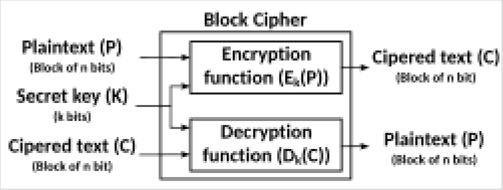
WIKIPEDIA
The Free Encyclopedia

# Block cipher

In cryptography, a **block cipher** is a deterministic algorithm that operates on fixed-length groups of bits, called *blocks*. Block ciphers are the elementary building blocks of many cryptographic protocols. They are ubiquitous in the storage and exchange of data, where such data is secured and authenticated via encryption.

A block cipher uses blocks as an unvarying transformation. Even a secure block cipher is suitable for the encryption of only a single block of data at a time, using a fixed key. A multitude of modes of operation have been designed to allow their repeated use in a secure way to achieve the security goals of confidentiality and authenticity. However, block ciphers may also feature as building blocks in other cryptographic protocols, such as universal hash functions and pseudorandom number generators.

## Definition



Block diagram of cipher block showing its inputs, outputs and components.

A block cipher consists of two paired algorithms, one for encryption, $E$, and the other for decryption, $D$.[1] Both algorithms accept two inputs: an input block of size $n$ bits and a key of size $k$ bits; and both yield an $n$-bit output block. The decryption algorithm $D$ is defined to be the inverse function of encryption, i.e., $D = E^{-1}$. More formally,[2][3] a block cipher is specified by an encryption function

$$E_K(P) := E(K,P) : \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n,$$

which takes as input a key $K$, of bit length $k$ (called the *key size*), and a bit string $P$, of length $n$ (called the *block size*), and returns a string $C$ of $n$ bits. $P$ is called the plaintext, and $C$ is termed the ciphertext. For each $K$, the function $E_K(P)$ is required to be an invertible mapping on $\{0,1\}^n$. The inverse for $E$ is defined as a function

$$E_K^{-1}(C) := D_K(C) = D(K,C) : \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n,$$

taking a key $K$ and a ciphertext $C$ to return a plaintext value $P$, such that

$$\forall P : D_K(E_K(P)) = P.$$

For example, a block cipher encryption algorithm might take a 128-bit block of plaintext as input, and output a corresponding 128-bit block of ciphertext. The exact transformation is controlled using a second input – the secret key. Decryption is similar: the decryption algorithm takes, in this example, a 128-bit block of ciphertext together with the secret key, and yields the original 128-bit block of plain text.[4]

For each key $K$, $E_K$ is a permutation (a bijective mapping) over the set of input blocks. Each key selects one permutation from the set of $(2^n)!$ possible permutations.[5]

# History

The modern design of block ciphers is based on the concept of an iterated product cipher. In his seminal 1949 publication, *Communication Theory of Secrecy Systems*, Claude Shannon analyzed product ciphers and suggested them as a means of effectively improving security by combining simple operations such as substitutions and permutations.[6] Iterated product ciphers carry out encryption in multiple rounds, each of which uses a different subkey derived from the original key. One widespread implementation of such ciphers named a Feistel network after Horst Feistel is notably implemented in the DES cipher.[7] Many other realizations of block ciphers, such as the AES, are classified as substitution–permutation networks.[8]

The root of all cryptographic block formats used within the Payment Card Industry Data Security Standard (PCI DSS) and American National Standards Institute (ANSI) standards lies with the Atalla Key Block (AKB), which was a key innovation of the Atalla Box, the first hardware security module (HSM). It was developed in 1972 by Mohamed M. Atalla, founder of Atalla Corporation (now Utimaco Atalla), and released in 1973. The AKB was a key block, which is required to securely interchange symmetric keys or PINs with other actors in the banking industry. This secure interchange is performed using the AKB format.[9] The Atalla Box protected over 90% of all ATM networks in operation as of 1998,[10] and Atalla products still secure the majority of the world's ATM transactions as of 2014.[11]

The publication of the DES cipher by the United States National Bureau of Standards (subsequently the U.S. National Institute of Standards and Technology, NIST) in 1977 was fundamental in the public understanding of modern block cipher design. It also influenced the academic development of cryptanalytic attacks. Both differential and linear cryptanalysis arose out of studies on DES design. As of 2016, there is a palette of attack techniques against which a block cipher must be secure, in addition to being robust against brute-force attacks.

# Design

## Iterated block ciphers

Most block cipher algorithms are classified as *iterated block ciphers* which means that they transform fixed-size blocks of plaintext into identically sized blocks of ciphertext, via the repeated application of an invertible transformation known as the *round function*, with each iteration referred to as a *round*.[12]

Usually, the round function *R* takes different *round keys* $K_i$ as a second input, which is derived from the original key:[13]

$$M_i = R_{K_i}(M_{i-1})$$

where $M_0$ is the plaintext and $M_r$ the ciphertext, with *r* being the number of rounds.

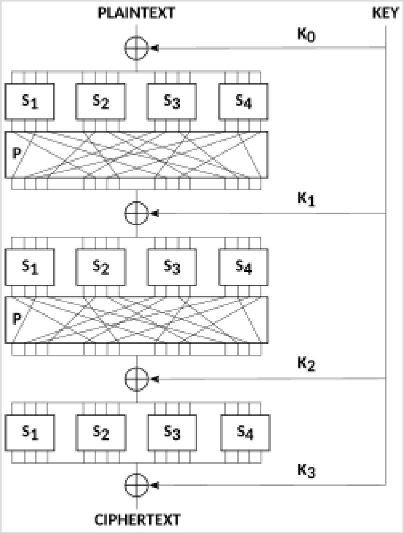
Frequently, key whitening is used in addition to this. At the beginning and the end, the data is modified with key material (often with XOR):

$$\begin{aligned}M_0 &= M \oplus K_0\\ M_i &= R_{K_i}(M_{i-1})\ ;\ i = 1\dots r\\ C &= M_r \oplus K_{r+1}\end{aligned}$$

Given one of the standard iterated block cipher design schemes, it is fairly easy to construct a block cipher that is cryptographically secure, simply by using a large number of rounds. However, this will make the cipher inefficient. Thus, efficiency is the most important additional design criterion for professional ciphers. Further, a good block cipher is designed to avoid side-channel attacks, such as branch prediction and input-dependent memory accesses that might leak secret data via the cache state or the execution time. In addition, the cipher should be concise, for small hardware and software implementations.

