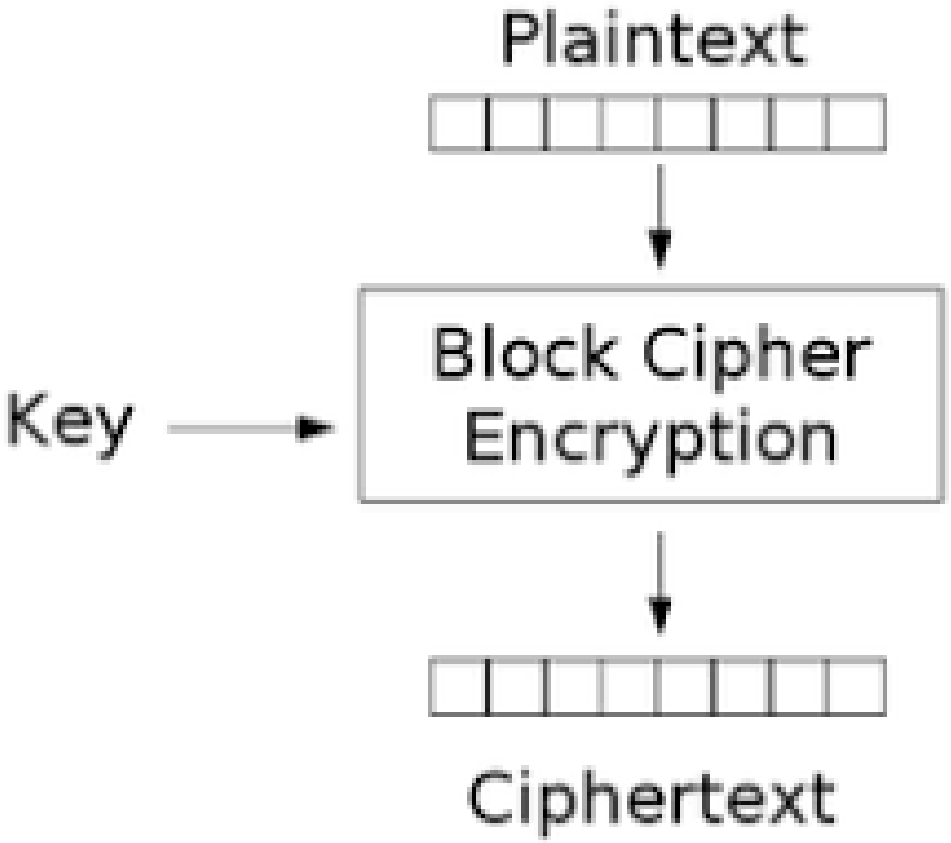
Lecture 12: Block ciphers

Thomas Johansson

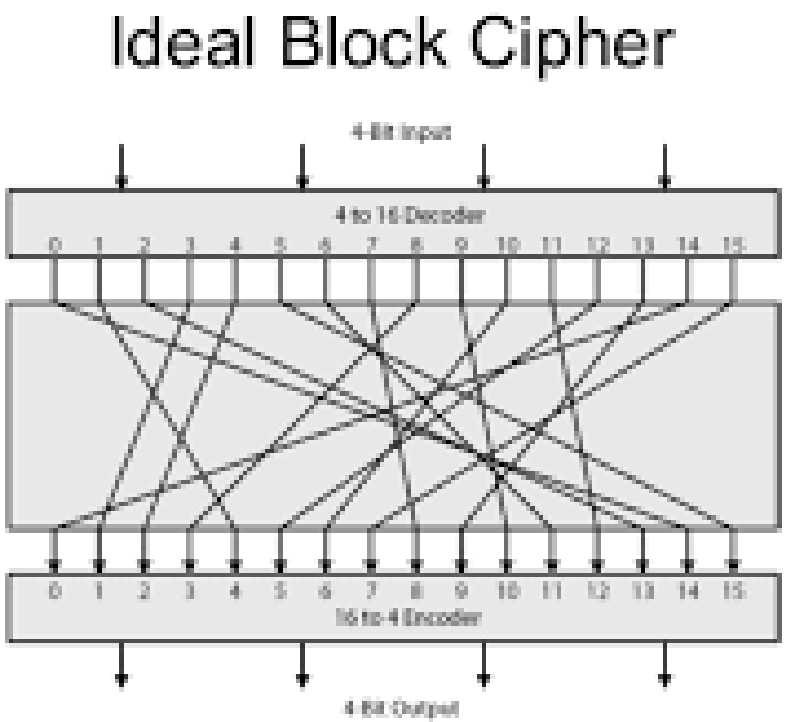
Block ciphers

- A block cipher encrypts a block of plaintext bits x to a block of ciphertext bits y. The transformation is controlled by a secret key K, and it is written $E_K(x) = y$.
- Each key defines a fixed mapping from the plaintext block to the ciphertext block.
- If the block size is b bits the number of different input values to the block ciphers is 2^b.
- a substitution cipher of alphabet size 2^b.
- Only a very small fraction of all possible permutations on alphabet size 2^b will correspond to a certain key.
- *The ideal block cipher:* every key corresponds to a randomly selected permutation

Block cipher



Ideal Block cipher



Iterated ciphers

the way to construct a block cipher...

