Authenticated Encryption

Gianluca Dini
Dept. of Ingegneria dell'Informazione
University of Pisa
Emai: gianluca.dini@unipi.it
Version: 2024-04-11

1

Secrecy and integrity

- We have primitives for secrecy and integrity
  - Secrecy: ciphers
  - Integrity: MAC
- What if we wish to achieve secrecy and integrity at the same time?

Apr-24 Authenticated encryption 2

2

Encrypt and authenticate (E&M)

- Alice and Bob want to achieve both confidentiality and integrity

Alice (k1, k2)
message x
$y = E_{k1}(x)$
$t = MAC_{K2}(x)$

---------- [y, t] --------- >

Bob (k1, k2)
$x = D_{k1}(y)$
if $V_{k2}(x, t)$ return x
else return «error»

Apr-24 Authenticated encryption 3

3

Is it secure?

- The tag t might leak information about x
  - Nothing in the definition of security for a MAC implies that it hides information about x
- If the MAC is deterministic (e.g., CBC-MAC and HMAC), then it leaks whether the same message is encrypted twice

Apr-24 Authenticated encryption 4

4

Encrypt then authenticate

- Alice and Bob want to achieve confidentiality and integrity

Alice (k1, k2) **Bob** (k1, k2)

x

$y = E_{k1}(x)$

$t = MAC_{K2}(y)$

------- [y, t] --- >

if (V_{k2}(y, t)) return ($x = D_{k1}(y)$)

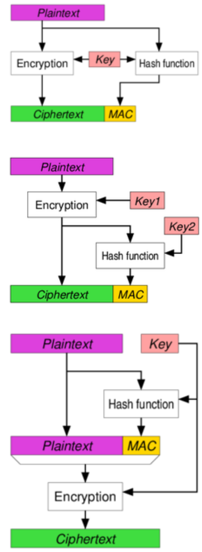
else return "error"

Security of encrypt then authenticate

- It can be proved that if Enc is CPA-secure and MAC is secure then:
  - The combination is CPA-secure (encryption must be randomized)
  - The combination is a secure MAC

Three different approaches

- Encrypt and MAC (E&M)
  - Discouraged
  - SSH
- Encrypt then MAC (EtM)
  - Always correct
  - Ipsec
- MAC then Encrypt (MtE)
  - correctness depends on Enc-MAC combinations
  - TLS/SSL



9

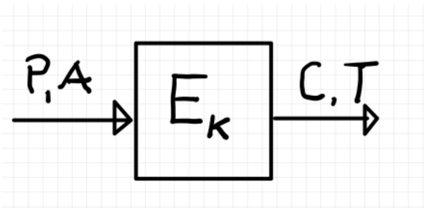
Authenticated Encryption

- Most of applications require *message privacy* and *message authentication*
- Combining privacy and authentication is a challenging task that is rarely done *securely* with *ad-hoc* constructions
- Authenticated Encryption (AE) are *encryption modes* which simultaneously assure the confidentiality and authenticity of data.

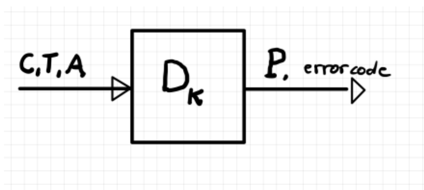
10

AE APIs

Encryption



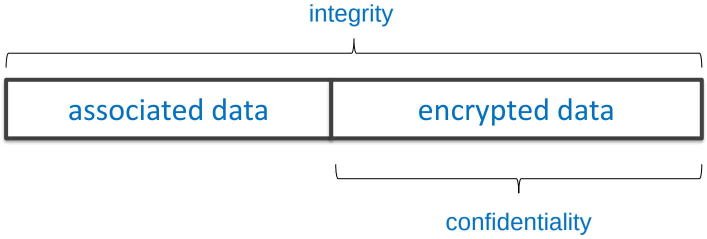
Decryption



11

Authenticated Encryption with Associated Data (AEAD)

- AEAD allows checking the integrity of both the encrypted and unencrypted information in a message.
  - E.g., network packets or frames where the header needs visibility, the payload needs confidentiality, and both need integrity and authenticity.