## Substitution–permutation networks

One important type of iterated block cipher known as a *substitution–permutation network (SPN)* takes a block of the plaintext and the key as inputs and applies several alternating rounds consisting of a substitution stage followed by a permutation stage—to produce each block of ciphertext output.[14] The non-linear substitution stage mixes the key bits with those of the plaintext, creating Shannon's *confusion*. The linear permutation stage then dissipates redundancies, creating *diffusion*.[15][16]



A sketch of a substitution–permutation network with 3 rounds, encrypting a plaintext block of 16 bits into a ciphertext block of 16 bits. The S-boxes are the $S_i$, the P-boxes are the same $P$, and the round keys are the $K_i$.

A *substitution box (S-box)* substitutes a small block of input bits with another block of output bits. This substitution must be one-to-one, to ensure invertibility (hence decryption). A secure S-box will have the property that changing one input bit will change about half of the output bits on average, exhibiting what is known as the avalanche effect—i.e. it has the property that each output bit will depend on every input bit.[17]

A *permutation box (P-box)* is a permutation of all the bits: it takes the outputs of all the S-boxes of one round, permutes the bits, and feeds them into the S-boxes of the next round. A good P-box has the property that the output bits of any S-box are distributed to as many S-box inputs as possible.[18]

At each round, the round key (obtained from the key with some simple operations, for instance, using S-boxes and P-boxes) is combined using some group operation, typically XOR.

Decryption is done by simply reversing the process (using the inverses of the S-boxes and P-boxes and applying the round keys in reversed order).[19]

## Feistel ciphers

In a *Feistel cipher*, the block of plain text to be encrypted is split into two equal-sized halves. The round function is applied to one half, using a subkey, and then the output is XORed with the other half. The two halves are then swapped.[20]

Let $\mathrm{F}$ be the round function and let $K_0, K_1, \ldots, K_n$ be the sub-keys for the rounds $0, 1, \ldots, n$ respectively.

Then the basic operation is as follows:[20]

Split the plaintext block into two equal pieces, ($L_0$, $R_0$)

For each round $i = 0, 1, \ldots, n$, compute

$$L_{i+1} = R_i$$
$$R_{i+1} = L_i \oplus \mathrm{F}(R_i, K_i).$$

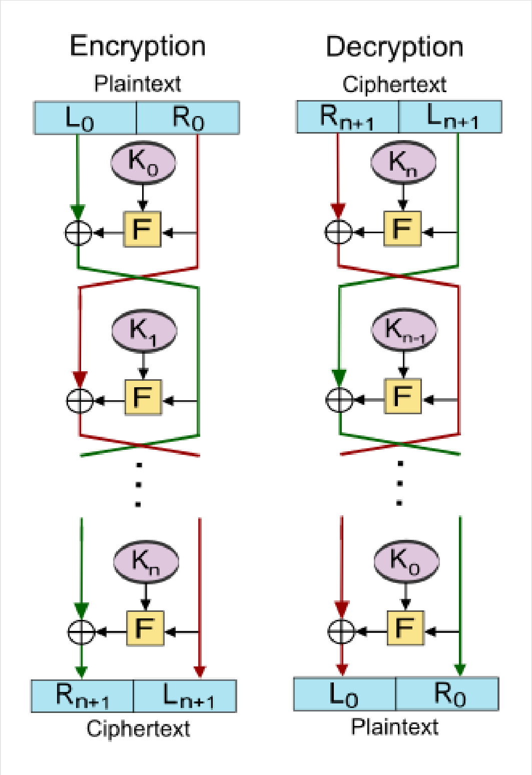
Then the ciphertext is $(R_{n+1}, L_{n+1})$.

The decryption of a ciphertext $(R_{n+1}, L_{n+1})$ is accomplished by computing for $i = n, n-1, \ldots, 0$

$$R_i = L_{i+1}$$
$$L_i = R_{i+1} \oplus \mathrm{F}(L_{i+1}, K_i).$$

Then $(L_0, R_0)$ is the plaintext again.

One advantage of the Feistel model compared to a substitution–permutation network is that the round function $\mathrm{F}$ does not have to be invertible.[21]



Many block ciphers, such as DES and Blowfish utilize structures known as *Feistel ciphers*

## Lai–Massey ciphers

The Lai–Massey scheme offers security properties similar to those of the Feistel structure. It also shares the advantage that the round function $\mathrm{F}$ does not have to be invertible. Another similarity is that it also splits the input block into two equal pieces. However, the round function is applied to the difference between the two, and the result is then added to both half blocks.

Let $\mathrm{F}$ be the round function and $\mathrm{H}$ a half-round function and let $K_0, K_1, \ldots, K_n$ be the sub-keys for the rounds $0, 1, \ldots, n$ respectively.

Then the basic operation is as follows:

Split the plaintext block into two equal pieces, ($L_0$, $R_0$)

For each round $i = 0, 1, \ldots, n$, compute

$$(L'_{i+1}, R'_{i+1}) = \mathrm{H}(L'_i + T_i, R'_i + T_i),$$

where $T_i = \mathrm{F}(L'_i - R'_i, K_i)$ and $(L'_0, R'_0) = \mathrm{H}(L_0, R_0)$

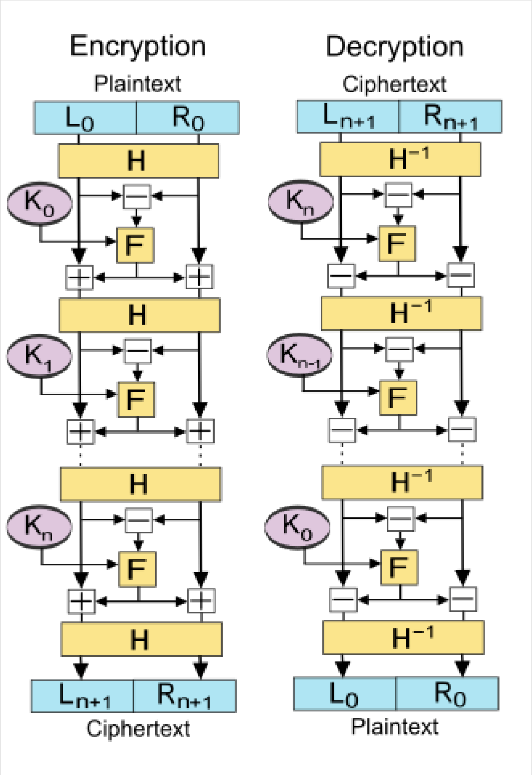
Then the ciphertext is $(L_{n+1}, R_{n+1}) = (L'_{n+1}, R'_{n+1})$.

