Análisis Comparativo de Algoritmos de Criptografía Ligera

Kevin Arturo Amaya Osorio

Universidad Nacional de Colombia

kaamayao@unal.edu.co

December 5, 2025

Índice

¿Qué es la Criptografía Ligera? [2]

Definición

La **criptografía ligera** (Lightweight Cryptography) es una rama de la criptografía que diseña algoritmos optimizados para dispositivos con recursos limitados.

Características principales:

- Bajo consumo de memoria (RAM y ROM)
- Menor consumo energético
- Implementaciones compactas en hardware
- Adecuada para IoT, RFID, sensores y sistemas embebidos

Cifrado Simétrico y Cifrado de Bloque

Cifrado Simétrico

Un esquema de cifrado simétrico es una tupla $(\mathcal{K}, \mathcal{M}, \mathcal{C}, E, D)$ donde:

- $\mathcal{K}$ es el espacio de claves, $\mathcal{M}$ el espacio de mensajes, $\mathcal{C}$ el espacio de textos cifrados
- $E : \mathcal{K} \times \mathcal{M} \to \mathcal{C}$ es la función de cifrado
- $D : \mathcal{K} \times \mathcal{C} \to \mathcal{M}$ es la función de descifrado
- $\forall k \in \mathcal{K}, \forall m \in \mathcal{M} : D(k, E(k, m)) = m$

Cifrado de Bloque

Es un cifrado simétrico donde $\mathcal{M} = \mathcal{C} = \{0,1\}^n$ para un tamaño de bloque n fijo.

- $E_k : \{0,1\}^n \to \{0,1\}^n$ es una permutación (biyección)
- Para cada clave k, existe $E_k^{-1} = D_k$
- Tamaños típicos: $n = 64$ (DES, PRESENT) o $n = 128$ (AES)

Confusión y Difusión (Shannon, 1945)

Confusión

Hace que la relación entre el texto cifrado y la clave sea lo más compleja posible.

- Cada bit del texto cifrado debe depender de varios bits de la clave
- Se logra principalmente con **S-boxes** (tablas de sustitución)
- Objetivo: que cambiar un bit de la clave cambie muchos bits del resultado

Difusión

Dispersa la influencia de cada bit del texto plano a lo largo del texto cifrado.

- Cada bit del texto plano debe afectar muchos bits del texto cifrado
- Se logra con **permutaciones** y **operaciones de mezcla**
- Objetivo: que cambiar un bit del mensaje cambie 50% del resultado

Red de Feistel [1]

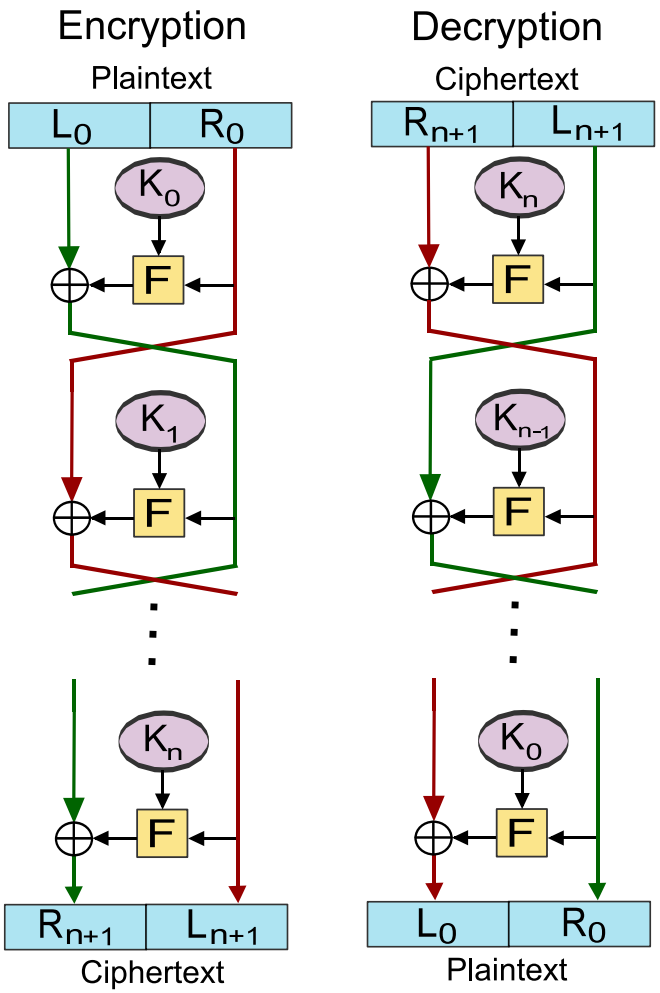


Imagen: Wikimedia Commons (CC BY-SA 3.0)