12

Standards and associated data

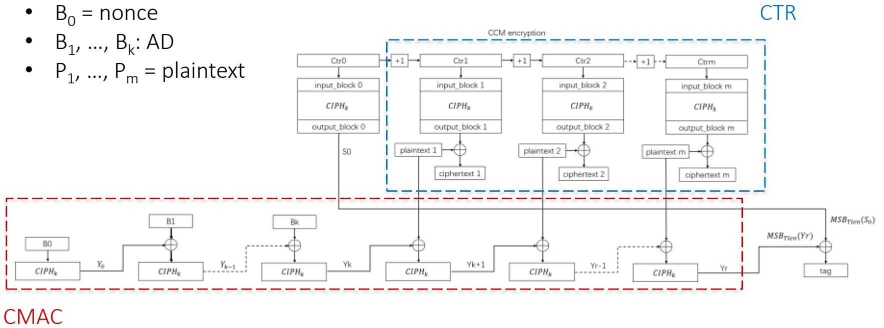
- NIST
  - CCM: CBC-MAC then CTR mode encryption
    - 802.11i
  - GCM: CTR mode encryption then MAC
    - Very efficient
- IETF
  - EAX: CTR mode encryption than OMAC
- NIST and IETF standards support AEAD

Cipher Block Chaining Message Authentication Code (CCM)

- NIST SP 800-38C
- For IEEE 802.11 WiFi
- AES-CTR and CMAC
- Single key K

CCM – encryption flow chart

- B_0 = nonce
- B_1, …, B_k: AD
- P_1, …, P_m = plaintext



15

CCM - drawbacks

- CCM is quite complex: it requires two passes through the plaintext

16

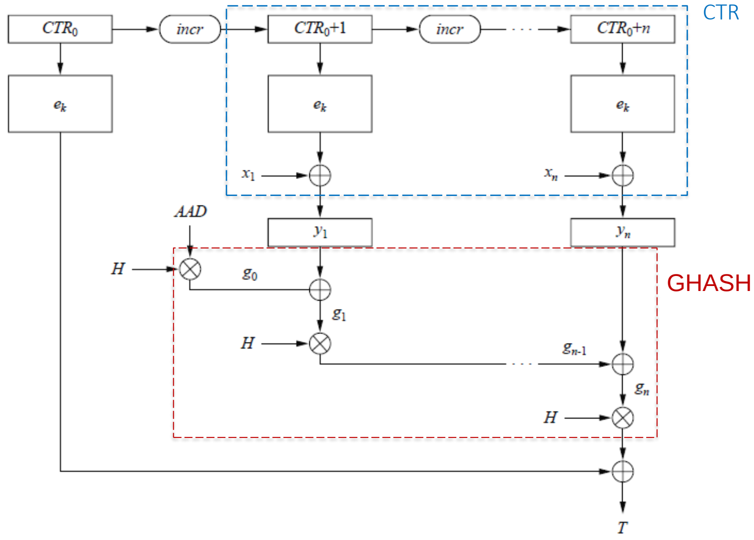
Galois Counter Mode (GCM)

- GCM is an encryption mode which also computes a MAC
  - Confidentiality and authenticity
- GCM protects
  - Confidentiality of a plaintext x
  - Authenticity of plaintext x and
  - Authenticity of AAD which is left in the clear

GCM - main components

- Cipher in the Counter Mode (CTR)
  - Confidentiality
  - Block size: 128 bit (e.g., AES-128)
- Galois field multiplication
  - Authentication
  - GMAC
    - Based on GHASH which exploits multiplication in $GF(2^{128})$
      - Irreducible polynomial $P(x) = x^{128} + x^7 + x^2 + x + 1$
      - Easy and efficient in HW

GCM – flow chart



GCM - advantage

- Assume that AAD and ciphertext constitute a sequence of blocks X = $X_1, X_2, \ldots, X_m$
- GHASH(X, H)
  - $Y_0 = 0^{128}$
  - $Y_i = (Y_{i-1} \oplus X_i) \cdot H$ which can be re-written as
  - $(X_1 \cdot H^m) \oplus (X_2 \cdot H^{m-1}) \oplus \cdots \oplus (X_{m-1} \cdot H^2) \oplus (X_m \cdot H^1)$
  - $H^2, H^3, \ldots, H^m$ can be *precomputed*
  - Xi's can be processed *in parallel*