The decryption of a ciphertext $(L_{n+1}, R_{n+1})$ is accomplished by computing for $i = n, n-1, \ldots, 0$

$$(L'_i, R'_i) = \mathrm{H}^{-1}(L'_{i+1} - T_i, R'_{i+1} - T_i)$$

where $T_i = \mathrm{F}(L'_{i+1} - R'_{i+1}, K_i)$ and $(L'_{n+1}, R'_{n+1}) = \mathrm{H}^{-1}(L_{n+1}, R_{n+1})$

Then $(L_0, R_0) = (L'_0, R'_0)$ is the plaintext again.



The Lai–Massey scheme. The archetypical cipher utilizing it is IDEA.

## Operations

### ARX (add–rotate–XOR)

Many modern block ciphers and hashes are **ARX** algorithms—their round function involves only three operations: (A) modular addition, (R) rotation with fixed rotation amounts, and (X) XOR. Examples include ChaCha20, Speck, XXTEA, and BLAKE. Many authors draw an ARX network, a kind of data flow diagram, to illustrate such a round function.[22]

These ARX operations are popular because they are relatively fast and cheap in hardware and software, their implementation can be made extremely simple, and also because they run in constant time, and therefore are immune to timing attacks. The rotational cryptanalysis technique attempts to attack such round functions.

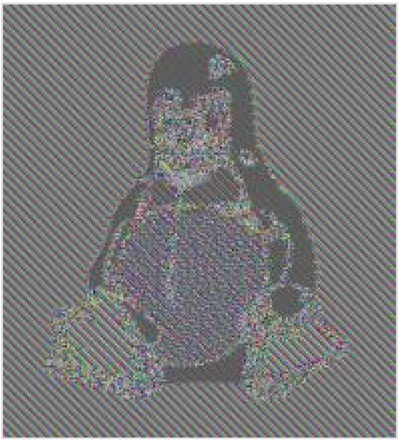
### Other operations

Other operations often used in block ciphers include data-dependent rotations as in RC5 and RC6, a substitution box implemented as a lookup table as in Data Encryption Standard and Advanced Encryption Standard, a permutation box, and multiplication as in IDEA.

# Modes of operation

A block cipher by itself allows encryption only of a single data block of the cipher's block length. For a variable-length message, the data must first be partitioned into separate cipher blocks. In the simplest case, known as electronic codebook (ECB) mode, a message is first split into separate blocks of the cipher's block size (possibly extending the last block with padding bits), and then each block is encrypted and decrypted independently. However, such a naive method is generally insecure because equal plaintext blocks will always generate equal ciphertext blocks (for the same key), so patterns in the plaintext message become evident in the ciphertext output.[23]

To overcome this limitation, several so-called block cipher modes of operation have been designed[24][25] and specified in national recommendations such as NIST 800-38A[26] and BSI TR-02102[27] and international standards such as ISO/IEC 10116.[28] The general concept is to use randomization of the plaintext data based on an additional input value, frequently called an initialization vector, to create what is termed probabilistic encryption.[29] In the popular cipher block chaining (CBC) mode, for encryption to be secure the initialization vector passed along with the plaintext message must be a random or pseudorandom value, which is added in an exclusive-or manner to the first plaintext block before it is encrypted. The resultant ciphertext block is then used as the new initialization vector for the next plaintext block. In the cipher feedback (CFB) mode, which emulates a self-synchronizing stream cipher, the initialization vector is first encrypted and then added to the plaintext block. The output feedback (OFB) mode repeatedly encrypts the initialization vector to create a key stream for the emulation of a synchronous stream cipher. The newer counter (CTR) mode similarly creates a key stream, but has the advantage of only needing unique and not (pseudo-)random values as initialization vectors; the needed randomness is derived internally by using the initialization vector as a block counter and encrypting this counter for each block.[26]


Insecure encryption of an image (depicting Tux) as a result of electronic codebook (ECB) mode encoding

From a security-theoretic point of view, modes of operation must provide what is known as semantic security.[30] Informally, it means that given some ciphertext under an unknown key one cannot practically derive any information from the ciphertext (other than the length of the message) over what one would have known without seeing the ciphertext. It has been shown that all of the modes discussed above, with the exception of the ECB mode, provide this property under so-called chosen plaintext attacks.

# Padding

Some modes such as the CBC mode only operate on complete plaintext blocks. Simply extending the last block of a message with zero bits is insufficient since it does not allow a receiver to easily distinguish messages that differ only in the number of padding bits. More importantly, such a simple solution gives rise to very efficient padding oracle attacks.[31] A suitable padding scheme is therefore needed to extend the last plaintext block to the cipher's block size. While many popular schemes described in standards and in the literature have been shown to be vulnerable to padding oracle attacks,[31][32] a solution that adds a one-bit and then extends the last block with zero-bits, standardized as "padding method 2" in ISO/IEC 9797-1,[33] has been proven secure against these attacks.[32]

# Cryptanalysis

Cryptanalysis is the technique in which ciphers are decrypted without knowledge of the used key. Different attacks can be employed based on the information available to the cryptanalyst, these Attack models are:

- *Ciphertext-only*: the cryptanalyst has access only to a collection of ciphertexts or codetexts.
- *Known-plaintext*: the attacker has a set of ciphertexts to which they know the corresponding plaintext.
- *Chosen-plaintext* (*chosen-ciphertext*): the attacker can obtain the ciphertexts (plaintexts) corresponding to an arbitrary set of plaintexts (ciphertexts) of their own choosing.
- *Adaptive chosen-plaintext*: like a chosen-plaintext attack, except the attacker can choose subsequent plaintexts based on information learned from previous encryptions, similarly to the *Adaptive chosen ciphertext attack*.