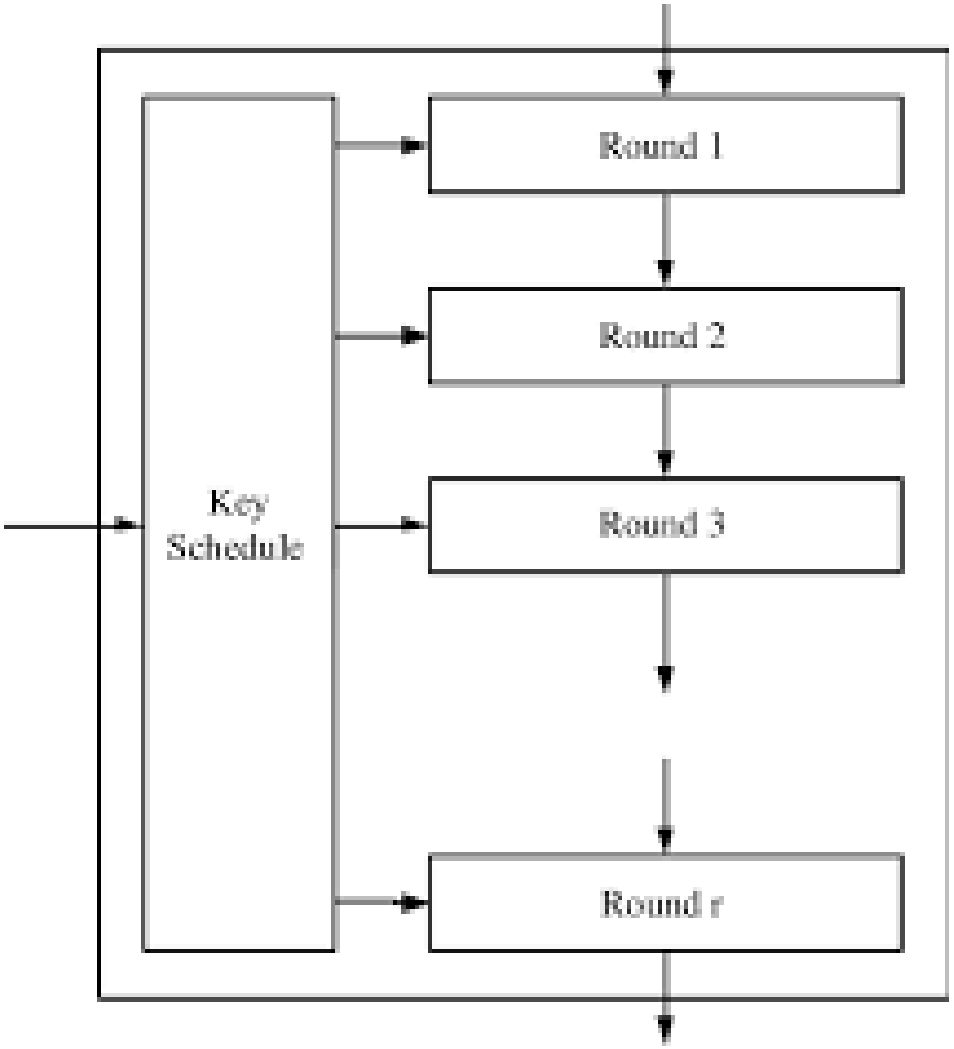
- In an iterated cipher we apply a simple encryption function iteratively a number of times (*rounds*), say N.
- The simple function applied is called the *round function*.
- It often uses a *round key*, which is derived from the key K. The way the round keys are derived from K is called the *key schedule*.

Iterated ciphers

- Let w_i, $i = 0..N$, denote intermediate values in the implementation process of the block cipher, where $w_0 = x$ is the plaintext block.
- $$\begin{aligned} w_0 &= x \\ w_1 &= h(w_0, K_1) \\ w_2 &= h(w_1, K_2) \\ &\vdots \\ w_{N-1} &= h(w_{N-2}, K_{N-1}) \\ w_N &= h(w_{N-1}, K_N) \end{aligned},$$

  and $w_N = y$ is the ciphertext block.
- $h(w_{i-1}, K_i)$ denotes the round function and K_i is the round key used in round i.

Iterated block cipher



Efficient decryption

- For an iterated cipher this property can be reduced to the property of efficiently being able to invert the round function $h(w_{i-1}, K_i)$,
- We need a function $h^{-1}()$ such that

$$h^{-1}(h(w_{i-1}, K_i), K_i) = w_{i-1}.$$

- Then the decryption function $D_K(y)$ for an iterated cipher can be implemented in the same iterated style as the encryption with the difference that the round keys are used in reverse order,

