication Function

- There may need to be some automa**ted means of determining at B** **whether Y is legitimate plaintext** and therefore must have come from A.
- The implications of the **line of reasoning** in the preceding paragraph are profound from the **point of view of authentication**.
- Suppose the **message M** can be **any arbitrary bit pattern.** In that case, **there is no way to determine automatically, at the destination, whether an incoming message is the ciphertext of a legitimate message**.
- This conclusion is incontrovertible: If **M can be any bit pattern**, then regardless of the value of X, the value Y = D(K,X) is **some bit pattern** and therefore must be **accepted as authentic plaintext.**

Message authentication Function

- **PUBLIC-KEY ENCRYPTION .**
- The straightforward use of public-key encryption provides **confidentiality but not authentication**.
- The source (A) uses the public key $P_U b$ of the destination (**B) to encrypt M**.
- Because only B has the corresponding private key $P_R b$, **only B can decrypt the message**.
- This scheme **provides no authentication**, because **any opponent could also use B's public key to encrypt a message** and claim to be A.
- To provide authentication, A uses its **private key** to encrypt the message, and B uses A's public key to decrypt (Figure c).

Message authentication Function

- **PUBLIC-KEY ENCRYPTION .**
- The message must have come **from A because A is the only party that possesses PRa** and therefore the only party with the information necessary to construct ciphertext that can be decrypted with PUa.
- Again, the **same reasoning as before applies**: There **must be some internal structure** to the **plaintext** so that the **receiver can distinguish between** **well-formed plaintext and random bits**.

- To provide both **confidentiality and authentication**,
- A can **encrypt M** first using its **private key,** => **the digital signature**, and
- then using **B's public key**, => **confidentiality** (Figure d).
- **disadvantage**
- The **disadvantage** of this approach is that the public-key algorithm, which is complex, must be exercised **four times rather** than two in each communication.

Message Authentication Code

- An **alternative authentication** technique involves the use of a **secret key** to generate a **small fixed-size block of data**, known as a **cryptographic checksum or MAC**, that is **appended** to the message.
- This technique assumes that two communicating parties, say A and B, share a common secret key K. When A has a message to send to B, it calculates the MAC as a **function of the message and the key**:

$$MAC = C(K, M)$$

- **where**

  M = input message

  C = MAC function

  K = shared secret key

  MAC = message authentication code

Message Authentication Code

- The **message** plus **MAC** are transmitted to the intended recipient.
- The **recipient** performs the **same calculation** on the received message, using the **same secret key**, to generate a new MAC.
- The **received MAC** is compared to the **calculated MAC** . If we assume that only the receiver and the sender know the identity of the secret key, and if the received MAC matches the calculated MAC, then

1. The receiver is assured that the message has **not been altered**.

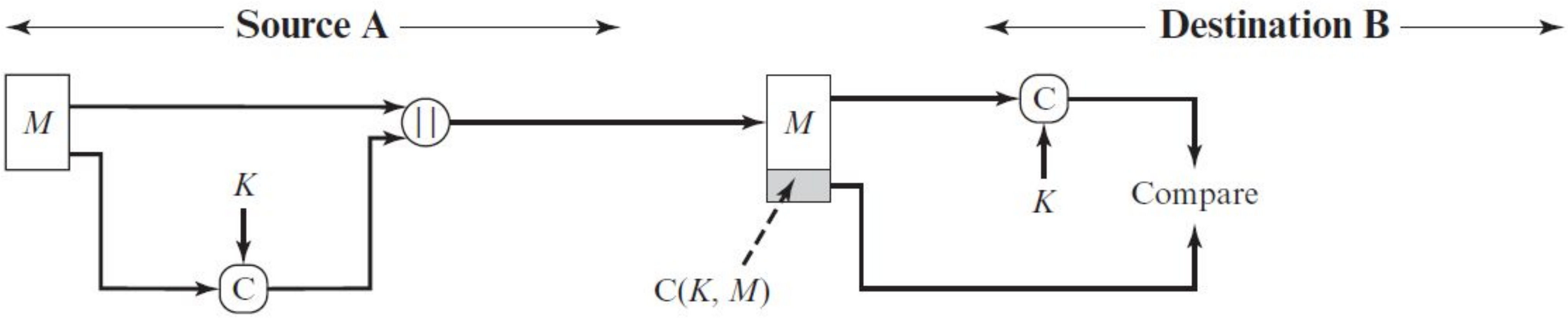
If an **attacker alters** the message but **does not alter the MAC**, then the receiver's calculation of the **MAC will differ** from the **received MAC**.

Because the **attacker is assumed** **not to know the secret key**, the attacker **cannot alter the MAC** to correspond to the **alterations in the message**.