Características:

- Divide el bloque en dos mitades (L, R)
- Aplica función F con subclave en cada ronda
- Resultado se XOR con la otra mitad
- Las mitades se intercambian
- **Ventaja**: Cifrado y descifrado usan la misma estructura

Usado en: DES, 3DES, Blowfish, Twofish

Red de Sustitución-Permutación (SPN) [3]

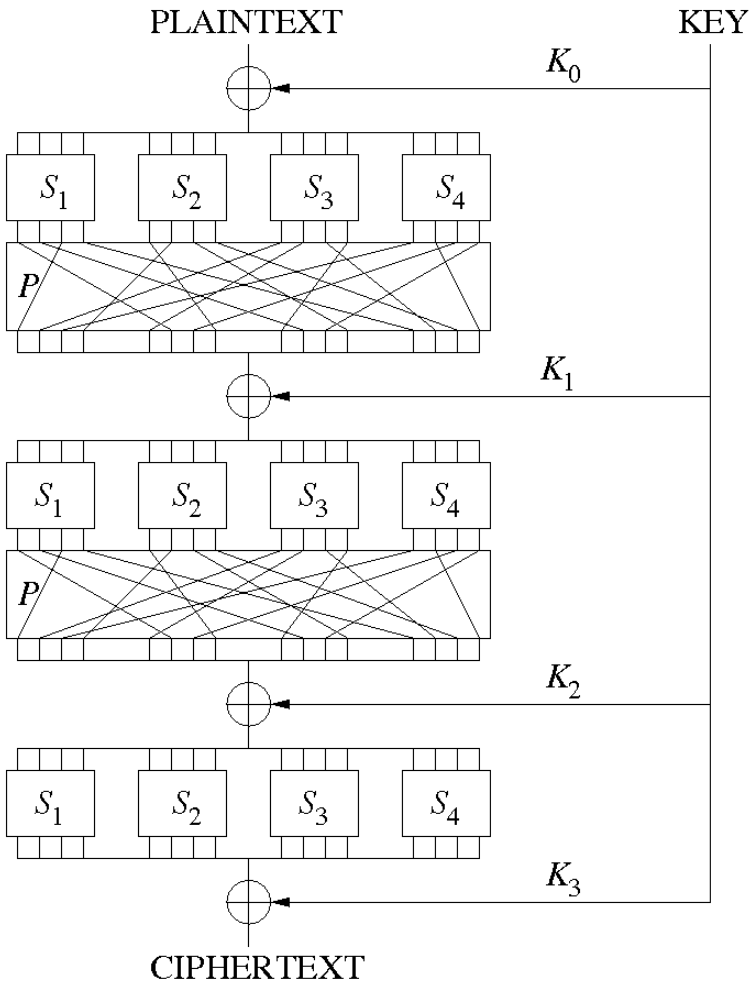


Imagen: Wikimedia Commons (CC BY-SA 3.0)

Componentes:

- **S-box**: Sustitución no lineal (confusión)
- **P-box**: Permutación de bits (difusión)
- **Key mixing**: XOR con subclave de ronda

Características:

- Múltiples rondas de S-P-Key
- Alta difusión y confusión
- Paralelizable en hardware

Usado en: AES, PRESENT, SHARK

Fundamentos Matemáticos de Seguridad

Teorema de Luby-Rackoff (1988) - Feistel [5]

Si la función de ronda f es una **función pseudoaleatoria** (PRF) segura:

- **3 rondas** $\Rightarrow$ Permutación pseudoaleatoria (PRP)
- **4 rondas** $\Rightarrow$ Permutación pseudoaleatoria fuerte (SPRP)

Esto garantiza seguridad demostrable para redes de Feistel.