- *Related-key attack*: Like a chosen-plaintext attack, except the attacker can obtain ciphertexts encrypted under two different keys. The keys are unknown, but the relationship between them is known; for example, two keys that differ in the one bit.

## Brute-force attacks

This property results in the cipher's security degrading quadratically, and needs to be taken into account when selecting a block size. There is a trade-off though as large block sizes can result in the algorithm becoming inefficient to operate.[34] Earlier block ciphers such as the DES have typically selected a 64-bit block size, while newer designs such as the AES support block sizes of 128 bits or more, with some ciphers supporting a range of different block sizes.[35]

## Differential cryptanalysis

## Linear cryptanalysis

*A linear cryptanalysis* is a form of cryptanalysis based on finding affine approximations to the action of a cipher. Linear cryptanalysis is one of the two most widely used attacks on block ciphers; the other being differential cryptanalysis.[36]

The discovery is attributed to Mitsuru Matsui, who first applied the technique to the FEAL cipher (Matsui and Yamagishi, 1992).[37]

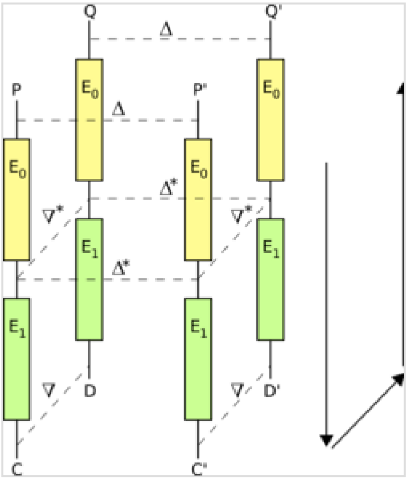
## Integral cryptanalysis

*Integral cryptanalysis* is a cryptanalytic attack that is particularly applicable to block ciphers based on substitution–permutation networks. Unlike differential cryptanalysis, which uses pairs of chosen plaintexts with a fixed XOR difference, integral cryptanalysis uses sets or even multisets of chosen plaintexts of which part is held constant and another part varies through all possibilities. For example, an attack might use 256 chosen plaintexts that have all but 8 of their bits the same, but all differ in those 8 bits. Such a set necessarily has an XOR sum of 0, and the XOR sums of the corresponding sets of ciphertexts provide information about the cipher's operation. This contrast between the differences between pairs of texts and the sums of larger sets of texts inspired the name "integral cryptanalysis", borrowing the terminology of calculus.

## Other techniques

In addition to linear and differential cryptanalysis, there is a growing catalog of attacks: truncated differential cryptanalysis, partial differential cryptanalysis, integral cryptanalysis, which encompasses square and integral attacks, slide attacks, boomerang attacks, the XSL attack, impossible differential cryptanalysis, and algebraic attacks. For a new block cipher design to have any credibility, it must demonstrate evidence of security against known attacks.[38]

# Provable security



The development of the boomerang attack enabled differential cryptanalysis techniques to be applied to many ciphers that had previously been deemed secure against differential attacks

When a block cipher is used in a given mode of operation, the resulting algorithm should ideally be about as secure as the block cipher itself. ECB (discussed above) emphatically lacks this property: regardless of how secure the underlying block cipher is, ECB mode can easily be attacked. On the other hand, CBC mode can be proven to be secure under the assumption that the underlying block cipher is likewise secure. Note, however, that making statements like this requires formal mathematical definitions for what it means for an encryption algorithm or a block cipher to "be secure". This section describes two common notions for what properties a block cipher should have. Each corresponds to a mathematical model that can be used to prove properties of higher-level algorithms, such as CBC.

This general approach to cryptography – proving higher-level algorithms (such as CBC) are secure under explicitly stated assumptions regarding their components (such as a block cipher) – is known as *provable security*.

## Standard model

Informally, a block cipher is secure in the standard model if an attacker cannot tell the difference between the block cipher (equipped with a random key) and a random permutation.

To be a bit more precise, let $E$ be an $n$-bit block cipher. We imagine the following game:

1. The person running the game flips a coin.
   - If the coin lands on heads, he chooses a random key $K$ and defines the function $f = E_K$.
   - If the coin lands on tails, he chooses a random permutation $\pi$ on the set of $n$-bit strings and defines the function $f = \pi$.
2. The attacker chooses an $n$-bit string $X$, and the person running the game tells him the value of $f(X)$.
3. Step 2 is repeated a total of $q$ times. (Each of these $q$ interactions is a *query*.)
4. The attacker guesses how the coin landed. He wins if his guess is correct.

The attacker, which we can model as an algorithm, is called an *adversary*. The function $f$ (which the adversary was able to query) is called an *oracle*.

Note that an adversary can trivially ensure a 50% chance of winning simply by guessing at random (or even by, for example, always guessing "heads"). Therefore, let $P_E(A)$ denote the probability that adversary $A$ wins this game against $E$, and define the *advantage* of $A$ as $2(P_E(A) - 1/2)$. It follows that if $A$ guesses randomly, its advantage will be 0; on the other hand, if $A$ always wins, then its advantage is 1. The block cipher $E$ is a *pseudo-random permutation* (PRP) if no adversary has an advantage significantly greater than 0, given specified restrictions on $q$ and the adversary's running time. If in Step 2 above adversaries have the option of learning $f^{-1}(X)$ instead of $f(X)$ (but still have only small advantages) then $E$

is a *strong* PRP (SPRP). An adversary is *non-adaptive* if it chooses all $q$ values for $X$ before the game begins (that is, it does not use any information gleaned from previous queries to choose each $X$ as it goes).

