AES: Advanced Encryption Standard

La Crittografia Simmetrica Moderna

Prof. Fedeli Massimo
IIS Fermi Sacconi Cpia - Ascoli Piceno

7 gennaio 2026

Indice

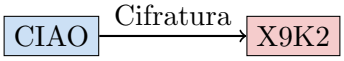
1 Introduzione alla Crittografia

1.1 Cos'è la Crittografia?

La **crittografia** è la scienza che studia le tecniche per rendere un messaggio incomprensibile a persone non autorizzate.

Obiettivi principali:

- Confidenzialità: solo destinatari autorizzati possono leggere il contenuto.
- Integrità: garantisce che il messaggio non sia stato modificato.
- Autenticità: verifica l'identità del mittente.



1.2 Crittografia Simmetrica vs Asimmetrica

Crittografia Simmetrica: Usa la stessa chiave per cifrare e decifrare. Esempi: AES, DES, 3DES. Veloce ed efficiente, ma richiede uno scambio sicuro della chiave.

Crittografia Asimmetrica: Usa due chiavi diverse (pubblica e privata). Esempi: RSA, ECC. Più lenta, ma non richiede il canale sicuro per lo scambio della chiave.

L'**AES** è un algoritmo di crittografia simmetrica.

2 Storia e Contesto di AES

2.1 Prima di AES: Il DES

Il **DES (Data Encryption Standard)** fu standardizzato nel 1977:

- Chiave di 56 bit → oggi **troppo corta** per resistere agli attacchi brute-force.
- Blocchi di 64 bit.

- Negli anni '90, diventò insicuro.
  Come soluzione temporanea si adottò il **3DES**, che applica DES tre volte (chiave effettiva di 168 bit), ma era lento e inefficiente. Emerse la necessità di un nuovo standard.

2.2 La Nascita di AES

Nel 1997, il NIST (National Institute of Standards and Technology) lanciò un concorso internazionale per un nuovo algoritmo di crittografia.

Requisiti:

- Crittografia a blocchi simmetrica
- Blocco: 128 bit
- Chiavi: 128, 192, 256 bit
- Sicurezza ¿ 3DES
- Efficienza su hardware e software

Da 15 candidati, ne furono selezionati 5 finalisti. Nel 2000 fu scelto **Rijndael**, creato dai crittografi belgi **Joan Daemen** e **Vincent Rijmen**.

2.3 Caratteristiche di AES

- **Tipo:** Cifrario a blocchi
- **Dimensione blocco:** 128 bit (16 byte)
- **Dimensioni chiave e round:**
  - AES-128: 10 round
  - AES-192: 12 round
  - AES-256: 14 round

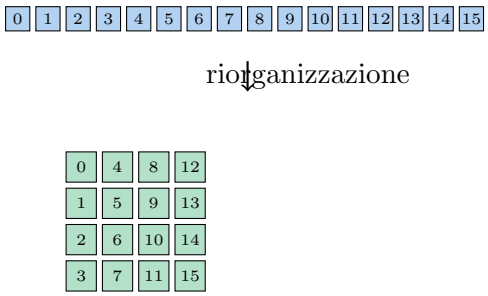
Vantaggi: Sicuro, veloce, flessibile, standard mondiale.
Utilizzi comuni:

- Wi-Fi (WPA2/WPA3)
- HTTPS/TLS
- VPN
- Crittografia disco (BitLocker, FileVault)
- App di messaggistica (Signal, WhatsApp)

3 Struttura di AES

3.1 Rappresentazione dei Dati: La Matrice di Stato

AES organizza i 16 byte del blocco in una **matrice 4×4**, detta *State Matrix*. I byte sono inseriti **per colonna**, non per riga.



3.2 Schema Generale

AES applica una sequenza fissa di operazioni:

1. **AddRoundKey** iniziale.
2. **N round** (10, 12 o 14 a seconda della chiave), ognuno composto da:
   - SubBytes
   - ShiftRows
   - MixColumns
   - AddRoundKey
3. **Round finale** (senza MixColumns):
   - SubBytes
   - ShiftRows
   - AddRoundKey

4 Le Quattro Operazioni di AES

4.1 1. SubBytes – Sostituzione non lineare

Ogni byte è sostituito tramite una tabella fissa chiamata **S-Box**. - Non lineare → introduce *confusione*. - Basata su operazioni nel campo di Galois $GF(2^8)$.

Esempio: il byte `0x53` diventa `0xED` dopo SubBytes.

4.2 2. ShiftRows – Diffusione orizzontale

Le righe della matrice di stato vengono spostate ciclicamente a sinistra:

- Riga 0: 0 posizioni
- Riga 1: 1 posizione
- Riga 2: 2 posizioni
- Riga 3: 3 posizioni

Questo diffonde i byte tra le colonne.

4.3 3. MixColumns – Diffusione verticale

Ogni colonna è trasformata tramite moltiplicazione matriciale in $GF(2^8)$:

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \times \begin{bmatrix} s_{0,j} \\ s_{1,j} \\ s_{2,j} \\ s_{3,j} \end{bmatrix} = \begin{bmatrix} s'_{0,j} \\ s'_{1,j} \\ s'_{2,j} \\ s'_{3,j} \end{bmatrix}$$

Nota: Non applicata nell'ultimo round. Garantisce che ogni byte di output dipenda da tutti i byte della colonna.

4.4 4. AddRoundKey – Combinazione con la chiave

La matrice di stato è combinata con la chiave del round tramite XOR bit-a-bit (operazione $\oplus$).

$$s'_{i,j} = s_{i,j} \oplus k_{i,j}$$

XOR è reversibile: la stessa operazione serve per decifrare.