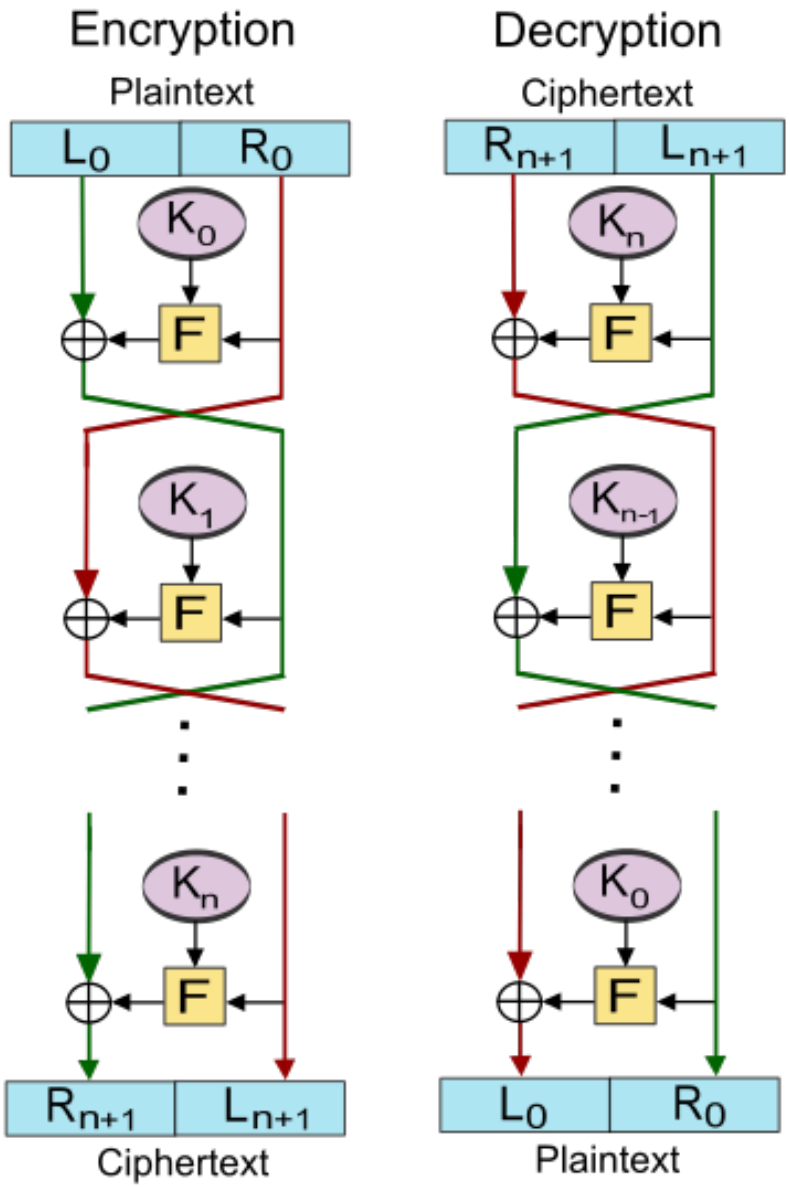
$$\begin{array}{rcc} w_N & = & y \\ w_{N-1} & = & h^{-1}(w_N, K_N) \\ w_{N-2} & = & h^{-1}(w_{N-1}, K_{N-1}) \\ \vdots & & \\ w_1 & = & h^{-1}(w_2, K_2) \\ w_0 & = & h^{-1}(w_1, K_1) \end{array},$$

and $w_0 = x$.

First type of iterated cipher: Feistel cipher

- we split the intermediate values w_i in a left half and a right half, denoted
$$w_i = (L_i, R_i).$$
- the round function $(L_i, R_i) = h(L_{i-1}, R_{i-1}, K_i)$ is implemented as
$$\begin{aligned} L_i &= R_{i-1}, \\ R_i &= L_{i-1} \oplus f(R_{i-1}, K_i), \end{aligned}$$
where $f(R_{i-1}, K_i)$ can be any function.
- decryption function $h^{-1}()$, implemented as
$$\begin{aligned} R_{i-1} &= L_i, \\ L_{i-1} &= R_i \oplus f(R_{i-1}, K_i), \end{aligned}$$
and (L_0, R_0) gives back x.

