

AES: Advanced Encryption Standard

La Crittografia Simmetrica Moderna

Prof. Fedeli Massimo

IIS Fermi Sacconi Cpia - Ascoli Piceno

7 gennaio 2026

Contenuti

- 1 Introduzione alla Crittografia
- 2 Storia e Contesto di AES
- 3 Struttura di AES
- 4 Le Quattro Operazioni di AES
- 5 Key Expansion
- 6 Decifratura
- 7 Modi di Operazione
- 8 Sicurezza di AES
- 9 Applicazioni Pratiche
- 10 Conclusioni

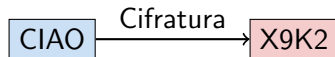
Cos'è la Crittografia?

Definizione

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

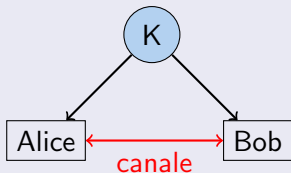
Obiettivi principali:

- **Confidenzialità**: solo destinatari autorizzati possono leggere
- **Integrità**: il messaggio non è stato modificato
- **Autenticità**: verifica dell'identità del mittente



Tipi di Crittografia

Crittografia Simmetrica

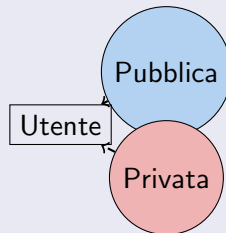


Stessa chiave per cifrare e decifrare

Esempi: AES, DES, 3DES

Veloce ed efficiente

Crittografia Asimmetrica



Chiavi diverse: pubblica e privata

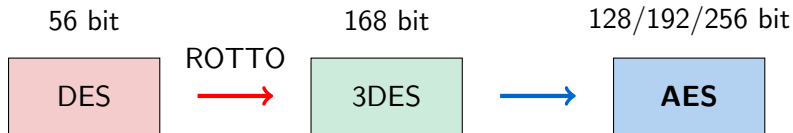
Esempi: RSA, ECC

Più lenta ma non richiede scambio sicuro

AES è un algoritmo di **crittografia simmetrica**

DES - Data Encryption Standard (1977)

- Sviluppato da IBM, standardizzato da NIST
- Chiave di **56 bit** (troppo corta!)
- Blocchi di 64 bit
- Negli anni '90 diventa vulnerabile agli attacchi brute force



3DES: soluzione temporanea (applica DES tre volte)

Necessità: nuovo standard più sicuro e veloce

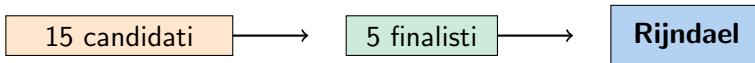
La Nascita di AES

Concorso NIST (1997-2000)

Il NIST (National Institute of Standards and Technology) bandisce un concorso pubblico per trovare un successore del DES

Requisiti:

- Crittografia a blocchi simmetrica
- Lunghezza blocco: 128 bit
- Lunghezze chiave: 128, 192, 256 bit
- Sicurezza superiore a 3DES
- Efficienza su varie piattaforme



Vincitore (2000): [Rijndael](#), sviluppato dai crittografi belgi Joan Daemen e Vincent Rijmen

Caratteristiche di AES

Parametri Fondamentali

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

Vantaggi

- Altamente sicuro
- Veloce ed efficiente
- Flessibile
- Standard mondiale

Utilizzi:

- Wi-Fi (WPA2/WPA3)
- HTTPS/TLS
- VPN
- Crittografia disco
- Messaggistica sicura

Come AES Rappresenta i Dati

La Matrice di Stato (State Matrix)