5 Key Expansion (Espansione della Chiave)

AES richiede una chiave diversa per ogni round. L'algoritmo **Key Expansion** genera tutte le chiavi di round a partire dalla chiave iniziale (128/192/256 bit).

- AES-128: genera 11 chiavi (10 round + iniziale) - Ogni 4 word, si applicano trasformazioni: `RotWord`, `SubWord`, e XOR con costante `Rcon`.

Questo garantisce un *effetto valanga*: piccole modifiche alla chiave influenzano tutti i round.

6 Decifratura

La decifratura è l'inverso della cifratura, con operazioni inverse applicate in ordine inverso:

- **InvSubBytes**: usa la S-Box inversa
- **InvShiftRows**: sposta a destra

- **InvMixColumns**: usa matrice inversa
$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix}$$
- **AddRoundKey**: identica (XOR è auto-inversa)

Le chiavi di round sono usate in ordine inverso.

7 Modi di Operazione

AES cifra solo blocchi di 128 bit. Per messaggi lunghi, si usano **modi di operazione**:

ECB (Electronic Codebook): Ogni blocco è cifrato indipendentemente. Pericoloso! Blocchi identici → cifrati identici (es. immagini rivelano pattern).

CBC (Cipher Block Chaining): Ogni blocco in chiaro è XOR con il blocco cifrato precedente. Richiede un **IV (Initialization Vector)** casuale. Più sicuro di ECB.

CTR (Counter): Trasforma AES in cifrario a flusso. Usa un contatore cifrato con AES, poi XOR con il plaintext. **Parallelizzabile** e adatto per accesso casuale.

GCM (Galois/Counter Mode): Combina CTR con autenticazione (**GHASH**). Fornisce *crittografia + integrità*. Usato in TLS 1.3. **Consigliato per nuove applicazioni.**

8 Sicurezza di AES

8.1 Robustezza contro Brute-Force

Variante	Chiavi possibili	Sicurezza
AES-128	$2^{128} \approx 3.4 \times 10^{38}$	Alta
AES-192	2^{192}	Altissima
AES-256	$2^{256} \approx 1.1 \times 10^{77}$	Estrema

Con 1 miliardo di tentativi/sec, servirebbero **miliardi di anni** per rompere anche solo l'AES-128.

8.2 Attacchi Conosciuti

- **Biclique attack**: teorico, riduce complessità a $2^{126.1}$ → non pratico. - **Related-key attacks**: richiedono scenari irrealistici. - **Minacce reali**: side-channel attacks, implementazioni deboli, riutilizzo di IV, uso di ECB.

8.3 AES e Computer Quantistici

L'algoritmo di Grover permette una ricerca quadratica: $O(\sqrt{N})$ invece di $O(N)$.

- AES-128 → sicurezza ridotta a 2^{64} → vulnerabile. - AES-256 → sicurezza 2^{128} → ancora sicuro.

Raccomandazione: usare AES-256 in contesti sensibili al rischio quantistico.

9 Applicazioni Pratiche

AES è usato ovunque:

- **Comunicazioni:** Wi-Fi (WPA2/3), HTTPS, VPN, SSH, Signal
- **Storage:** BitLocker, FileVault, LUKS, database cifrati

- **File:** PDF, Office, 7-Zip, WinRAR
- **Hardware:** Processori con AES-NI, smartphone, IoT

9.1 Accelerazione Hardware: AES-NI

Moderni processori (Intel, AMD, ARM) includono istruzioni dedicate (**AES-NI**) che:

- Accelerano AES di 4–10×
- Riducono consumo energetico
- Mitigano side-channel attacks

10 Esempio Pratico in Python

"'python from cryptography.hazmat.primitives.ciphers import Cipher, algorithms, modes from cryptography.hazmat.backends import default$_b$*ackendimportos*

Chiave e IV casuali key = os.urandom(32) 256 bit iv = os.urandom(16) 128 bit

Cifrario AES in modalità CBC cipher = Cipher(algorithms.AES(key), modes.CBC(iv), backend=default$_b$*ack*

Padding manuale (in pratica usare PKCS7) plaintext = b"Messaggio segreto per la scuola!" plaintext$_p$*added* = $plaintext + b'00' * (16 - len(plaintext)$

Cifratura encryptor = cipher.encryptor() ciphertext = encryptor.update(plaintext$_p$*added*) + $encryptor.finalize()$

Decifratura decryptor = cipher.decryptor() decrypted = decryptor.update(ciphertext) + decryptor.finalize() print(decrypted.rstrip(b'00').decode())

11 Conclusioni

- AES è lo standard mondiale di crittografia simmetrica.
- Basato su 4 operazioni che garantiscono confusione e diffusione.
- Sicuro contro tutti gli attacchi pratici conosciuti.
- AES-256 è resistente anche ai computer quantistici.
- Attenzione alla scelta del **modo di operazione**: evitare ECB, preferire GCM.
- Grazie ad AES-NI, la crittografia è veloce e efficiente su hardware moderno.

12 Risorse per Approfondire

- **Documenti ufficiali:** FIPS 197 (NIST)
- **Tool online:**
  - `https://www.cryptool.org/`
  - `https://aesencryption.net/`
  - `https://www.javainuse.com/aesgenerator`
- **Libri:** "Understanding Cryptography" (Paar Pelzl), "The Design of Rijndael"
- **Video:** Computerphile (YouTube), Cryptography I (Coursera)