Message Authentication Codes

CSS322: Security and Cryptography

Sirindhorn International Institute of Technology
Thammasat University

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Contents

Message Authentication Requirements and Functions

Authentication with Message Encryption

Authentication with Message Authentication Codes

Security of MACs

MAC Algorithms

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Attacks on Communications across Network

1. Disclosure: encryption
2. Traffic analysis: encryption
3. Masquerade: message authentication
4. Content modification: message authentication
5. Sequence modification: message authentication
6. Timing modification: message authentication
7. Source repudiation: digital signatures
8. Destination repudiation: digital signatures

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Message Authentication Functions

▶ Message authentication (and digital signature) mechanisms have two parts:
  1. Function that produces authenticator
  2. Protocol that enables receiver to verify authenticity

▶ Three types of authentication functions:
  1. Hash function
  2. Message encryption
  3. Message authentication code (MAC)

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Contents

Message Authentication Requirements and Functions

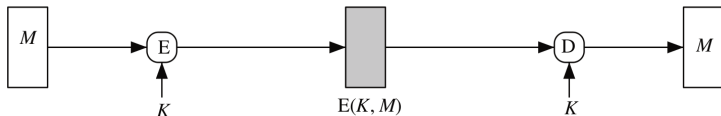Authentication with Message Encryption

Authentication with Message Authentication Codes

Security of MACs

MAC Algorithms

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Symmetric Encryption for Authentication

- ▶ Confidentiality: only B (and A) can recover plaintext
- ▶ Source Authentication: A is only other user with key; must have come from A
- ▶ Data Authentication: successfully decrypted; data has not been modified
- ▶ Assumption: decryptor can recognise correct plaintext

CSS322

Introduction

Functions
Auth. with
Encryption
Auth. with MAC
Security
Algorithms

Recognising Correct Plaintext

Example 1

B receives ciphertext (supposedly from A, using shared secret key K):

DPNFCTEJLYONCJAEZRCLASJTDQFY

B decrypts with key K to obtain plaintext:

SECURITYANDCRYPTOGRAPHYISFUN

- ▶ Was the plaintext encrypted with key K (and hence sent by A)?
- ▶ Is the ciphertext received the same as the ciphertext sent by A?

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Recognising Correct Plaintext

Example 2

B receives ciphertext (supposedly from A, using shared secret key K):

QEFPFPQEBTOLKDJBPPXDBPLOOVX

B decrypts with key K to obtain plaintext:

FTUEUEFTQIDAZSYQEEMSQEADDKM

- ▶ Was the plaintext encrypted with key K (and hence sent by A)?
- ▶ Is the ciphertext received the same as the ciphertext sent by A?

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Recognising Correct Plaintext

Example 3

B receives ciphertext (supposedly from A, using shared secret key K):

0110100110101101010110111000010

B decrypts with key K to obtain plaintext:

0101110100001101001010100101110

- ▶ Was the plaintext encrypted with key K (and hence sent by A)?
- ▶ Is the ciphertext received the same as the ciphertext sent by A?

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Recognising Correct Plaintext

Example 1

- ▶ Assume the message is English
- ▶ Plaintext had expected structure; assume the plaintext is correct
  - ▶ Sent by A and has not been modified

Example 2

- ▶ Assume the message is English
- ▶ Plaintext had no structure in expected language; assume plaintext is incorrect
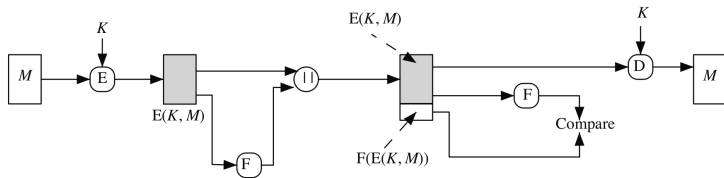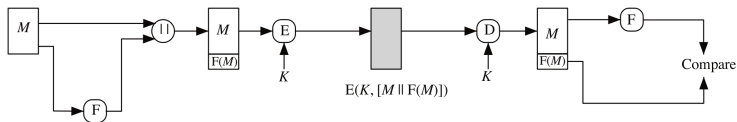 - ▶ Either not sent by A or modified

Example 3

- ▶ Binary data, e.g. image, compressed file
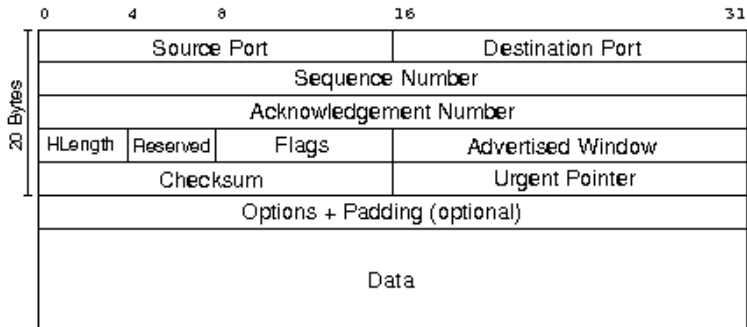- ▶ Cannot know whether correct or incorrect