Feistel cipher

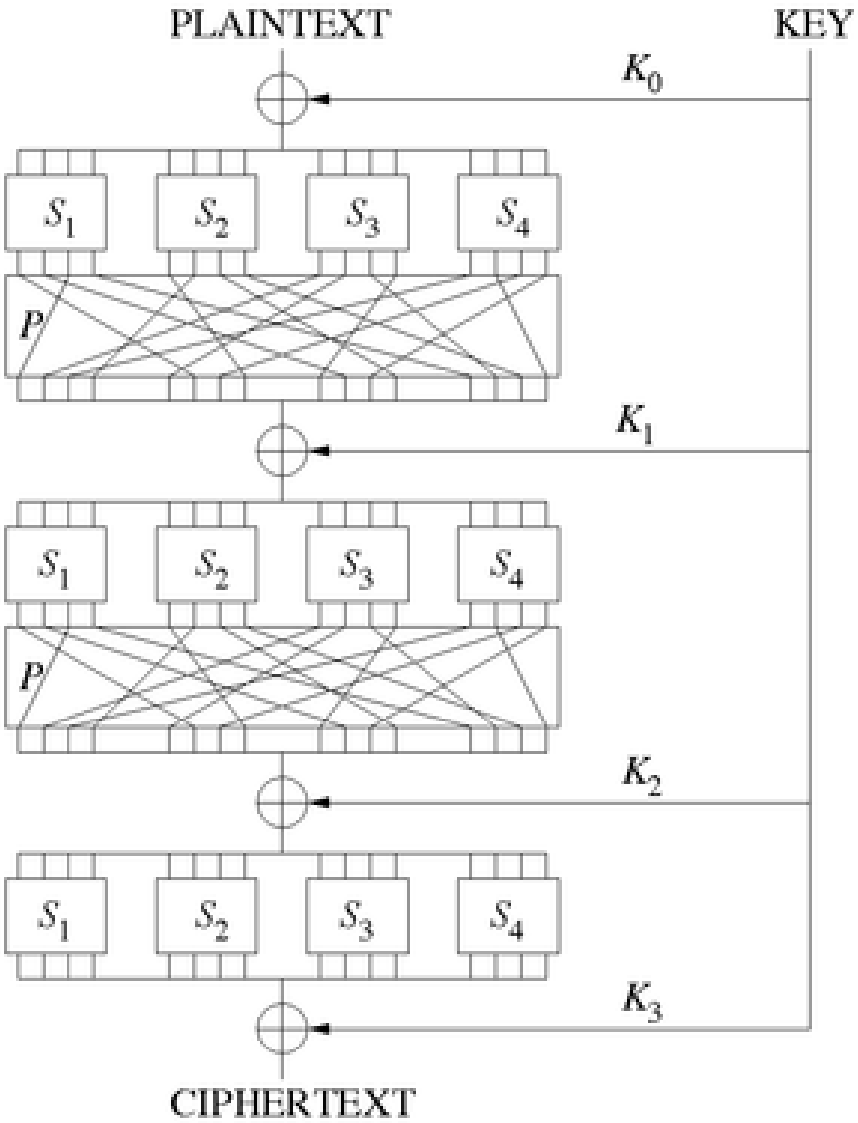


The old world standard: DES
DES was approved as a federal standard in November 1976, and published on 15 January 1977 as FIPS PUB 46, authorized for use on all unclassified data. It was subsequently reaffirmed as the standard in 1983, 1988 (revised as FIPS-46-1), 1993 (FIPS-46-2), and again in 1999 (FIPS-46-3), the latter prescribing "Triple DES" (see below). On 26 May 2002, DES was finally superseded by the Advanced Encryption Standard. (AES)

Second type of iterated cipher: SP network

- An SP network consists of a mix between many substitutions (substitution ciphers) and permutations or linear transformations (like a transposition cipher or affine cipher).
- The substitutions, also called S-boxes, are taken over a small alphabet, typically 4, 6, or 8 bits input.
- The round key is added to the input to the round function in some simple way, usually bitwise xor.
- AES

SP network


PLAINTEXT
KEY
K_0
S_1 S_2 S_3 S_4
P
K_1
S_1 S_2 S_3 S_4
P
K_2
S_1 S_2 S_3 S_4
K_3
CIPHERTEXT

Modes of Operation

Decide on how to encrypt a sequence of plaintext bits

- Split the plaintext sequence

$$x_1, x_2, \ldots, x_N, x_{N+1}, \ldots$$

  in blocks and encrypt each block.
- Let $X_1 = (x_1, x_2, \ldots, x_N)$ be the first block consisting of the first N bits, $X_2 = (x_{N+1}, x_{N+2}, \ldots, x_{2N})$ be the second block
- Encryption is done blockwise, i.e.,

$$C_i = E_K(X_i), i = 1, 2, \ldots.$$

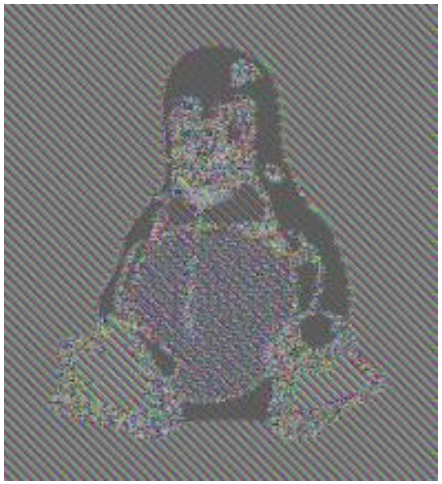
- ECB (*Electronic codebook*)

ECB

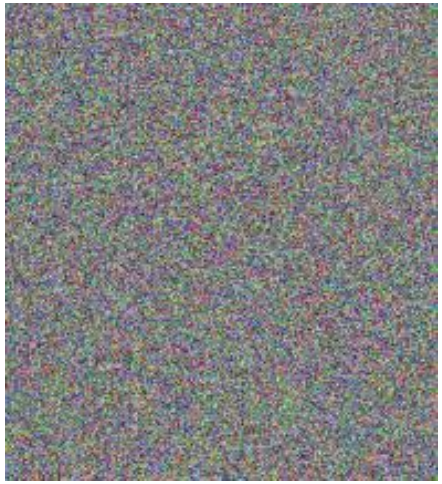
- A problem with the ECB mode: If two plaintext blocks are the same, then the two corresponding ciphertext blocks are also the same.
- We need better ways...