I 16 byte del blocco sono organizzati in una matrice 4×4

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

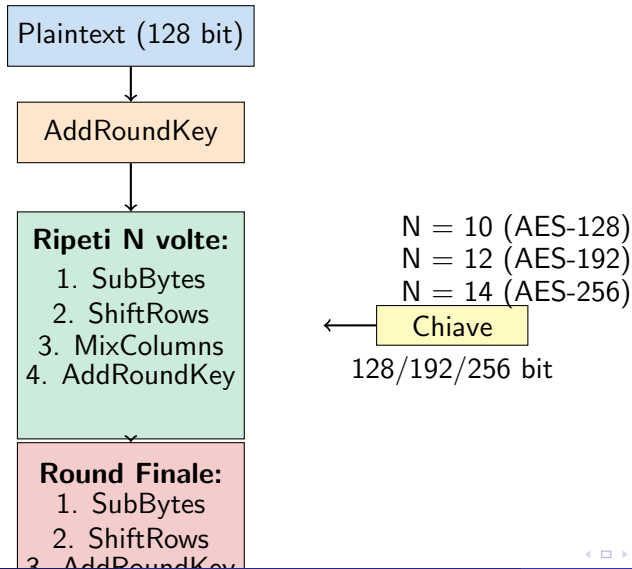
128 bit = 16 byte
↓
riorganizzazione

0	4	8	12
1	5	9	13
2	6	10	14
3	7	11	15

Matrice di Stato
(4 righe \times 4 colonne)

Nota: I byte sono disposti *per colonna*, non per riga!

Schema Generale di AES



1. SubBytes - Sostituzione

Obiettivo

Sostituire ogni byte della matrice con un altro byte usando una tabella speciale chiamata **S-Box** (Substitution Box)

Come funziona:

- 1 Prendi un byte (es. 0x53)
- 2 Dividi in due nibble: 5 e 3
- 3 Usa 5 come riga, 3 come colonna
- 4 Il valore nella S-Box è il byte sostituito

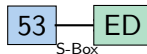
S-Box Semplificata (4×4)

	0	1	2	3
0	63	7C	77	7B
1	F2	6B	6F	C5
2	30	01	67	2B
3	FE	D7	AB	76

Proprietà:

- Non lineare (confusione)
- Iniettiva (ogni input \rightarrow output unico)
- Basata su matematica in $GF(2^8)$

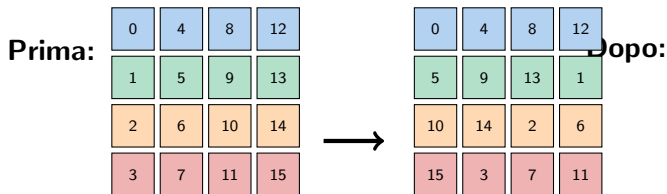
Esempio:



2. ShiftRows - Spostamento Righe

Obiettivo

Spostare ciclicamente i byte di ogni riga verso sinistra di un numero fisso di posizioni



Spostamenti:

- Riga 0: 0 posizioni (nessuno spostamento)
- Riga 1: 1 posizione a sinistra
- Riga 2: 2 posizioni a sinistra
- Riga 3: 3 posizioni a sinistra

Effetto: Diffonde i byte tra le colonne (*diffusione*)

3. MixColumns - Mescolamento Colonne

Obiettivo

Mescolare i byte all'interno di ogni colonna usando operazioni matematiche in $GF(2^8)$

Operazione: Moltiplicazione matriciale con una matrice fissa

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

Nota importante:

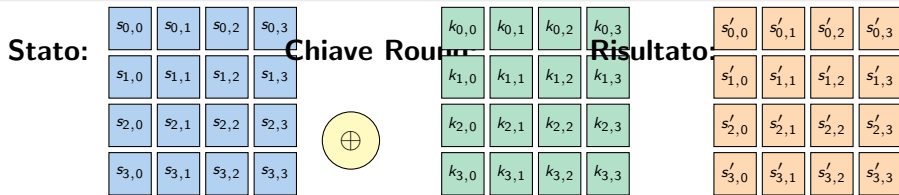
- Le operazioni sono in **campo di Galois** $GF(2^8)$
- Non è l'aritmetica normale! (es: $02 \times 03 \neq 6$)
- Ogni colonna viene processata indipendentemente
- **Non viene applicata** nell'ultimo round

Effetto: Massima diffusione - ogni byte di output dipende da tutti i byte di input della colonna

4. AddRoundKey - Aggiunta Chiave di Round

Obiettivo

Combinare la matrice di stato con la chiave del round corrente usando XOR bit a bit



