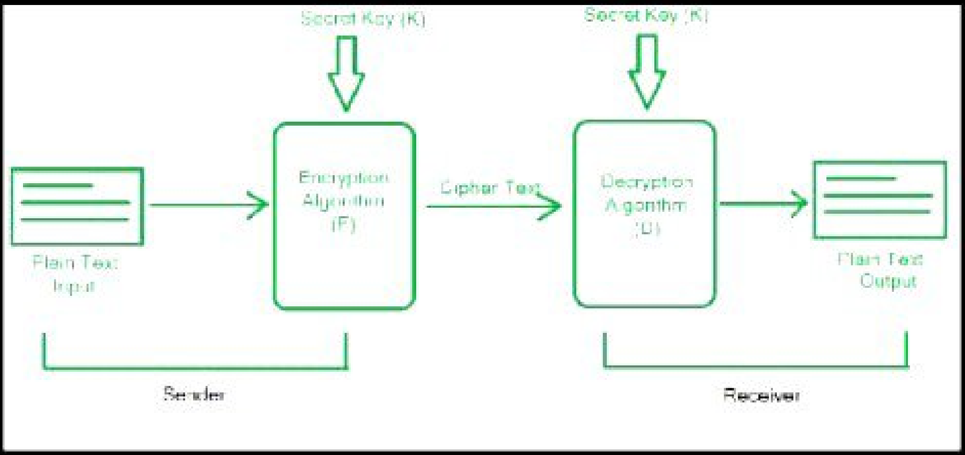
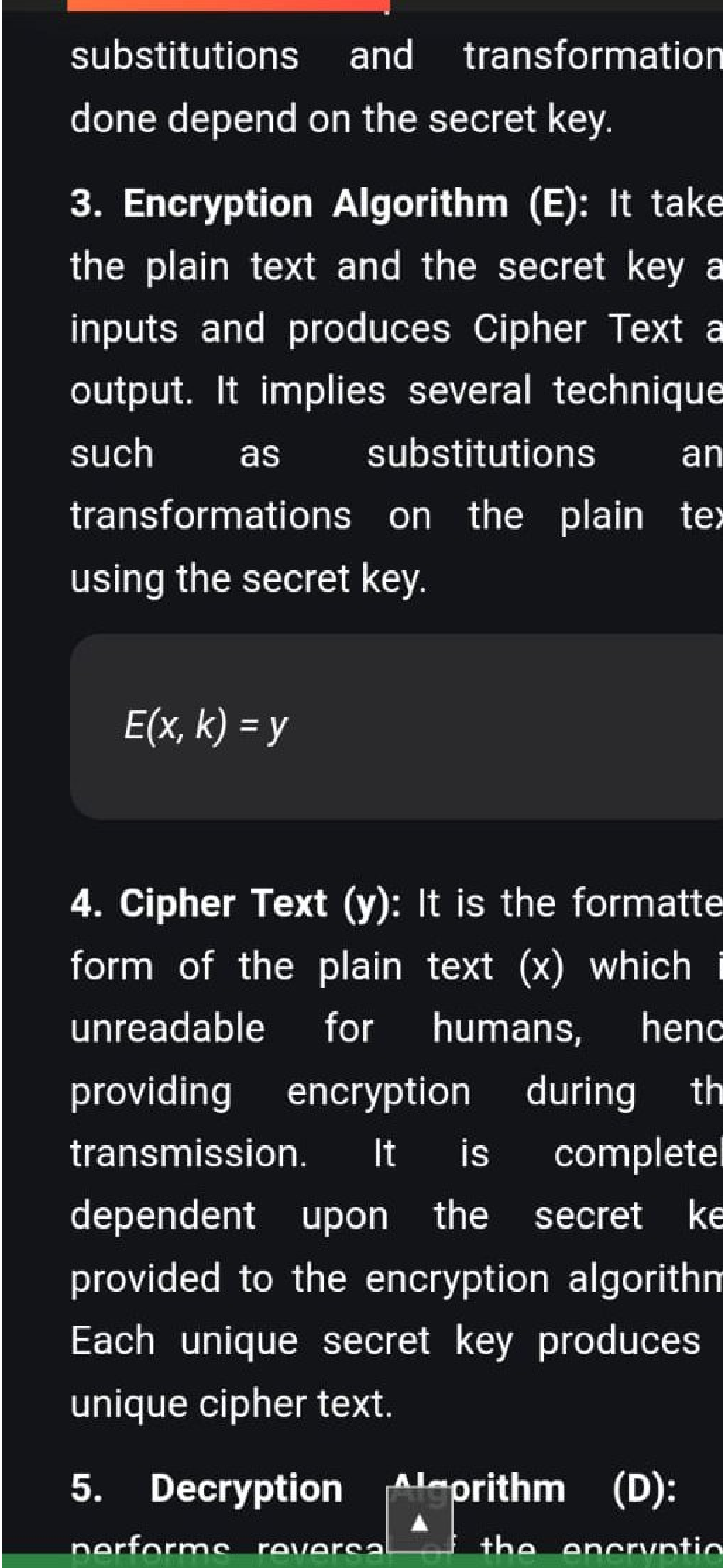
The Symmetric Cipher Model:

A symmetric cipher model is composed of five essential parts:



1. Plain Text (x): This is the original data/message that is to be communicated to the receiver by the sender. It is one of the inputs to the encryption algorithm.