Estrategia Wide Trail (2001) - SPN [4]

Diseñada por Daemen y Rijmen para AES:

- Maximiza el número de **S-boxes activas** en cualquier camino diferencial/lineal
- Usa el **branch number** para medir difusión
- Proporciona cotas demostrables contra criptoanálisis diferencial y lineal

Feistel vs SPN: Comparación

Característica	Feistel	SPN
Cifrado = Descifrado	Sí (mismo código)	No (inversos)
Paralelismo	Menor (mitad del bloque)	Mayor (bloque completo)
S-box invertible	No requerido	Requerido
Uso de recursos	Menor	Mayor
Velocidad en hardware	Menor	Mayor
Seguridad demostrable	Luby-Rackoff (1988)	Wide Trail (2001)

Trade-offs clave:

- **Feistel**: Ideal para implementaciones con recursos limitados donde el mismo circuito sirve para cifrar y descifrar
- **SPN**: Ideal para alto rendimiento en hardware paralelo, pero requiere más área/código

DES - Historia [1]

Origen

Desarrollado por IBM (1973-1974) basado en el cifrado **Lucifer** de Horst Feistel.

Cronología:

- **1973**: NBS (ahora NIST) solicita propuestas para un estándar de cifrado
- **1974**: IBM presenta una versión modificada de Lucifer
- **1975**: NSA colabora con IBM, reduce la clave de 112 a 56 bits
- **1977**: Publicado como FIPS PUB 46
- **1999**: Roto por fuerza bruta en 22 horas (EFF)
- **2002**: Reemplazado oficialmente por AES

Controversia

El diseño de las S-boxes fue clasificado. En 1990 se descubrió que resistían criptoanálisis diferencial, técnica que la NSA aparentemente conocía desde 1977.

DES/3DES - Aplicaciones

- Cifrado de PINs en cajeros ATM (estándar PCI PIN Security)
- Tarjetas EMV con chip — estándar global desde los años 90
- Sistemas de mensajería financiera interbancaria (SWIFT legacy)
- Terminales de punto de venta (POS) en comercios
- Sistemas legacy en bancos colombianos (migración a AES en curso)

Estado actual

NIST deprecó 3DES en 2023. EMVCo está migrando a AES.

DES - Características [1, 2]

Característica	**Valor**
Año de publicación	1977
Estructura	Red de Feistel
Tamaño de clave	56 bits (64 con paridad)
Tamaño de bloque	64 bits
Número de rondas	16
Nivel de seguridad	Bajo (obsoleto)

Nota

El DES fue declarado obsoleto en 2005 y es vulnerable a ataques de fuerza bruta con hardware moderno.

DES - Implementación

1. **Permutación inicial (IP)**: Reordena los 64 bits de entrada
2. **16 rondas Feistel**:
   - Bloque dividido en L (32 bits) y R (32 bits)
   - $L_i = R_{i-1}, \quad R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$
3. **Función f**: Expansión (32→48 bits) → XOR con subclave → 8 S-boxes (6→4 bits) → Permutación P
4. **Permutación final ($\mathbf{IP^{-1}}$)**: Inversa de IP

Key Schedule

56 bits de clave → 16 subclaves de 48 bits mediante rotaciones y permutaciones (PC-1, PC-2).

AES - Historia [3]

La Competencia NIST (1997-2000)

Proceso abierto de 5 años para reemplazar DES. 15 candidatos de 12 países fueron evaluados por la comunidad criptográfica mundial.

Cronología:

- **1997**: NIST anuncia competencia para nuevo estándar
- **1998**: 15 candidatos presentados (Round 1)
- **1999**: 5 finalistas seleccionados (Round 2)
- **Oct 2000**: **Rijndael** es anunciado ganador
- **Nov 2001**: Publicado como FIPS PUB 197

Creadores

Joan Daemen y Vincent Rijmen (Bélgica). El nombre "Rijndael" combina sus apellidos.