These definitions have proven useful for analyzing various modes of operation. For example, one can define a similar game for measuring the security of a block cipher-based encryption algorithm, and then try to show (through a reduction argument) that the probability of an adversary winning this new game is not much more than $P_E(A)$ for some $A$. (The reduction typically provides limits on $q$ and the running time of $A$.) Equivalently, if $P_E(A)$ is small for all relevant $A$, then no attacker has a significant probability of winning the new game. This formalizes the idea that the higher-level algorithm inherits the block cipher's security.

### Ideal cipher model

# Practical evaluation

Block ciphers may be evaluated according to multiple criteria in practice. Common factors include:[39][40]

- Key parameters, such as its key size and block size, both of which provide an upper bound on the security of the cipher.
- The *estimated security level*, which is based on the confidence gained in the block cipher design after it has largely withstood major efforts in cryptanalysis over time, the design's mathematical soundness, and the existence of practical or certificational[41] attacks.
- The cipher's *complexity* and its suitability for implementation in hardware or software. Hardware implementations may measure the complexity in terms of gate count or energy consumption, which are important parameters for resource-constrained devices.
- The cipher's *performance* in terms of processing throughput on various platforms, including its memory requirements.
- The *cost* of the cipher refers to licensing requirements that may apply due to intellectual property rights.
- The *flexibility* of the cipher includes its ability to support multiple key sizes and block lengths.

# Notable block ciphers

## Lucifer / DES

Lucifer is generally considered to be the first civilian block cipher, developed at IBM in the 1970s based on work done by Horst Feistel. A revised version of the algorithm was adopted as a U.S. government Federal Information Processing Standard: FIPS PUB 46 Data Encryption Standard (DES).[42] It was chosen by the U.S. National Bureau of Standards (NBS) after a public invitation for submissions and some internal changes by NBS (and, potentially, the NSA). DES was publicly released in 1976 and has been widely used.

DES was designed to, among other things, resist a certain cryptanalytic attack known to the NSA and rediscovered by IBM, though unknown publicly until rediscovered again and published by Eli Biham and Adi Shamir in the late 1980s. The technique is called differential cryptanalysis and remains one of the few general attacks against block ciphers; linear cryptanalysis is another but may have been unknown even to the NSA, prior to its publication by Mitsuru Matsui. DES prompted a large amount of other work and publications in cryptography and cryptanalysis in the open community and it inspired many new cipher designs.

DES has a block size of 64 bits and a key size of 56 bits. 64-bit blocks became common in block cipher designs after DES. Key length depended on several factors, including government regulation. Many observers in the 1970s commented that the 56-bit key length used for DES was too short. As time went on, its inadequacy became apparent, especially after a special-purpose machine designed to break DES was demonstrated in 1998 by the Electronic Frontier Foundation. An extension to DES, Triple DES, triple-encrypts each block with either two independent keys (112-bit key and 80-bit security) or three independent keys (168-bit key and 112-bit security). It was widely adopted as a replacement. As of 2011, the three-key version is still considered secure, though the National Institute of Standards and Technology (NIST) standards no longer permit the use of the two-key version in new applications, due to its 80-bit security level.[43]

## IDEA

The *International Data Encryption Algorithm* (*IDEA*) is a block cipher designed by James Massey of ETH Zurich and Xuejia Lai; it was first described in 1991, as an intended replacement for DES.

IDEA operates on 64-bit blocks using a 128-bit key and consists of a series of eight identical transformations (a *round*) and an output transformation (the *half-round*). The processes for encryption and decryption are similar. IDEA derives much of its security by interleaving operations from different groups – modular addition and multiplication, and bitwise *exclusive or (XOR)* – which are algebraically "incompatible" in some sense.

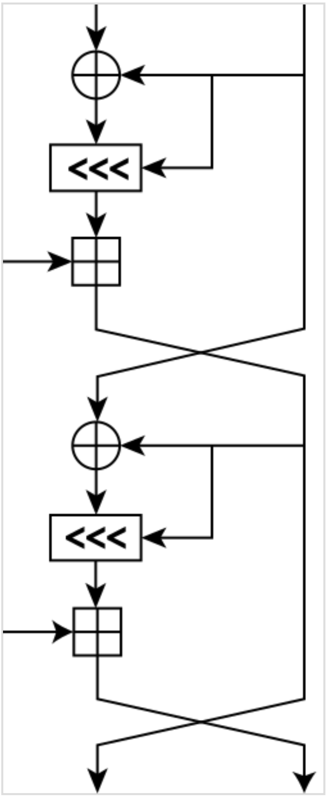
The designers analysed IDEA to measure its strength against differential cryptanalysis and concluded that it is immune under certain assumptions. No successful linear or algebraic weaknesses have been reported. As of 2012, the best attack which applies to all keys can break a full 8.5-round IDEA using a narrow-bicliques attack about four times faster than brute force.

## RC5

RC5 is a block cipher designed by Ronald Rivest in 1994 which, unlike many other ciphers, has a variable block size (32, 64, or 128 bits), key size (0 to 2040 bits), and a number of rounds (0 to 255). The original suggested choice of parameters was a block size of 64 bits, a 128-bit key, and 12 rounds.

A key feature of RC5 is the use of data-dependent rotations; one of the goals of RC5 was to prompt the study and evaluation of such operations as a cryptographic primitive. RC5 also consists of a number of modular additions and XORs. The general structure of the algorithm is a Feistel-like a network. The encryption and decryption routines can be specified in a few lines of code. The key schedule, however, is more complex, expanding the key using an essentially one-way function with the binary expansions of both e and the golden ratio as sources of "nothing up my sleeve numbers". The tantalizing simplicity of the algorithm together with the novelty of the data-dependent rotations has made RC5 an attractive object of study for cryptanalysts.

12-round RC5 (with 64-bit blocks) is susceptible to a differential attack using $2^{44}$ chosen plaintexts.[44] 18–20 rounds are suggested as sufficient protection.



One round (two half-rounds) of the RC5 block cipher

### Rijndael / AES

The *Rijndael* cipher developed by Belgian cryptographers, Joan Daemen and Vincent Rijmen was one of the competing designs to replace DES. It won the 5-year public competition to become the AES (Advanced Encryption Standard).