The problem with ECB mode

The original image



ECB encrypted



Better mode of operation

CBC and counter mode

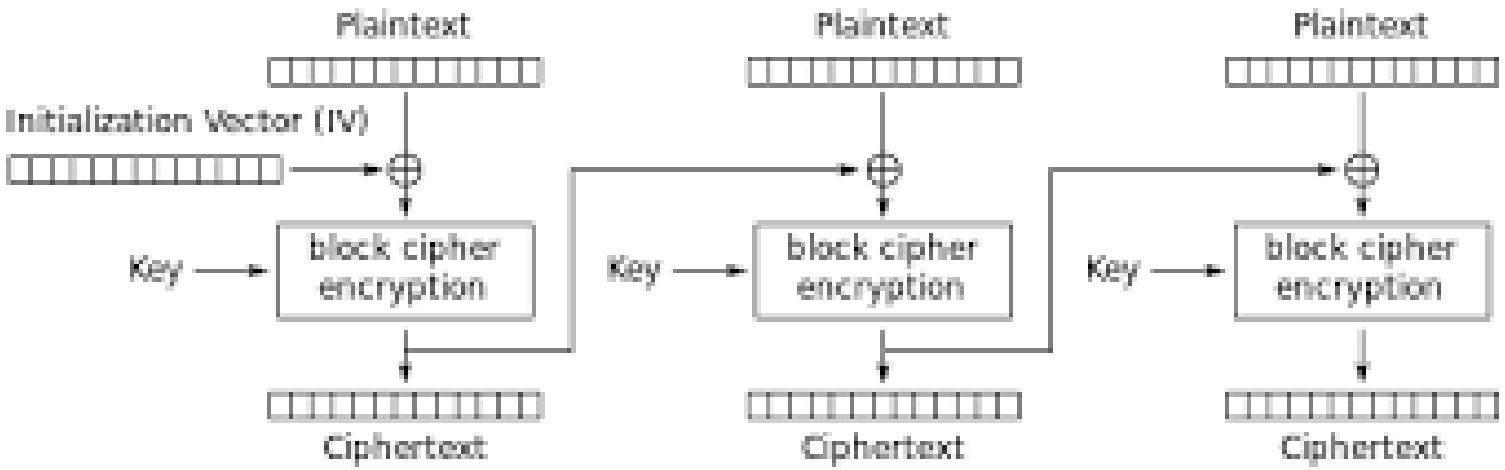
- CBC mode (Cipher block chaining mode). It follows the rule

$$Y_i = E_K(Y_{i-1} \oplus X_i),$$

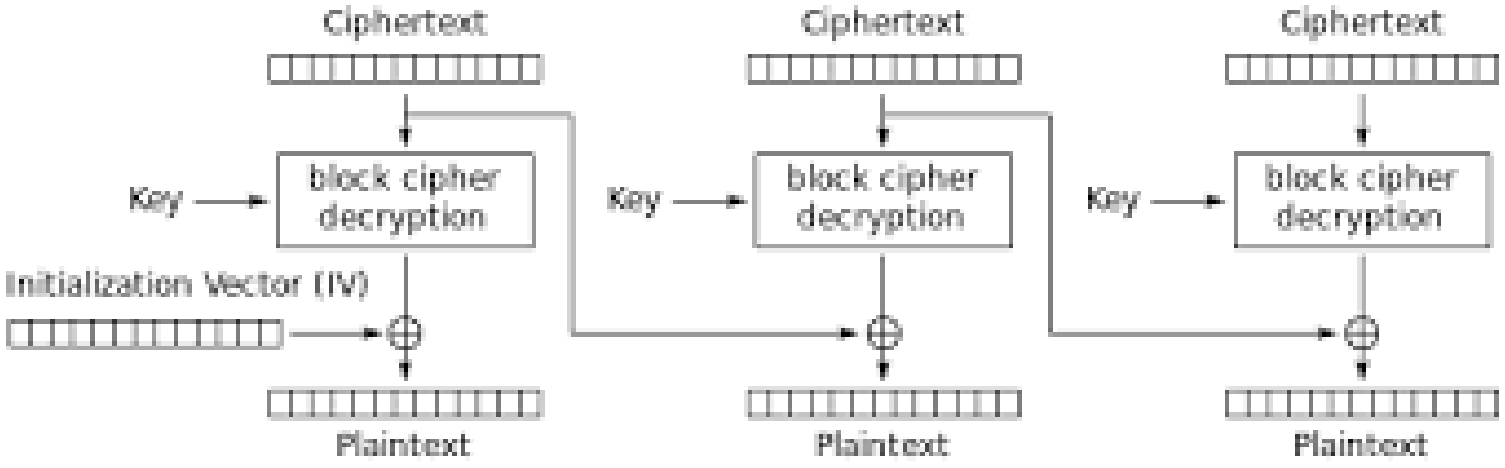
  where Y_0 is some fixed and known initial value. This will produce the output sequence $Y_1, Y_2, \ldots$.
- Verify how to decrypt!
- Another mode of operation that will turn the block cipher into a stream cipher is *counter mode*. The keystream sequence is given by

