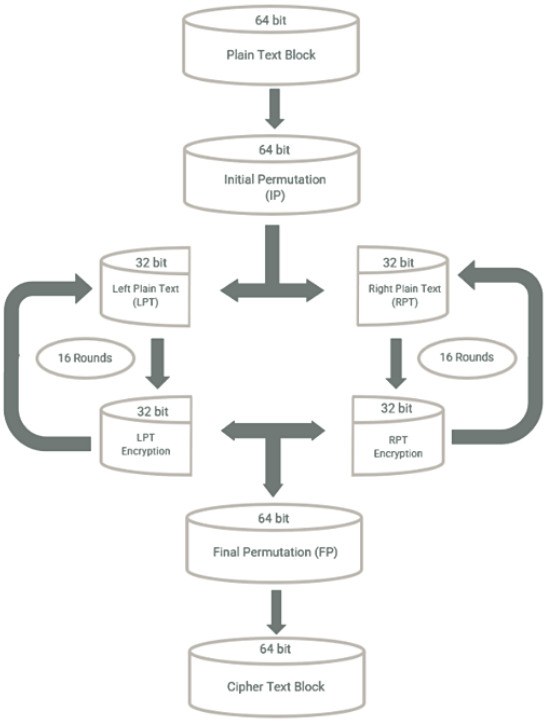
DATA ENCRYPTION STANDARD (DES)



- DES is a symmetric-key block cipher developed by IBM and adopted as a federal standard in 1977. It encrypts data in 64-bit blocks using a 56-bit key through 16 rounds of permutation and substitution operations.
- The algorithm uses a Feistel network structure where the data is split into two halves and processed through expansion, XOR operations, S-box substitutions, and permutations. Each round uses a different 48-bit subkey derived from the original key.
- DES was the dominant encryption standard for over two decades but is now considered insecure due to its short key length. Modern computers can perform brute-force attacks to crack DES encryption in a matter of hours.

Key Specifications

- **Block Size**: 64 bits
- **Key Size**: 56 bits (64 bits with 8 parity bits)
- **Rounds**: 16
- **Structure**: Feistel Network
- **Mode**: Encryption and Decryption use the same algorithm

Created by Nandhan K

DES Algorithm Structure

1. Initial Permutation (IP)

- The 64-bit plaintext block undergoes an initial permutation
- Rearranges the bits according to a fixed permutation table
- Output is divided into two 32-bit halves: L_0 and R_0

2. Key Generation

1. 64-bit key input (8 bits are parity bits)
2. Permuted Choice 1 (PC-1): Reduces to 56 bits
3. Split into two 28-bit halves (C_0 and D_0)
4. For each of 16 rounds:
    - Left circular shift (1 or 2 positions based on round number)
    - Permuted Choice 2 (PC-2): Generates 48-bit subkey

Shift Schedule:

- Rounds 1, 2, 9, 16: 1-bit left shift
- Other rounds: 2-bit left shift

3. Round Function (16 Rounds)

For each round i (1 to 16):

Input: L_{i-1} and R_{i-1} **Process**:

1. **Expansion (E)**: Expand R_{i-1} from 32 bits to 48 bits
2. **XOR with subkey**: Expanded R $\oplus$ K_i (48-bit subkey)
3. **Substitution (S-boxes)**:
    - Divide 48 bits into 8 groups of 6 bits
    - Each group processed by one of 8 S-boxes
    - Each S-box converts 6 bits to 4 bits
    - Output: 32 bits
4. **Permutation (P)**: Rearrange 32 bits
5. **XOR with L**: Result $\oplus$ L_{i-1}

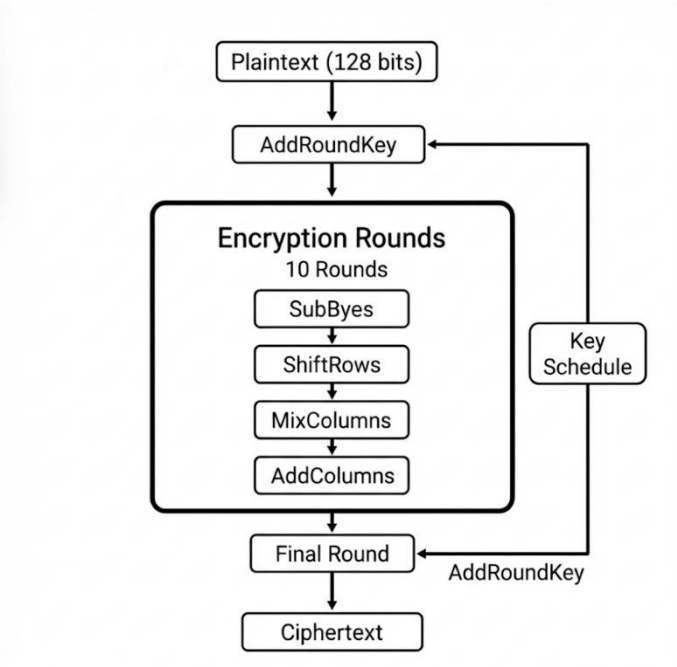
Output:

- $L_i = R_{i-1}$
- $R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$

4. Final Permutation (FP)

- After 16 rounds, swap L_{16} and R_{16}
- Apply Final Permutation (inverse of IP)
- Produces 64-bit ciphertext.