AES - Aplicaciones

- WiFi WPA2/WPA3 - estándar IEEE 802.11i
- HTTPS/TLS 1.3 - navegación web segura
- BitLocker, FileVault, VeraCrypt - cifrado de disco
- WhatsApp, Signal - mensajería end-to-end (documentado)
- Bancolombia: 23,000 dispositivos con BitLocker (Microsoft, 2021)

Hardware

Intel AES-NI, AMD-V, ARM Cryptography Extensions - aceleración nativa en procesadores modernos.

AES-128 - Características [3, 2]

Característica	**Valor**
Año de publicación	2001
Estructura	Red de Sustitución-Permutación (SPN)
Tamaño de clave	128 bits
Tamaño de bloque	128 bits
Número de rondas	10
Nivel de seguridad	Excelente

Nota

AES es el estándar actual para cifrado simétrico (FIPS 197). Soporta también claves de 192 y 256 bits.

AES-128 - Implementación

1. **AddRoundKey**: XOR estado con subclave
2. **10 rondas** (última sin MixColumns):
   - **SubBytes**: Sustitución via S-box (inversión en $GF(2^8)$ + transformación afín)
   - **ShiftRows**: Rotación de filas (0,1,2,3 posiciones)
   - **MixColumns**: Multiplicación en $GF(2^8)$ por columna
   - **AddRoundKey**: XOR con subclave de ronda

Key Schedule

128 bits $\rightarrow$ 11 subclaves de 128 bits usando RotWord, SubWord y Rcon.

PRESENT - Historia [1]

Origen

Desarrollado en 2007 por Orange Labs (Francia), Ruhr University Bochum (Alemania) y Technical University of Denmark.

Cronología:

- **2007**: Presentado en CHES 2007 (Vienna)
- **2012**: Estandarizado en ISO/IEC 29192-2
- **2014**: Incluido en ISO/IEC 29167-11 para RFID
- **2019**: Actualizado en ISO/IEC 29192-2:2019

Objetivo de Diseño

Cifrado **ultra-ligero** donde seguridad y eficiencia en hardware son igualmente importantes. Solo 1570 GE (gate equivalents), competitivo con cifrados de flujo.

PRESENT - Aplicaciones y Legado

- Estándar ISO/IEC 29192-2 e ISO/IEC 29167-11
- Sin implementaciones comerciales confirmadas
- Su S-box y diseño influenciaron: GIFT, LED, PHOTON, SPONGENT
- **GIFT-COFB** (basado en PRESENT) fue finalista NIST 2021

Futuro

NIST seleccionó **ASCON** como estándar (2023). PRESENT no se usa directamente, pero su legado vive en cifrados modernos.

PRESENT-80 - Características [1, 2]

Característica	**Valor**
Año de publicación	2007
Estructura	Red de Sustitución-Permutación (SPN)
Tamaño de clave	80 bits (también 128 bits)
Tamaño de bloque	64 bits
Número de rondas	31
Nivel de seguridad	Adecuado para IoT

Nota

PRESENT es un cifrado ultra-ligero diseñado para dispositivos con recursos limitados (ISO/IEC 29192-2:2012).

PRESENT-80 - Implementación

1. **31 rondas**:
   - **AddRoundKey**: XOR estado (64 bits) con subclave
   - **S-box layer**: 16 S-boxes de 4 bits en paralelo
   - **pLayer**: Permutación de bits (bit $i \rightarrow$ bit $P(i)$)
2. **AddRoundKey final**: XOR con subclave 32

Características de diseño

- S-box de 4 bits: máxima no-linealidad, mínimo hardware
- pLayer: solo cableado (0 GE en hardware)
- Total: 1570 GE - comparable a cifrados de flujo