$$E_K(0), E_K(1), E_K(2), \ldots.$$

CBC mode

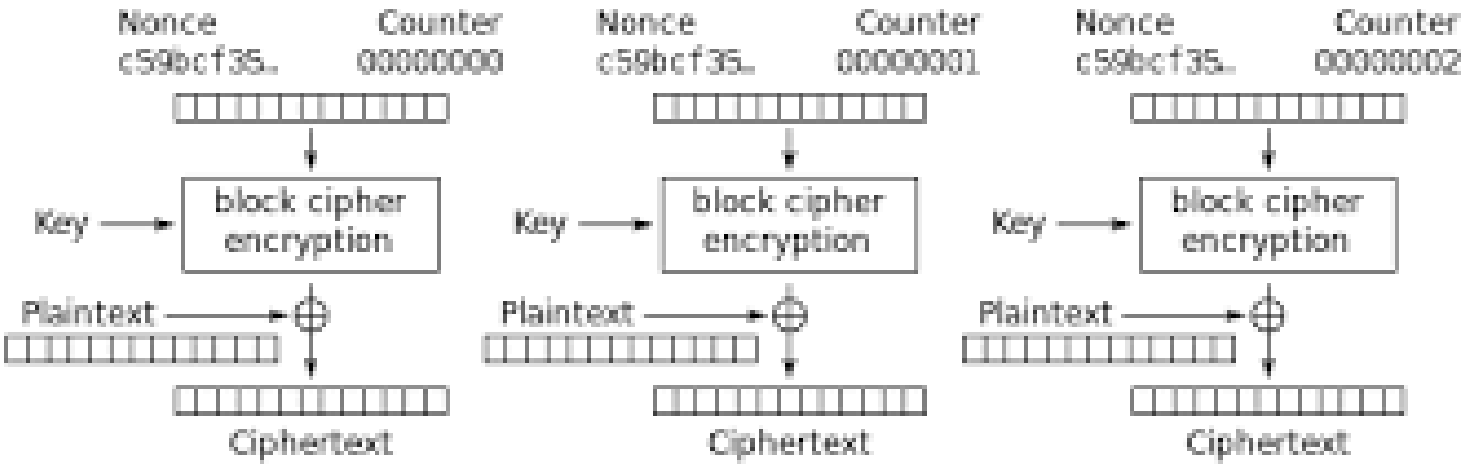


Cipher Block Chaining (CBC) mode encryption



Cipher Block Chaining (CBC) mode decryption

Counter mode



Counter (CTR) mode encryption

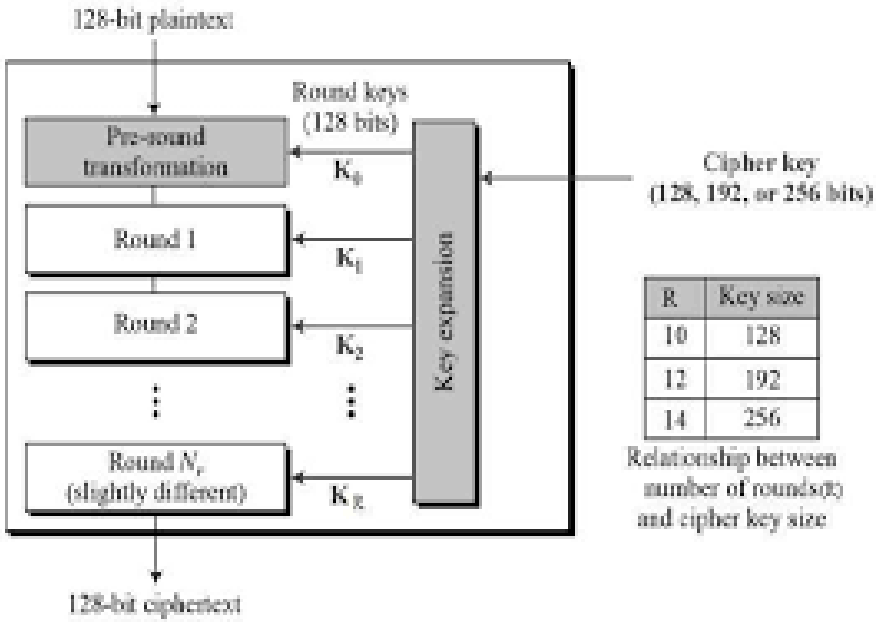
The world standard AES

- The Advanced Encryption Standard (AES) is a specification established by NIST in 2001.
- AES has been adopted by the U.S. government and is now used worldwide.
- A five-year standardization process with fifteen competing designs

AES

- SP network
- designed to be fast in both software and hardware
- a fixed block size of 128 bits, and a key size of 128, 192, or 256 bits.
- 10,12, or 14 rounds depending on key size

High-level description of AES



R	Key size
10	128
12	192
14	256

Relationship between number of rounds(R) and cipher key size

High-level description of AES

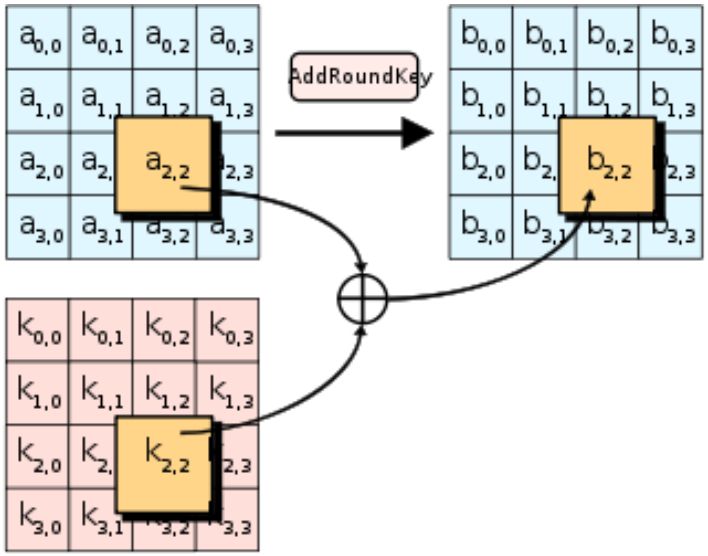
AddRoundKey: each byte of the state is combined with the round key using bitwise xor