2. The **receiver is assured** that the message is from the alleged sender. Because **no one else knows the secret key,** no one else could prepare a message with a proper MAC.

Message Authentication Code

3. If the **message includes a sequence number** (such as is used with HDLC, X.25, 3. and TCP), then the **receiver can be assured of the proper sequence** because an attacker cannot successfully alter the sequence number.



(a) Message authentication

Message Authentication Code

- A **MAC function is similar to encryption**.
- **One difference** is that the MAC algorithm **need not be reversible**, as it must be for decryption.
- In general, the **MAC function** is a **many-to-one function**.
- The domain of the function **consists of messages of some arbitrary length**, whereas the range consists of **all possible MACs and all possible keys**.
- If an n-bit MAC is used, then there are 2^n possible MACs, whereas there are N possible messages with $N >> 2^n$. Furthermore, with a k-bit key, there are 2^k possible keys.

Security of MACs

- Brute Force attacks
- Cryptoanalysis

Security of MACs

- **Brute Force attacks**
- A **brute-force attack on a MAC** is a **more difficult** undertaking than a brute-force attack on a **hash function** because it requires **known message-tag pairs.**
- To attack a hash code, we can proceed in the following way. Given a fixed message x with n-bit hash code h = H(x), a brute-force method of finding a collision is to pick a random bit string y and check if H(y) = H(x). The attacker can do this repeatedly offline.
- Whether an off-line attack can be used on a MAC algorithm depends on the relative size of the key and the tag.

Known message-tag pairs refer to a scenario where an attacker has access to a set of **messages along with their corresponding Message Authentication Codes** (MACs) or cryptographic tags. These pairs can be used as the basis for various attacks

Security of MACs

- **Brute Force attacks**
- **security property of a MAC algorithm**, which can be expressed as follows.
- **Computation resistance**: Given one or more text-MAC pairs [x_i, MAC(K, x_i)],
- it is **computationally infeasible** to compute any text-MAC pair [x, MAC(K, x)] for any new input $x \neq x_i$.

- There are **two lines of attack** possible:
- attack the **key space** and
- Attack **the MAC value**.

Security of MACs

- **Brute Force attacks**
- **security property of a MAC algorithm**, which can be expressed as follows.
- **Computation resistance**: Given one or more text-MAC pairs [x_i, MAC(K, x_i)],
- it is **computationally infeasible** to compute any text-MAC pair [x, MAC(K, x)]
- for any new input $x \neq x_i$.

- There are **two lines of attack** possible:
- attack the **key space** and
- Attack **the MAC value**.

Security of MACs

- **Brute Force attacks**
- attack the **key space**
- If an attacker can determine the **MAC key,** then it is possible to generate a **valid MAC value for any input x**.
- Suppose the **key size is k bits** and that the attacker has one **known text-tag pair.** Then the attacker can **compute the n-bit tag on the known text for all possible keys**.
- At least **one key is guaranteed** to produce the correct tag, namely, the valid key that was initially used to produce the known text-tag pair.
- This attack takes a level of **effort proportional** to 2^k

Security of MACs

- **Brute Force attacks**
- attack the **key space**
- because the **MAC is a many-to-one mapping**, there may be **other keys that** produce the **correct value**. => more than one key is found to produce the correct value, **additional text-tag pairs** must be tested.
- It can be shown that the level of **effort drops off rapidly** with each **additional text-MAC pair and that the overall level of effort is roughly 2^k** [MENE97].

Security of MACs

- **Brute Force attacks**
- Attack the MAC value.
- An **attacker** can also work on the tag **without attempting to recover the key**.
- Here, the objective is to **generate a valid tag** for a **given message**

or

- to **find a message** that matches a given tag.
- In either case, the **level of effort is comparable** to that for attacking the **one-way or weak collision-resistant property of a hash code**, or Y.
- In the case of the MAC, the attack **cannot be conducted** off line without further input; the **attacker will require chosen text-tag** pairs or **knowledge of the key**.

Security of MACs

- **Cryptoanalysis**
- As with encryption algorithms and hash functions, cryptanalytic attacks on MAC algorithms seek to exploit **some property of the algorithm** to perform some attack other than an exhaustive search.
- The way to **measure the resistance** of a MAC algorithm to cryptanalysis is **to compare its strength** to the **effort required for a brute- force attack.**
- That is, an **ideal MAC algorithm** will require a **cryptanalytic effort greater than or equal to the brute-force effort**.
- There is **much more variety in the structure of MACs** than in **hash functions**, so it is **difficult to generalize about the cryptanalysis of MACs**.
- Furthermore, **far less work has been done** on developing such attacks.

Thank You