Operazione XOR (\oplus):

A	B	A \oplus B
0	0	0
0	1	1
1	0	1
1	1	0

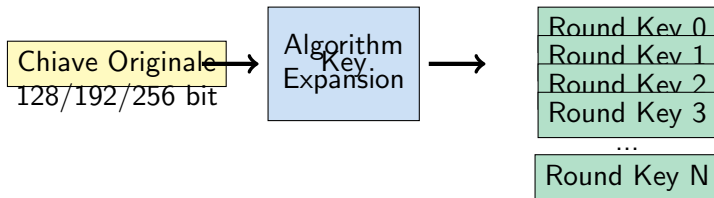
Esempio: 10110011 \oplus 11001010 = 01111001

Key Expansion - Espansione della Chiave

Problema

AES necessita di una chiave diversa per ogni round, ma l'utente fornisce una sola chiave iniziale

Soluzione: L'algoritmo di **Key Expansion** genera tutte le chiavi di round dalla chiave originale



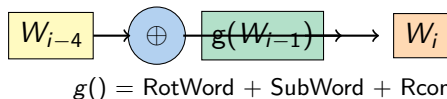
Numero di chiavi generate:

- AES-128: 11 chiavi (1 iniziale + 10 round)
- AES-192: 13 chiavi (1 iniziale + 12 round)
- AES-256: 15 chiavi (1 iniziale + 14 round)

Come Funziona la Key Expansion

Processo semplificato per AES-128:

- ① La chiave di 128 bit viene divisa in 4 *word* (parole) di 32 bit ciascuna: W_0, W_1, W_2, W_3
- ② Per generare le nuove word (W_4, W_5, \dots, W_{43}):
 - **Se** la posizione è multipla di 4: applica una trasformazione speciale
 - ① **RotWord**: ruota i byte
 - ② **SubWord**: applica S-Box
 - ③ **XOR con costante**: Rcon
 - **Altrimenti**: semplice XOR con le word precedenti



Proprietà importante: Ogni bit della chiave influenza tutti i round successivi (effetto valanga)

Principio Fondamentale

La decifratura è il **processo inverso** della cifratura, applicato in ordine inverso

CIFRATURA

- ① AddRoundKey
- ② **Per ogni round:**
 - SubBytes
 - ShiftRows
 - MixColumns
 - AddRoundKey
- ③ **Round finale:**
 - SubBytes
 - ShiftRows
 - AddRoundKey

DECIFRATURA

- ① AddRoundKey
- ② **Per ogni round:**
 - InvShiftRows
 - InvSubBytes
 - AddRoundKey
 - InvMixColumns
- ③ **Round finale:**
 - InvShiftRows
 - InvSubBytes
 - AddRoundKey

InvSubBytes

- Usa la **S-Box inversa**
- Stessa logica ma tabella diversa
- $s = \text{S-Box}(x) \implies x = \text{InvS-Box}(s)$

InvShiftRows

- Spostamento verso **destra**
- Riga 0: 0 posizioni
- Riga 1: 1 posizione (\rightarrow)
- Riga 2: 2 posizioni (\rightarrow)
- Riga 3: 3 posizioni (\rightarrow)

InvMixColumns

- Usa una **matrice inversa**
- Stessa operazione matematica
- Annulla l'effetto di MixColumns

$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix}$$

AddRoundKey

- **Identica!** XOR è auto-inversa

Problema

AES cifra solo blocchi di 128 bit. Come gestiamo messaggi più lunghi?