Iterate r Rounds

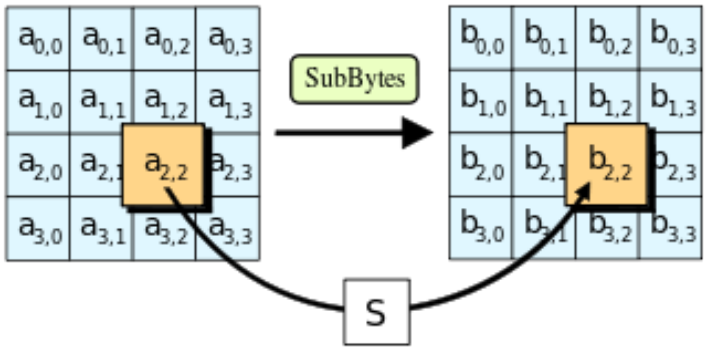
- **SubBytes:** non-linear substitution step where each byte is replaced with another according to a lookup table.
- **ShiftRows:** transposition step where each row of the state is shifted cyclically a certain number of steps.
- **MixColumns:** mixing operation which operates on the columns of the state, combining the four bytes in each column.
- **AddRoundKey:**

(Final Round is without MixColumns)

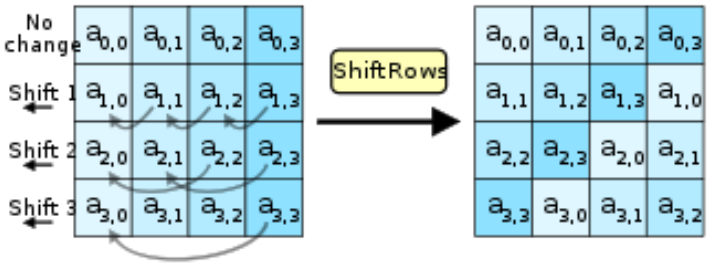
AES - AddRoundKey



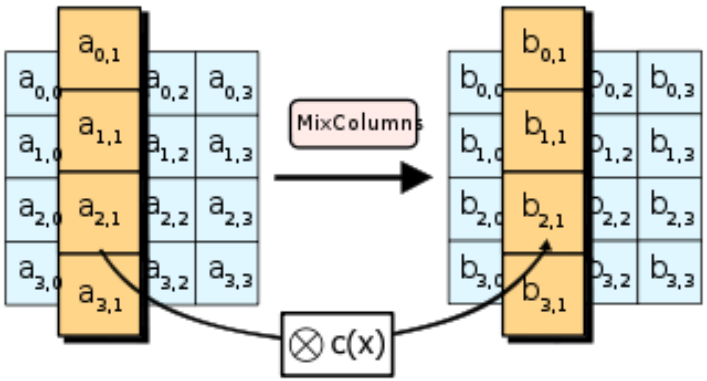
AES - SubBytes



AES - ShiftRows



AES - MixColumns



Attacks on AES

- the best known attacks were on 7 rounds for 128-bit keys, 8 rounds for 192-bit keys, and 9 rounds for 256-bit keys.
- related-key attack on the 192-bit and 256-bit versions of AES. a complexity of 2^{96} for one out of every 2^{35} keys.
- The first key-recovery attacks on full AES were published in 2011. It requires $2^{126.1}$ operations to recover an AES-128 key.
- Side-channel attacks are still possible... cache-timing attacks, differential fault analysis

Conclusions on AES

- it is the standard symmetric cipher used everywhere
- it has been analyzed for 20 years now and it looks very secure...
- In many CPUs (Intel, AMD, Arm,..) there are special instructions that compute the full AES round function. Extremely fast in software!
- Many other primitives rely on the security of AES (a proof that the primitive is as secure as AES)

Attacks: Linear cryptanalysis

One of the most powerful general ideas for cryptanalysis of block ciphers is called *linear cryptanalysis*.

- Assume the cipher consist only of linear operations. Then there would be a linear relationship between the plaintext bits and the ciphertext bits.
- Then for each ciphertext bit c_i we would be able to write one equation

$$c_i = \sum_{j=1}^{N} d_{ij} x_j + e_i(K), \tag{1}$$

where $e(K)$ denotes the direct contribution from the key K.

Linear cryptanalysis

- The cipher contains nonlinear operations and the above is not true.
- Introduce linear approximations of nonlinear parts
- A linear approximation is when we replace a nonlinear function with a linear one. When we do so, we will not always get the correct value.
- Probability p that the linear function gives the same value as the original nonlinear one = a measure of the effectiveness of a linear approximation.

Linear cryptanalysis

- Our goal: find a linear expression through the whole cipher, i.e., a sum of expressions of the form (1) such that the probability p is as far away from 0.5 as possible.
- Consider a known-ciphertext attack and write the key contribution $e(K)$ as a sum of known values,

$$e(K) = \sum_{j=1}^{N} d_j c_j \oplus \sum_{j=1}^{N} d'_j x_j.$$

  Note that this expression holds with probability p.
- It is clear that if we evaluate the right hand side above for different plaintext-ciphertext pairs, we will eventually know the value of $e(K)$.

Linear cryptanalysis

- The hard problem in linear cryptanalysis is to find the best linear approximation.
- Introducing many, but still as few as possible, linear approximations along a *trail* in the cipher.
- A trail corresponds to a chain of linear expressions (using intermediate values) through the cipher connecting the plaintext bits and the ciphertext bits.

Project 4

- Notation: P plaintext block, C ciphertext block, and

$$P[i_1, i_2, \ldots, i_n] = \bigoplus_{t=1}^{n} p_{i_t},$$

  etc.