Adopted by NIST in 2001, AES has a fixed block size of 128 bits and a key size of 128, 192, or 256 bits, whereas Rijndael can be specified with block and key sizes in any multiple of 32 bits, with a minimum of 128 bits. The block size has a maximum of 256 bits, but the key size has no theoretical maximum. AES operates on a 4×4 column-major order matrix of bytes, termed the *state* (versions of Rijndael with a larger block size have additional columns in the state).

### Blowfish

*Blowfish* is a block cipher, designed in 1993 by Bruce Schneier and included in a large number of cipher suites and encryption products. Blowfish has a 64-bit block size and a variable key length from 1 bit up to 448 bits.[45] It is a 16-round Feistel cipher and uses large key-dependent S-boxes. Notable features of the design include the key-dependent S-boxes and a highly complex key schedule.

It was designed as a general-purpose algorithm, intended as an alternative to the aging DES and free of the problems and constraints associated with other algorithms. At the time Blowfish was released, many other designs were proprietary, encumbered by patents, or were commercial/government secrets. Schneier has stated that "Blowfish is unpatented, and will remain so in all countries. The algorithm is hereby placed in the public domain, and can be freely used by anyone." The same applies to Twofish, a successor algorithm from Schneier.

## Generalizations

### Tweakable block ciphers

M. Liskov, R. Rivest, and D. Wagner have described a generalized version of block ciphers called "tweakable" block ciphers.[46] A tweakable block cipher accepts a second input called the *tweak* along with its usual plaintext or ciphertext input. The tweak, along with the key, selects the permutation computed by the cipher. If changing tweaks is sufficiently lightweight (compared with a usually fairly expensive key setup operation), then some interesting new operation modes become possible. The disk encryption theory article describes some of these modes.

### Format-preserving encryption

Block ciphers traditionally work over a binary alphabet. That is, both the input and the output are binary strings, consisting of *n* zeroes and ones. In some situations, however, one may wish to have a block cipher that works over some other alphabet; for example, encrypting 16-digit credit card numbers in such a way that the ciphertext is also a 16-digit number might facilitate adding an encryption layer to legacy software. This is an example of *format-preserving encryption*. More generally, format-preserving encryption requires a keyed permutation on some finite language. This makes format-preserving encryption schemes a natural generalization of (tweakable) block ciphers. In contrast, traditional encryption schemes, such as CBC, are not permutations because the same plaintext can encrypt multiple different ciphertexts, even when using a fixed key.

## Relation to other cryptographic primitives

Block ciphers can be used to build other cryptographic primitives, such as those below. For these other primitives to be cryptographically secure, care has to be taken to build them the right way.

- Stream ciphers can be built using block ciphers. OFB mode and CTR mode are block modes that turn a block cipher into a stream cipher.
- Cryptographic hash functions can be built using block ciphers.[47][48] See the one-way compression function for descriptions of several such methods. The methods resemble the block cipher modes of operation usually used for encryption.
- Cryptographically secure pseudorandom number generators (CSPRNGs) can be built using block ciphers.[49][50]
- Secure pseudorandom permutations of arbitrarily sized finite sets can be constructed with block ciphers; see Format-Preserving Encryption.
- A publicly known unpredictable permutation combined with key whitening is enough to construct a block cipher -- such as the single-key Even–Mansour cipher, perhaps the simplest possible provably secure block cipher.[51]
- Message authentication codes (MACs) are often built from block ciphers. CBC-MAC, OMAC, and PMAC are such MACs.
- Authenticated encryption is also built from block ciphers. It means to both encrypt and MAC at the same time. That is to both provide confidentiality and authentication. CCM, EAX, GCM, and OCB are such authenticated encryption modes.

Just as block ciphers can be used to build hash functions, like SHA-1 and SHA-2 are based on block ciphers which are also used independently as SHACAL, hash functions can be used to build block ciphers. Examples of such block ciphers are BEAR and LION.

## See also

- Cipher security summary
- Topics in cryptography
- XOR cipher

## References

1. Cusick, Thomas W.; Stanica, Pantelimon (2009). *Cryptographic Boolean functions and applications* (https://books.google.com/books?id=OAkhkLSxxxMC&pg=PA158). Academic Press. pp. 158–159. ISBN 9780123748904.
2. Menezes, Alfred J.; van Oorschot, Paul C.; Vanstone, Scott A. (1996). "Chapter 7: Block Ciphers". *Handbook of Applied Cryptography* (https://web.archive.org/web/20210203194011/https://cacr.uwaterloo.ca/hac/). CRC Press. ISBN 0-8493-8523-7. Archived from the original (http://cacr.uwaterloo.ca/hac/) on 2021-02-03. Retrieved 2012-07-15.
3. Bellare, Mihir; Rogaway, Phillip (11 May 2005), *Introduction to Modern Cryptography* (http://www.cs.ucdavis.edu/~rogaway/classes/227/spring05/book/main.pdf) (Lecture notes), archived (https://ghostarchive.org/archive/20221009/http://www.cs.ucdavis.edu/~rogaway/classes/227/spring05/book/main.pdf) (PDF) from the original on 2022-10-09, chapter 3.
4. Chakraborty, D.; Rodriguez-Henriquez, F. (2008). "Block Cipher Modes of Operation from a Hardware Implementation Perspective" (https://books.google.com/books?id=nErZY4vYHIoC&pg=PA321). In Koç, Çetin K. (ed.). *Cryptographic Engineering*. Springer. p. 321. ISBN 9780387718163.
5. Menezes, van Oorschot & Vanstone 1996, section 7.2.
6. Shannon, Claude (1949). "Communication Theory of Secrecy Systems" (https://web.archive.org/web/20070605092733/http://netlab.cs.ucla.edu/wiki/files/shannon1949.pdf) (PDF). *Bell System Technical Journal*. **28** (4): 656–715. doi:10.1002/j.1538-7305.1949.tb00928.x (https://doi.org/10.1002%2Fj.1538-7305.1949.tb00928.x). Archived from the original (http://netlab.cs.ucla.edu/wiki/files/shannon1949.pdf) (PDF) on 2007-06-05. Retrieved 2012-04-09.
7. van Tilborg, Henk C. A.; Jajodia, Sushil, eds. (2011). *Encyclopedia of Cryptography and Security* (https://books.google.com/books?id=UuNKmgv70lMC&pg=PA455). Springer. ISBN 978-1-4419-5905-8., p. 455.
8. van Tilborg & Jajodia 2011, p. 1268.
9. Rupp, Martin (16 August 2019). "The Benefits of the Atalla Key Block" (https://web.archive.org/web/20201017215047/https://content.hsm.utimaco.com/blog/the-benefits-of-atalla-key-block). *Utimaco*. Archived from the original (https://content.hsm.utimaco.com/blog/the-benefits-of-atalla-key-block) on 17 October 2020. Retrieved 10 September 2019.
10. Hamscher, Walter (1998). "Electronic Business without Fear: The Tristrata Security Architecture" (https://web.archive.org/web/20050529185702/http://www.standardadvantage.com/docs/fear.pdf) (PDF). CiteSeerX 10.1.1.123.2371 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.123.2371). Archived from the original (http://www.standardadvantage.com/docs/fear.pdf) (PDF) on 29 May 2005.
11. Stiennon, Richard (17 June 2014). "Key Management a Fast Growing Space" (https://securitycurrent.com/key-management-a-fast-growing-space/). *SecurityCurrent*. IT-Harvest. Retrieved 21 August 2019.
12. Junod, Pascal & Canteaut, Anne (2011). *Advanced Linear Cryptanalysis of Block and Stream Ciphers* (https://books.google.com/books?id=pMnRhjStTZoC&pg=PA2). IOS Press. p. 2. ISBN 9781607508441.
13. Aumasson, Jean-Philippe (6 November 2017). *Serious Cryptography: A Practical Introduction to Modern Encryption* (https://books.google.com/books?id=W1v6DwAAQBAJ&pg=PA56). No Starch Press. p. 56. ISBN 978-1-59327-826-7. OCLC 1012843116 (https://search.worldcat.org/oclc/1012843116).
14. Keliher, Liam; et al. (2000). "Modeling Linear Characteristics of Substitution–Permutation Networks". In Hays, Howard; Carlisle, Adam (eds.). *Selected areas in cryptography: 6th annual international workshop, SAC'99, Kingston, Ontario, Canada, August 9–10, 1999 : proceedings* (https://archive.org/details/springer_10.1007-3-540-46513-8). Springer. p. 79 (https://archive.org/details/springer_10.1007-3-540-46513-8/page/n87). ISBN 9783540671855.