Soluzione: I **modi di operazione** definiscono come applicare AES a messaggi di lunghezza arbitraria

Modi principali:

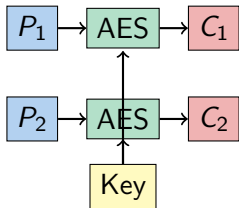
- **ECB** - Electronic Codebook
- **CBC** - Cipher Block Chaining
- **CTR** - Counter
- **GCM** - Galois/Counter Mode

Caratteristiche:

- **Parallelizzazione**
- **Resistenza agli attacchi**
- **Gestione errori**
- **Autenticazione**

ECB vs CBC

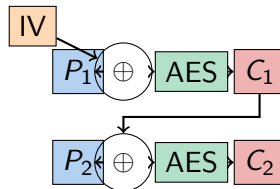
ECB - Electronic Codebook



PROBLEMA: Blocchi identici \rightarrow cifrati identici

Esempio: l'immagine del pinguino Tux cifrata con ECB mostra il pattern originale

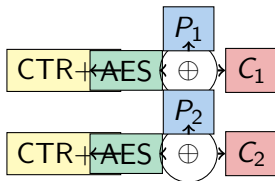
CBC - Cipher Block Chaining



VANTAGGIO: Ogni blocco dipende dai precedenti

- Usa un IV (Initialization Vector)
- Più sicuro di ECB

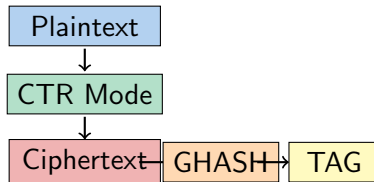
CTR - Counter Mode



Vantaggi:

- Parallelizabile
- Accesso casuale ai blocchi
- Simmetrico (cifra=decifra)

GCM - Galois/Counter Mode



GCM = CTR + Autenticazione

Vantaggi:

- Cifratura + autenticazione
- Rileva manomissioni
- Standard per TLS 1.3
- **Molto usato oggi!**

Quanto è Sicuro AES?

Forza Brute-Force

Tentare tutte le chiavi possibili

Variante	Chiavi Possibili	Tempo Stimato*	Sicurezza
AES-128	$2^{128} \approx 3.4 \times 10^{38}$	Miliardi di anni	Alta
AES-192	$2^{192} \approx 6.3 \times 10^{57}$	Eoni	Altissima
AES-256	$2^{256} \approx 1.1 \times 10^{77}$	Oltre l'età dell'universo	Estrema

*Con computer attuali che provano 1 miliardo di chiavi/secondo

Per confronto:

- Atomi nell'universo osservabile: $\approx 10^{80}$
- AES-256 ha quasi lo stesso numero di chiavi!

Conclusione: AES è praticamente **inattaccabile** con la tecnologia attuale

Attacchi Teorici

- **Biclique Attack (2011)**

- Riduce la complessità di AES-128 a $2^{126.1}$
- Miglioramento trascurabile
- Non pratico

- **Related-Key Attacks**

- Funzionano solo in scenari irrealistici
- Richiedono chiavi correlate
- Non applicabili in pratica

Minacce Reali

- **Implementazioni deboli**

- Side-channel attacks
- Timing attacks
- Cache attacks

- **Gestione chiavi**

- Chiavi deboli
- Riutilizzo di IV
- Storage non sicuro

- **Modi d'uso errati**

- ECB per dati ripetitivi
- Padding oracle attacks

Importante

AES è sicuro, ma deve essere **implementato e usato correttamente!**

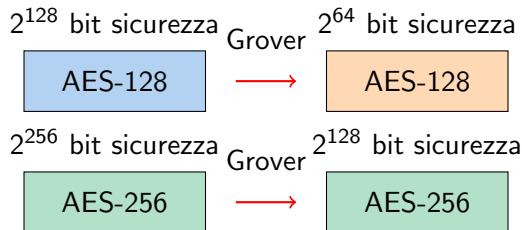
La Minaccia Quantistica

I computer quantistici potrebbero rompere molti algoritmi crittografici

Algoritmo di Grover (1996):

- Ricerca in uno spazio di N elementi
- Computer classico: $O(N)$ operazioni
- Computer quantistico: $O(\sqrt{N})$ operazioni

Impatto su AES:



Applicazioni di AES nel Mondo Reale

Comunicazioni:

- **Wi-Fi:** WPA2/WPA3
- **VPN:** OpenVPN, IPsec
- **TLS/SSL:** HTTPS
- **Messaggistica:** Signal, WhatsApp
- **SSH:** connessioni sicure

Storage:

- BitLocker (Windows)
- FileVault (macOS)
- LUKS (Linux)
- Crittografia database
- Cloud storage (Google Drive, Dropbox)