- The purpose of linear cryptanalysis is to find the following "effective" linear expression for a given cipher algorithm:

$$P[i_1, i_2, \ldots, i_a] \oplus C[j_1, j_2, \ldots, j_b] = K[k_1, k_2, \ldots, k_c], \qquad (2)$$

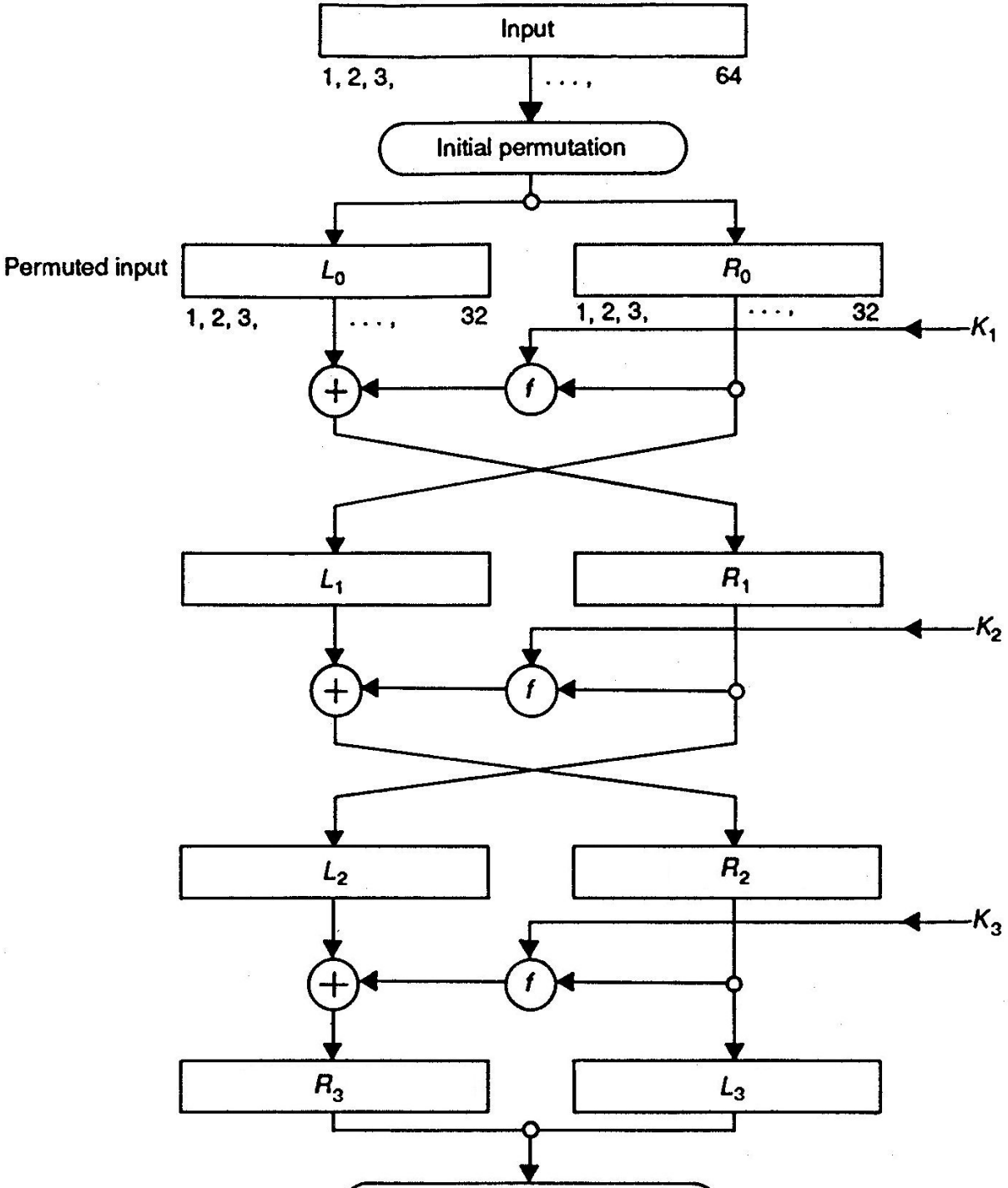
- holds with probability $p \neq 1/2$ for randomly given plaintext P and the corresponding ciphertext C.

Project 4

- Once we succeed in reaching an effective linear expression, it is possible to determine one key bit $K[k_1, k_2, \ldots, k_c]$.
- Define a noise random variable to be:

$$N = P[i_1, i_2, \ldots, i_a] \oplus C[j_1, j_2, \ldots, j_b] \oplus K[k_1, k_2, \ldots, k_c], \qquad (3)$$

which has some bias $P(N = 0) = p$.


Input
1, 2, 3, . . . , 64
Initial permutation
Permuted input
L_0
R_0
1, 2, 3, . . . , 32
1, 2, 3, . . . , 32
K_1
f
L_1
R_1
K_2
f
L_2
R_2
K_3
f
R_3
L_3

Your task: Find the best linear approximation

Hints:

- First find the relation

$$R_1[3, 8, 14, 25] \oplus P[\ldots] = K[\ldots],$$

and then

$$R_1[3, 8, 14, 25] \oplus C[\ldots] = K[\ldots].$$

The sum gives us the required relation.

- The best linear approximation for the f-function is found to be:

$$B[26] = f(R_x, K_x)[3, 8, 14, 25], \tag{4}$$

i.e., the noise $N = B[26] \oplus f(R_x, K_x)[3, 8, 14, 25]$ has the maximum bias and the corresponding probability is $P(N = 0) \approx 0.19$.

Your task: Simulate to check your linear approximation

- For the linear approximation that you have derived, find the bias by simulation.
- Make a sufficient number ($\approx 10^5$) of random selections of (P, K), and calculate C. Compute an estimate of p.