15. Baigneres, Thomas; Finiasz, Matthieu (2007). "Dial 'C' for Cipher" (https://books.google.com/books?id=yb99g5G7FS4C&pg=PA77). In Biham, Eli; Yousseff, Amr (eds.). *Selected areas in cryptography: 13th international workshop, SAC 2006, Montreal, Canada, August 17–18, 2006 : revised selected papers*. Springer. p. 77. ISBN 9783540744610.
16. Cusick, Thomas W.; Stanica, Pantelimon (2009). *Cryptographic Boolean functions and applications* (https://books.google.com/books?id=OAkhkLSxxxMC&pg=PA164). Academic Press. p. 164. ISBN 9780123748904.
17. Katz, Jonathan; Lindell, Yehuda (2008). *Introduction to modern cryptography* (https://archive.org/details/Introduction_to_Modern_Cryptography). CRC Press. p. 166 (https://archive.org/details/Introduction_to_Modern_Cryptography/page/n184). ISBN 9781584885511., pages 166–167.
18. Nayaka, Raja Jitendra; Biradar, R. C. (2013). "Key based S-box selection and key expansion algorithm for substitution-permutation network cryptography". *2013 Annual International Conference on Emerging Research Areas and 2013 International Conference on Microelectronics, Communications and Renewable Energy*. pp. 1–6. doi:10.1109/AICERA-ICMiCR.2013.6575944 (https://doi.org/10.1109%2FAICERA-ICMiCR.2013.6575944). ISBN 978-1-4673-5149-2.
19. Subhabrata Samajder (2017). *Block Cipher Cryptanalysis: An Overview*. Kolkata: Indian Statistical Institute. pp. 5/52.
20. Katz & Lindell 2008, pp. 170–172.
21. Katz & Lindell 2008, p. 171.
22. Aumasson, Jean-Philippe; Bernstein, Daniel J. (2012). "SipHash: a fast short-input PRF" (https://web.archive.org/web/20200312053222/https://131002.net/siphash/siphash.pdf) (PDF). In Galbraith, Steven; Nandi, Mridul (eds.). *Progress in cryptology-- INDOCRYPT 2012 : 13th International Conference on Cryptology in India, Kolkata, India, December 9-12, 2012, proceedings*. Berlin: Springer. p. 494. doi:10.1007/978-3-642-34931-7_28 (https://doi.org/10.1007%2F978-3-642-34931-7_28). ISBN 978-3-642-34931-7. Archived from the original (https://131002.net/siphash/siphash.pdf) (PDF) on 2020-03-12.
23. Menezes, van Oorschot & Vanstone 1996, pp. 228–230, Chapter 7.
24. "Block Cipher Modes" (http://csrc.nist.gov/groups/ST/toolkit/BCM/index.html). NIST Computer Security Resource Center. 4 January 2017.
25. Menezes, van Oorschot & Vanstone 1996, pp. 228–233.
26. Morris Dworkin (December 2001), "Recommendation for Block Cipher Modes of Operation – Methods and Techniques" (http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf) (PDF), *Special Publication 800-38A*, National Institute of Standards and Technology (NIST), doi:10.6028/NIST.SP.800-38A (https://doi.org/10.6028%2FNIST.SP.800-38A), archived (https://ghostarchive.org/archive/20221009/http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf) (PDF) from the original on 2022-10-09
27. "Kryptographische Verfahren: Empfehlungen und Schlüssellängen", *Bsi Tr-02102* (Technische Richtlinie) (Version 1.0), June 20, 2008
28. "ISO/IEC 10116:2006 *Information technology — Security techniques — Modes of operation for an n-bit block cipher*" (http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=38761).
29. Bellare & Rogaway 2005, p. 101, section 5.3.
30. Bellare & Rogaway 2005, section 5.6.
31. Serge Vaudenay (2002). "Security Flaws Induced by CBC Padding — Applications to SSL, IPSEC, WTLS". *Advances in Cryptology — EUROCRYPT 2002* (http://infoscience.epfl.ch/record/99410). Lecture Notes in Computer Science. Vol. 2332. Springer Verlag. pp. 534–545. doi:10.1007/3-540-46035-7_35 (https://doi.org/10.1007%2F3-540-46035-7_35). ISBN 978-3-540-43553-2.