File e Archivi:

- 7-Zip (cifratura archivi)
- WinRAR
- PDF cifrati
- Office (Word, Excel)

Multimedia:

- Streaming protetto (DRM)
- Videogiochi (protezione)

Hardware:

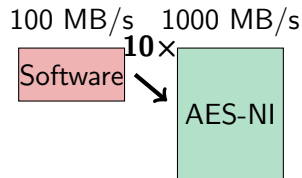
- Processori moderni (AES-NI)
- Smartphone
- Dispositivi IoT

AES-NI (AES New Instructions)

Set di istruzioni nei processori moderni per accelerare AES

Vantaggi:

- **Velocità:** 4-10× più veloce
- **Sicurezza:** resistente a side-channel
- **Efficienza:** meno consumo energetico



Disponibilità:

- Intel: dal 2010 (Westmere)
- AMD: dal 2011 (Bulldozer)
- ARM: ARMv8 Cryptography Extensions
- Smartphone moderni

Esempio Pratico in Python

Uso di AES con la libreria cryptography:

```
from cryptography.hazmat.primitives.ciphers import Cipher, algorithms, modes
from cryptography.hazmat.backends import default_backend
import os

# Generazione chiave casuale (256 bit)
key = os.urandom(32) # 32 byte = 256 bit

# Generazione IV casuale (128 bit)
iv = os.urandom(16) # 16 byte = 128 bit

# Creazione del cifrario AES in modalita CBC
cipher = Cipher(
    algorithms.AES(key),
    modes.CBC(iv),
    backend=default_backend()
)

# Cifratura
plaintext = b"Messaggio-segreto-per-la-scuola!"
# Padding necessario per blocchi di 16 byte
plaintext_padded = plaintext + b'\x00' * (16 - len(plaintext) % 16)

encryptor = cipher.encryptor()
ciphertext = encryptor.update(plaintext_padded) + encryptor.finalize()

print(f"Testo-cifrato: {ciphertext.hex()}")

# Decifratura
```

① **AES è lo standard mondiale** per la crittografia simmetrica

- Sicuro, veloce, versatile

② **Struttura basata su 4 operazioni**

- SubBytes (confusione)
- ShiftRows (diffusione)
- MixColumns (diffusione)
- AddRoundKey (combinazione con chiave)

③ **Sicurezza eccezionale**

- Nessun attacco pratico conosciuto
- AES-256 resistente anche ai computer quantistici

④ **Modi di operazione** essenziali per sicurezza reale

- Evitare ECB
- Preferire GCM per autenticazione

⑤ **Accelerazione hardware** rende AES velocissimo

Documentazione Ufficiale:

- FIPS 197 - Specifica ufficiale AES
- NIST Special Publications

Tool Online:

- <https://www.cryptool.org/> - CrypTool (simulazioni)
- <https://aesencryption.net/> - AES Calculator
- <https://www.javainuse.com/aesgenerator> - AES Generator

Libri Consigliati:

- "Understanding Cryptography" - Paar & Pelzl
- "The Design of Rijndael" - Daemen & Rijmen

Video e Corsi:

- Coursera - Cryptography I (Stanford)
- YouTube - Computerphile (video su AES)

Quiz di Verifica

- ❶ Qual è la dimensione del blocco in AES?
 - ☐ 64 bit
 - ☐ 128 bit
 - ☐ 256 bit
- ❷ Quanti round ha AES-192?
 - ☐ 10
 - ☐ 12
 - ☐ 14
- ❸ Quale operazione NON viene applicata nell'ultimo round?
 - ☐ SubBytes
 - ☐ MixColumns
 - ☐ AddRoundKey
- ❹ Quale modo di operazione include autenticazione?
 - ☐ ECB
 - ☐ CBC
 - ☐ GCM

Risposte: 1-B, 2-B, 3-B, 4-C

Grazie per l'attenzione!

Domande?

Prof. Massimo

IIS Fermi Sacconi Ceci
Ascoli Piceno

*"La crittografia è l'arte di scrivere in codice.
AES è il suo capolavoro moderno."*