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Recognising Correct Plaintext

- Valid plaintexts should be small subset of all possible messages
    - E.g. 26^n possible messages of length n; only small subset are valid English phrases
- Plaintext messages have structure
- BUT automatically detecting structure can be difficult
- Add structure to make it easier, e.g.
    - Error detecting code or Frame Check Sequence
    - Packet header

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Adding Error Detecting Code

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

TCP Segment

CSS322

Introduction

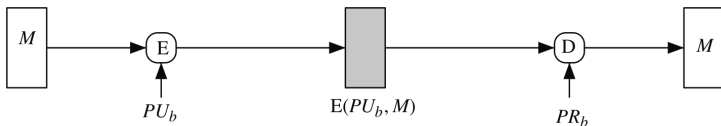Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Public-Key Encryption for Authentication

- ▶ Only provides confidentiality
- ▶ Same assumption as before: plaintext must have structure so can be recognised as correct or incorrect
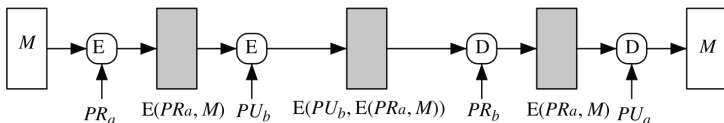
CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Public-Key Encryption for Authentication

- Data authentication (data has not been modified)
- Digital signature: proof that message came from A

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms
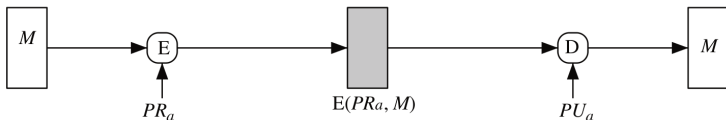
Public-Key Encryption for Authentication

▶ Both confidentiality, authentication and digital signature

M → E → ▮ → E → ▮ → D → ▮ → D → M

PR_a $E(PR_a, M)$ PU_b $E(PU_b, E(PR_a, M))$ PR_b $E(PR_a, M)$ PU_a

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Contents

Message Authentication Requirements and Functions

Authentication with Message Encryption

Authentication with Message Authentication Codes

Security of MACs

MAC Algorithms

CSS322

Introduction

Functions
Auth. with
Encryption
Auth. with MAC

Security

Algorithms

Authentication with Message Authentication Codes

- ▶ Append small, fixed-size block of data to message: cryptographic checksum or MAC
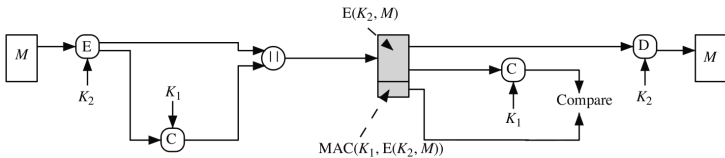
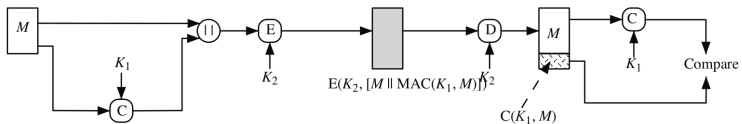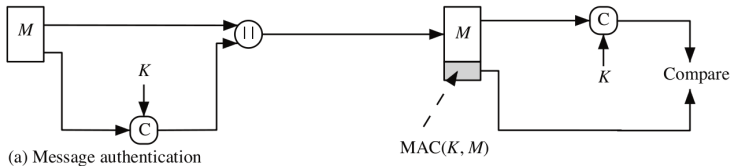$$\mathrm{T} = \mathrm{MAC}(K, M)$$

M = input message
MAC = MAC function
K = shared secret key of k bits
T = message authentication code (or tag) of n bits

- ▶ MAC function also called *keyed hash function*
- ▶ MAC function similar to encryption, but does not need to be reversible
  - ▶ Easier to design stronger MAC functions than encryption functions

(a) Message authentication

MAC(K, M)

E(K_2, [$M \parallel$ MAC(K_1, M)])

C(K_1, M)

E(K_2, M)

MAC(K_1, E(K_2, M))

Contents

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Message Authentication Requirements and Functions

Authentication with Message Encryption

Authentication with Message Authentication Codes

Security of MACs

MAC Algorithms

CSS322

Introduction

Functions
Auth. with
Encryption
Auth. with MAC
Security
Algorithms

Requirement of MACs

Objective of Attacker

- Assume MAC function is known, key K is not
- For valid MAC code for given message x