32. Kenneth G. Paterson; Gaven J. Watson (2008). "Immunising CBC Mode Against Padding Oracle Attacks: A Formal Security Treatment". *Security and Cryptography for Networks*. Lecture Notes in Computer Science. Vol. 5229. Springer Verlag. pp. 340–357. doi:10.1007/978-3-540-85855-3_23 (https://doi.org/10.1007%2F978-3-540-85855-3_23). ISBN 978-3-540-85854-6.
33. *ISO/IEC 9797-1: Information technology – Security techniques – Message Authentication Codes (MACs) – Part 1: Mechanisms using a block cipher* (http://www.iso.org/iso/iso_catalogue/catalogue_ics/catalogue_detail_ics.htm?csnumber=50375), ISO/IEC, 2011
34. Martin, Keith M. (2012). *Everyday Cryptography: Fundamental Principles and Applications* (https://books.google.com/books?id=5DZ_vv-gl4oC&pg=PA114). Oxford University Press. p. 114. ISBN 9780199695591.
35. Paar, Christof; et al. (2010). *Understanding Cryptography: A Textbook for Students and Practitioners* (https://books.google.com/books?id=f24wFELSzkoC&pg=PA30). Springer. p. 30. ISBN 9783642041006.
36. Matsui, Mitsuru. "Linear Cryptanalysis of DES Cipher" (http://ftp.esat.kuleuven.ac.be/pub/cosic/rijmen/misc/lin_cr.ps). *Mitsubishi Electric Corporation*. **1** (3): 43 – via Computer & Information Systems Laboratory.
37. Matsui, M. & Yamagishi, A. "A new method for known plaintext attack of FEAL cipher". *Advances in Cryptology – EUROCRYPT 1992*.
38. Wu, Shengbao; Wang, Mingsheng (2011), *Security Evaluation against Differential Cryptanalysis for Block Cipher Structures* (https://eprint.iacr.org/2011/551), retrieved 2025-01-01
39. Menezes, van Oorschot & Vanstone 1996, p. 227.
40. James Nechvatal; Elaine Barker; Lawrence Bassham; William Burr; Morris Dworkin; James Foti; Edward Roback (October 2000), *Report on the Development of the Advanced Encryption Standard (AES)* (http://csrc.nist.gov/archive/aes/round2/r2report.pdf) (PDF), National Institute of Standards and Technology (NIST), archived (https://ghostarchive.org/archive/20221009/http://csrc.nist.gov/archive/aes/round2/r2report.pdf) (PDF) from the original on 2022-10-09
41. Attacks that show that the cipher does not perform as advertised (i.e., the level of difficulty involved in breaking it is lower than claimed), which are nevertheless of high enough complexity so that they are not practically achievable.
42. FIPS PUB 46-3 *Data Encryption Standard (DES)* (http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf) (This is the third edition, 1999, but includes historical information in the preliminary section 12.)
43. NIST Special Publication 800-57 *Recommendation for Key Management — Part 1: General (Revised)*, March, 2007 (http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-revised2_Mar08-2007.pdf) Archived (https://web.archive.org/web/20140606050814/http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-revised2_Mar08-2007.pdf) June 6, 2014, at the Wayback Machine.
44. Biryukov A. and Kushilevitz E. (1998). Improved Cryptanalysis of RC5. EUROCRYPT 1998.
45. Bruce Schneier (1994). "Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish)" (http://www.schneier.com/paper-blowfish-fse.html). *Dr. Dobb's Journal*. **19** (4): 38–40.
46. Liskov, M.; Rivest, R.; Wagner, D. "Tweakable Block Ciphers" (http://www.cs.colorado.edu/~jrblack/class/csci7000/f03/papers/tweak-crypto02.pdf) (PDF). *Crypto 2002*. Archived (https://ghostarchive.org/archive/20221009/http://www.cs.colorado.edu/~jrblack/class/csci7000/f03/papers/tweak-crypto02.pdf) (PDF) from the original on 2022-10-09.
47. "ISO/IEC 10118-2:2010 *Information technology — Security techniques — Hash-functions — Part 2: Hash-functions using an n-bit block cipher*" (http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=44737).
48. Menezes, van Oorschot & Vanstone 1996, Chapter 9: Hash Functions and Data Integrity.

49. Barker, E. B.; Kelsey, J. M. (2012). "NIST Special Publication 800-90A *Recommendation for Random Number Generation Using Deterministic Random Bit Generators*" (http://csrc.nist.gov/publications/nistpubs/800-90A/SP800-90A.pdf) (PDF). doi:10.6028/NIST.SP.800-90A (https://doi.org/10.6028%2FNIST.SP.800-90A). `{{cite journal}}`: Cite journal requires `|journal=` (help)
50. Menezes, van Oorschot & Vanstone 1996, Chapter 5: Pseudorandom Bits and Sequences.
51. Orr Dunkelman, Nathan Keller, and Adi Shamir. "Minimalism in Cryptography: The Even–Mansour Scheme Revisited" (http://eprint.iacr.org/2011/541.pdf).

## Further reading

- Knudsen, Lars R.; Robshaw, Matthew (2011). *The Block Cipher Companion* (https://books.google.com/books?id=YiZKt_FcmYQC). Springer. ISBN 9783642173417.

## External links

- A list of many symmetric algorithms, the majority of which are block ciphers. (http://www.users.zetnet.co.uk/hopwood/crypto/scan/cs.html)
- The block cipher lounge (http://www.mat.dtu.dk/people/Lars.R.Knudsen/bc.html)
- What is a block cipher? (https://web.archive.org/web/20070306055504/http://www.rsa.com/rsalabs/node.asp?id=2168) from RSA FAQ
- Block Cipher based on Gold Sequences and Chaotic Logistic Tent System (https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8746274)