ADVANCED ENCRYPTION STANDARD (AES)



- AES is a symmetric-key block cipher that became the U.S. federal encryption standard in 2001, replacing DES. It encrypts data in 128-bit blocks using key sizes of 128, 192, or 256 bits through 10, 12, or 14 rounds respectively.

- The algorithm uses a Substitution-Permutation Network (SPN) structure consisting of four main operations: SubBytes (substitution), ShiftRows (transposition), MixColumns (mixing), and AddRoundKey (key addition). These operations provide strong confusion and diffusion properties essential for cryptographic security.

- It offers significantly better security than DES due to its larger key size and block size, making brute-force attacks computationally infeasible. AES is implemented in various applications including WiFi security (WPA2/WPA3), VPNs, file encryption, steganography, secure communications, and government/military systems.

Key Specifications

- **Block Size**: 128 bits (fixed)
- **Key Sizes**: 128, 192, or 256 bits
- **Rounds**:
    - AES-128: 10 rounds
    - AES-192: 12 rounds
    - AES-256: 14 rounds
- **Structure**: Substitution-Permutation Network (SPN)

AES State Representation

- Data represented as 4×4 matrix of bytes (State Array)
- Each cell contains 1 byte (8 bits)
- Total: 16 bytes = 128 bits

State Matrix:

[b_0 b_4 b_8 b_{12}]

[b_1 b_5 b_9 b_{13}]

[b_2 b_6 b_{10} b_{14}]

[b_3 b_7 b_{11} b_{15}]

AES Round Structure

Initial Round

1. **AddRoundKey**: XOR state with initial round key

Main Rounds (9, 11, or 13 rounds depending on key size)

1. **SubBytes**: Non-linear byte substitution
2. **ShiftRows**: Cyclically shift rows
3. **MixColumns**: Mix data within columns
4. **AddRoundKey**: XOR with round key

Final Round

1. **SubBytes**
2. **ShiftRows**
3. **AddRoundKey** (no MixColumns in final round)

AES Transformations in Detail

1. SubBytes Transformation

- **Operation**: Byte-by-byte substitution using S-box
- **S-box Construction**:
    i. Calculate multiplicative inverse in GF(2^8)
    ii. Apply affine transformation
- **Properties**: Provides non-linearity (confusion)
- **Implementation**: 16×16 lookup table
- **Inverse**: InvSubBytes uses inverse S-box

2. ShiftRows Transformation

- **Row 0**: No shift
- **Row 1**: Left circular shift by 1 byte
- **Row 2**: Left circular shift by 2 bytes
- **Row 3**: Left circular shift by 3 bytes
- **Purpose**: Provides diffusion across columns
- **Inverse**: InvShiftRows shifts right

Before: After:

$[S_0\ S_4\ S_8\ S_{12}]$ $[S_0\ S_4\ S_8\ S_{12}]$

$[S_1\ S_5\ S_9\ S_{13}]$ $[S_5\ S_9\ S_{13}\ S_1]$

$[S_2\ S_6\ S_{10}\ S_{14}]$ $[S_{10}\ S_{14}\ S_2\ S_6]$

$[S_3\ S_7\ S_{11}\ S_{15}]$ $[S_{15}\ S_3\ S_7\ S_{11}]$

3. MixColumns Transformation

- **Operation**: Matrix multiplication in GF(2^8)
- **Matrix**:

[02 03 01 01]

[01 02 03 01]

[01 01 02 03]

[03 01 01 02]

- **Process**: Each column treated as polynomial and multiplied
- **Purpose**: Provides diffusion within columns
- **Inverse**: InvMixColumns uses different matrix

4. AddRoundKey Transformation

- **Operation**: Bitwise XOR between state and round key
- **Formula**: State ⊕ RoundKey
- **Properties**:
  - Self-inverse operation
  - Same operation for encryption and decryption
- **Key Schedule**: Different round key for each round

AES Key Expansion

Purpose: Generate round keys from cipher key

Process (for AES-128):

1. Initial key = First 4 words (16 bytes)
2. For each subsequent word:
   - Every 4th word undergoes special transformation:
     - **RotWord**: Circular left shift by 1 byte
     - **SubWord**: Apply S-box to each byte
     - **Rcon**: XOR with round constant
   - XOR with word 4 positions earlier

Key Schedule Size:

- AES-128: 44 words (176 bytes)
- AES-192: 52 words (208 bytes)
- AES-256: 60 words (240 bytes)

Differentiate Between DES and AES

Feature	DES	AES
Block Size	64 bits	128 bits
Key Length	56 bits (effectively)	128, 192, or 256 bits
Rounds	16	10, 12, or 14
Structure	Feistel Network	Substitution-Permutation Network
Security	Insecure (brute-forceable)	Secure
Speed	Slower	Faster
Implementation	Simple	More complex but efficient
Current Status	Deprecated	Current standard
Key Schedule	Simple permutations	Complex expansion algorithm
S-boxes	8 S-boxes (6→4 bits)	1 S-box (8→8 bits)