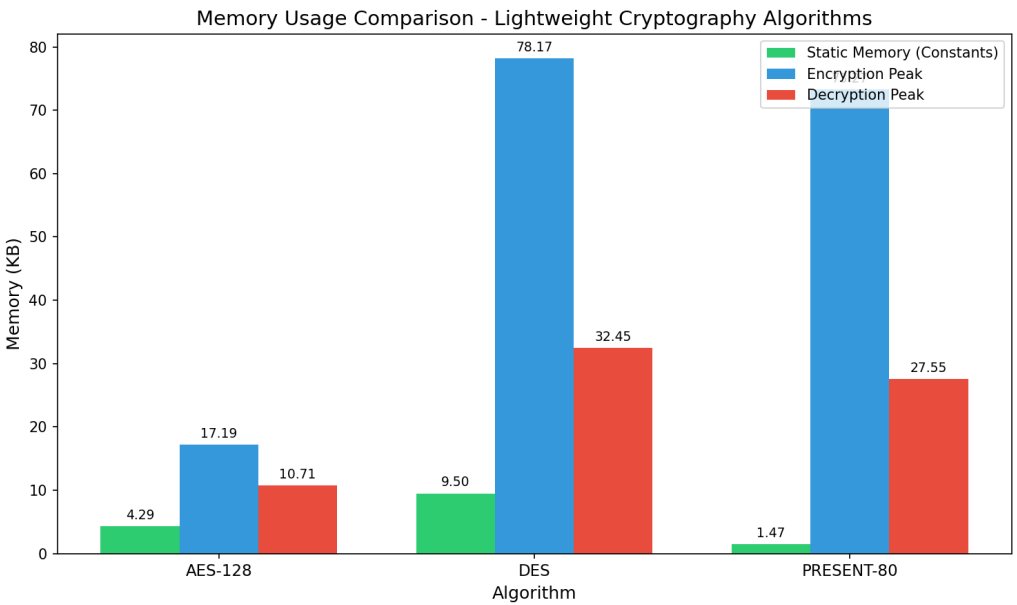
Comparación de Características [2]

Característica	DES	AES-128	PRESENT-80
Año de publicación	1977	2001	2007
Estructura	Feistel	SPN	SPN
Tamaño de clave (bits)	56	128	80
Tamaño de bloque (bits)	64	128	64
Número de rondas	16	10	31
Nivel de seguridad	Bajo	Excelente	Adecuado IoT

Table: Características estáticas de los algoritmos evaluados

SPN = Red de Sustitución-Permutación

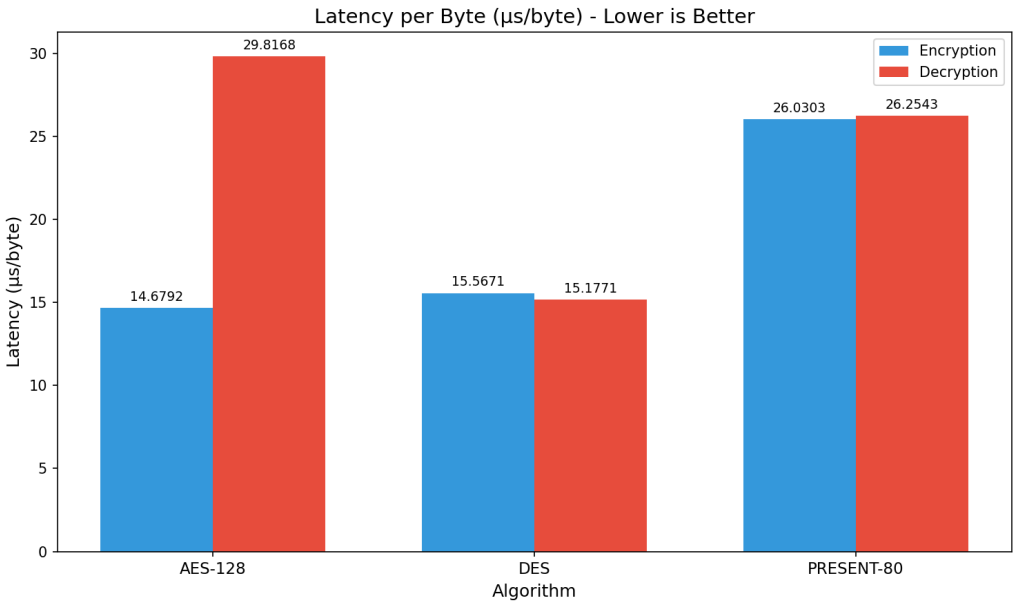
Uso de Memoria [2]



Metodología:

- **Memoria estática**: Tamaño de constantes (S-boxes, tablas de permutación) usando `sys.getsizeof()`
- **Memoria dinámica**: Pico de memoria durante cifrado/descifrado usando `tracemalloc`
- 100 iteraciones por algoritmo

Latencia [3]



Metodología:

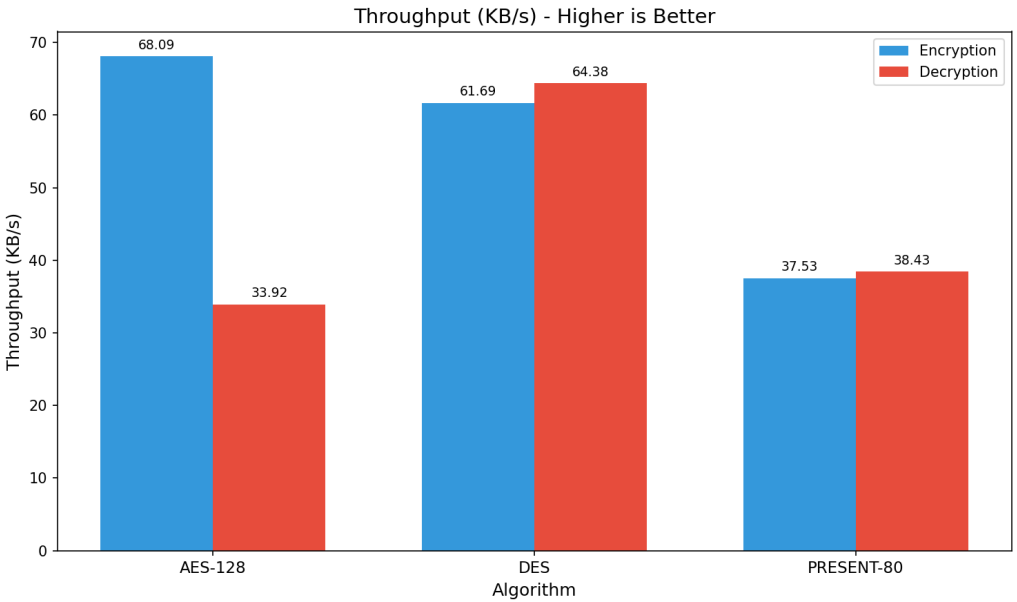
- Latencia por byte (µs/byte)
- 5000 iteraciones por algoritmo
- Fórmula:

$$\text{latencia} = \frac{\text{tiempo}}{n \times \text{bytes}} \times 10^6$$

Interpretación:

- Menor latencia = mejor rendimiento
- AES-128 es el más rápido (optimizado)
- PRESENT-80 prioriza bajo consumo sobre velocidad

Throughput [3]



Metodología:

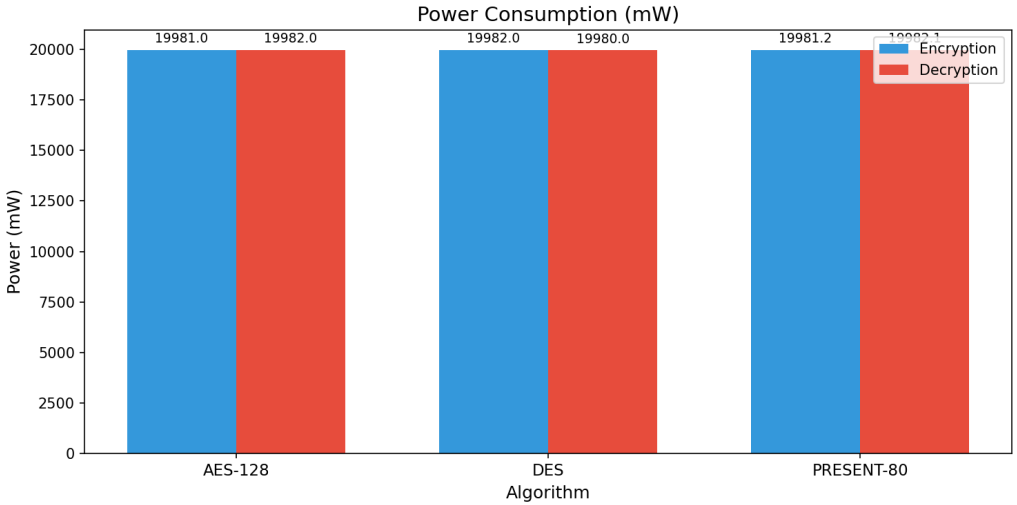
- Throughput en KB/s
- 5000 iteraciones por algoritmo
- Fórmula:

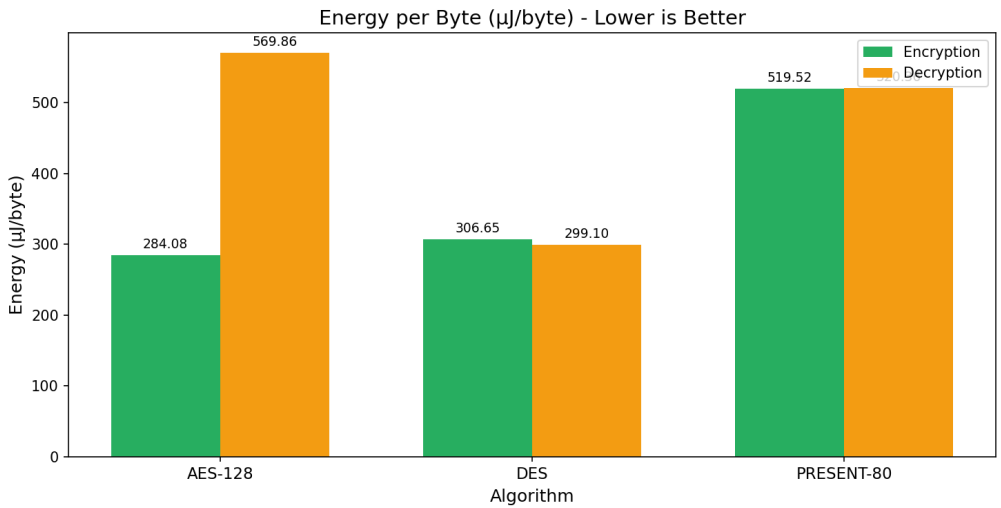
$$\text{throughput} = \frac{n \times \text{bytes}}{\text{tiempo}}$$

Interpretación:

- Mide cuántos bytes se pueden cifrar/descifrar por segundo
- AES-128: 68 KB/s cifrado, 34 KB/s descifrado
- DES: 63 KB/s balanceado
- PRESENT: 38 KB/s

Consumo de Potencia y Energía [3]





Metodología:

- Basado en utilización de CPU
- 5000 iteraciones por algoritmo
- Fórmulas:

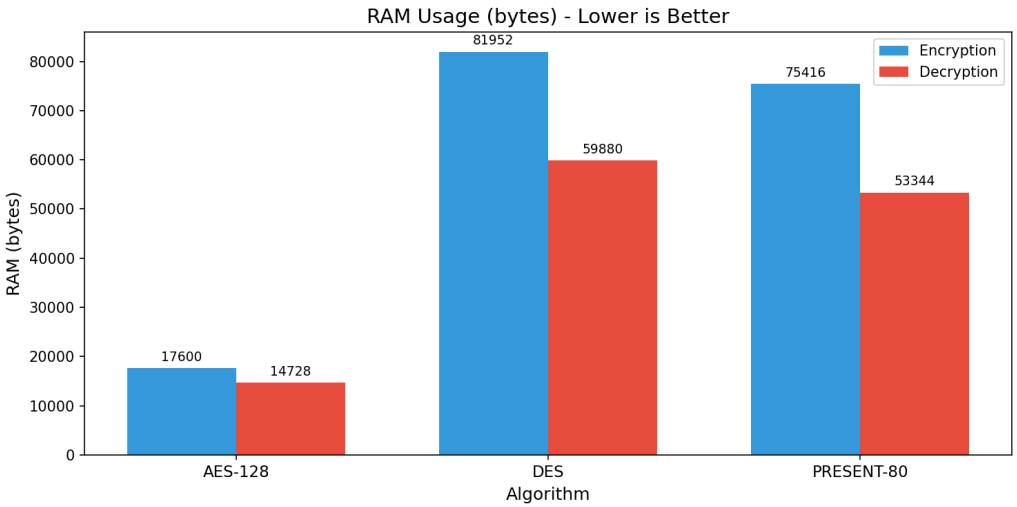
$$\text{Potencia} = V \times I$$

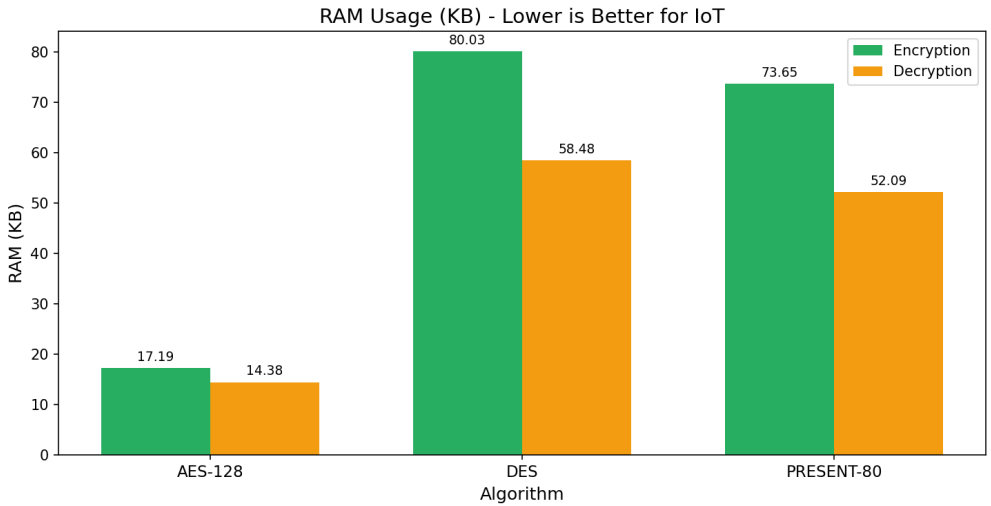
$$\text{Energía} = P \times \text{latencia}$$

Interpretación: Para IoT:

Menor energía = mayor duración de batería

Uso de Memoria RAM [3]





Metodología:

- Memoria pico durante ejecución
- 100 iteraciones por algoritmo
- Medición con `tracemalloc`

Interpretación:

- **AES-128**: Más eficiente en RAM (17 KB enc, 14 KB dec)
- **DES**: Mayor consumo (80 KB enc, 58 KB dec)
- **PRESENT**: Intermedio (74 KB enc, 52 KB dec)

Para IoT: Dispositivos como MSP430 solo tienen 512 bytes de

Ataques Conocidos Más Efectivos

Algoritmo	**Ataque más efectivo**	**Complejidad**	**¿Práctico?**
DES	Fuerza bruta	2^{56}	Sí (horas)
AES-128	Biclique	$2^{126.1}$	No
PRESENT-80	Fuerza bruta	2^{80}	No (aún)

Observaciones

- **DES**: Roto por EFF en 1998 (22 horas con hardware dedicado)
- **AES-128**: Biclique reduce de 2^{128} a $2^{126.1}$ — aún impracticable
- **PRESENT-80**: Margen de 80 bits es adecuado para IoT, pero no para largo plazo

Bibliografía (1/2)

- Bogdanov, A., et al. (2007). *PRESENT: An Ultra-Lightweight Block Cipher*. CHES 2007, LNCS 4727, pp. 450-466.
- Chatziadam, C., et al. (2025). *A Survey on Efficient Lightweight Cryptography*. Technologies, 13(1), 3. MDPI.
- Daemen, J., & Rijmen, V. (2001). *The Design of Rijndael: AES*. Springer-Verlag.
- Daemen, J., & Rijmen, V. (2001). *The Wide Trail Design Strategy*. LNCS 2260, pp. 222-238.
- Luby, M., & Rackoff, C. (1988). *How to Construct Pseudorandom Permutations from PRFs*. SIAM J. Computing, 17(2), pp. 373-386.

Bibliografía (2/2)


National Bureau of Standards (1977). *Data Encryption Standard (DES)*. FIPS PUB 46.


Bernstein, D.J. (2005). *Cache-timing attacks on AES*. `https://cr.yp.to/antiforgery/cachetiming-20050414.pdf`


Soto-Cruz, J., et al. (2024). *Efficient Lightweight Cryptography for Power-Constrained MCUs*. Technologies, 13(1), 3. MDPI.