Requirement of MAC Function

Computation Resistance : given one or more text-MAC pairs
$[x_i, MAC(K, x_i)]$, computationally infeasible to
computer any text-MAC pair $[x, MAC(K, x)]$
for new input $x \neq x_i$

CSS322

Introduction

Functions
Auth. with
Encryption
Auth. with MAC
Security
Algorithms

Security of MACs

Brute Force Attack on Key

- ▶ Attacker knows $[x_1, T_1]$ where $T_1 = MAC(K, x_1)$
- ▶ Key size of k bits: brute force on key, 2^k
- ▶ But ... many tags match T_1
- ▶ For keys that produce tag T_1, try again with $[x_2, T_2]$
- ▶ Effort to find K is approximately 2^k

Brute Force Attack on MAC value

- ▶ For x_m, find T_m without knowing K
- ▶ Similar effort required as one-way/weak collision resistant property for hash functions
- ▶ For n bit MAC value length, effort is 2^n

Effort to break MAC: $\min(2^k, 2^n)$

CSS322

Introduction

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Security of MACs

Cryptanalysis

- Many different MAC algorithms; attacks specific to algorithms
- MAC algorithms generally considered secure

Contents

Functions

Auth. with
Encryption

Auth. with MAC

Security

Algorithms

Message Authentication Requirements and Functions

Authentication with Message Encryption

Authentication with Message Authentication Codes

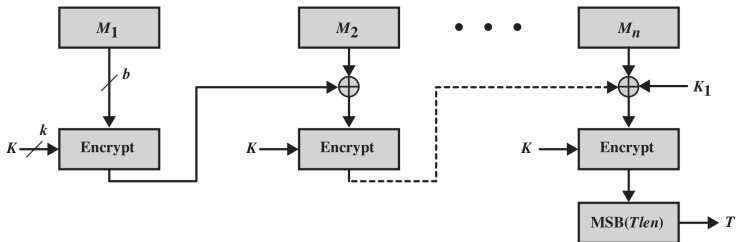Security of MACs

MAC Algorithms

CSS322

Introduction

Functions

Auth. with
Encryption
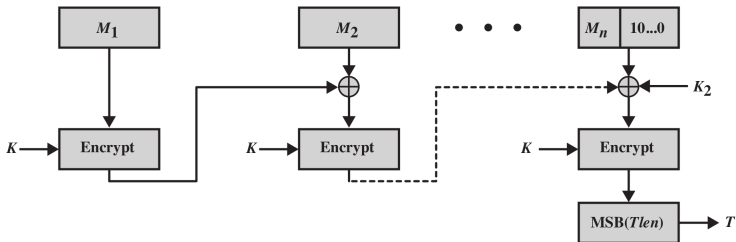
Auth. with MAC

Security

Algorithms

MACs Based on Block Ciphers

- ▶ Data Authentication Algorithm (DAA): based on DES; considered insecure
- ▶ Cipher-Based Message Authentication Code (CMAC): mode of operation used with Triple-DES and AES
- ▶ OMAC, PMAC, UMAC, VMAC, . . .

DAA
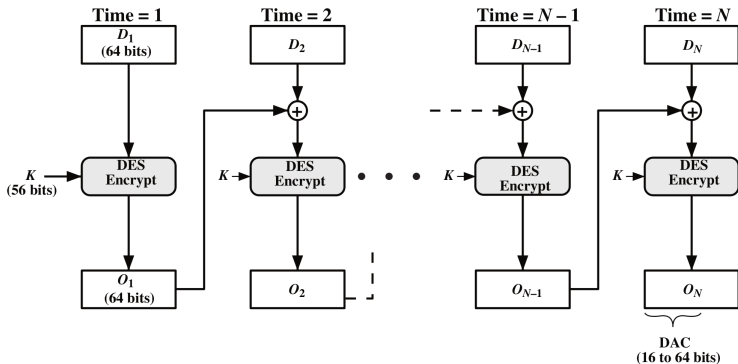
CMAC

(a) Message length is integer multiple of block size

(b) Message length is not integer multiple of block size

CSS322

Introduction

Functions
Auth. with
Encryption
Auth. with MAC
Security
Algorithms

HMAC

- ▶ MAC function derived from cryptographic hash functions
  - ▶ MD5/SHA are fast in software (compared to block ciphers)
  - ▶ Libraries for hash functions widely available

  $$\text{HMAC}(K, M) = \text{H}((K \oplus \text{opad})||\text{H}((K \oplus \textit{ipad})||M))$$

  where ipad= 00110110 repeated, opad= 01011100 repeated

- ▶ Security of HMAC depends on security of hash function used