2. Secret Key (k): It is a value/string/textfile used by the encryption and decryption algorithm to encode and decode the plain text to cipher text and vice-versa respectively. It is independent of the encryption algorithm. It governs all the conversions in plain text. All the substitutions and transformations



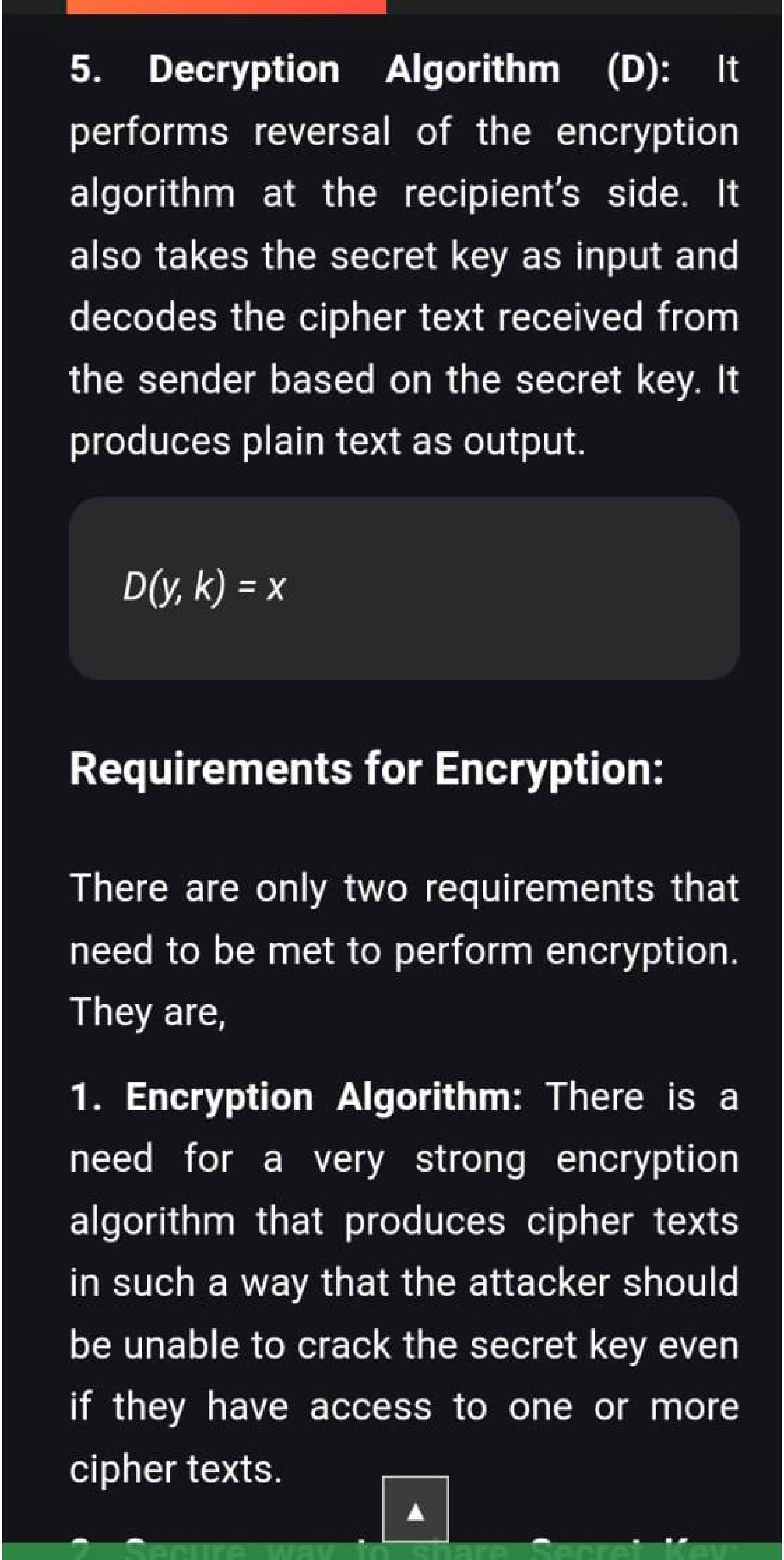
substitutions and transformation done depend on the secret key.

3. Encryption Algorithm (E): It take the plain text and the secret key a inputs and produces Cipher Text a output. It implies several technique such as substitutions an transformations on the plain tex using the secret key.

$$E(x, k) = y$$

4. Cipher Text (y): It is the formatte form of the plain text (x) which i unreadable for humans, henc providing encryption during th transmission. It is complete dependent upon the secret ke provided to the encryption algorithm Each unique secret key produces unique cipher text.

5. Decryption Algorithm (D): performs reversal of the encryptio



5. Decryption Algorithm (D): It performs reversal of the encryption algorithm at the recipient's side. It also takes the secret key as input and decodes the cipher text received from the sender based on the secret key. It produces plain text as output.

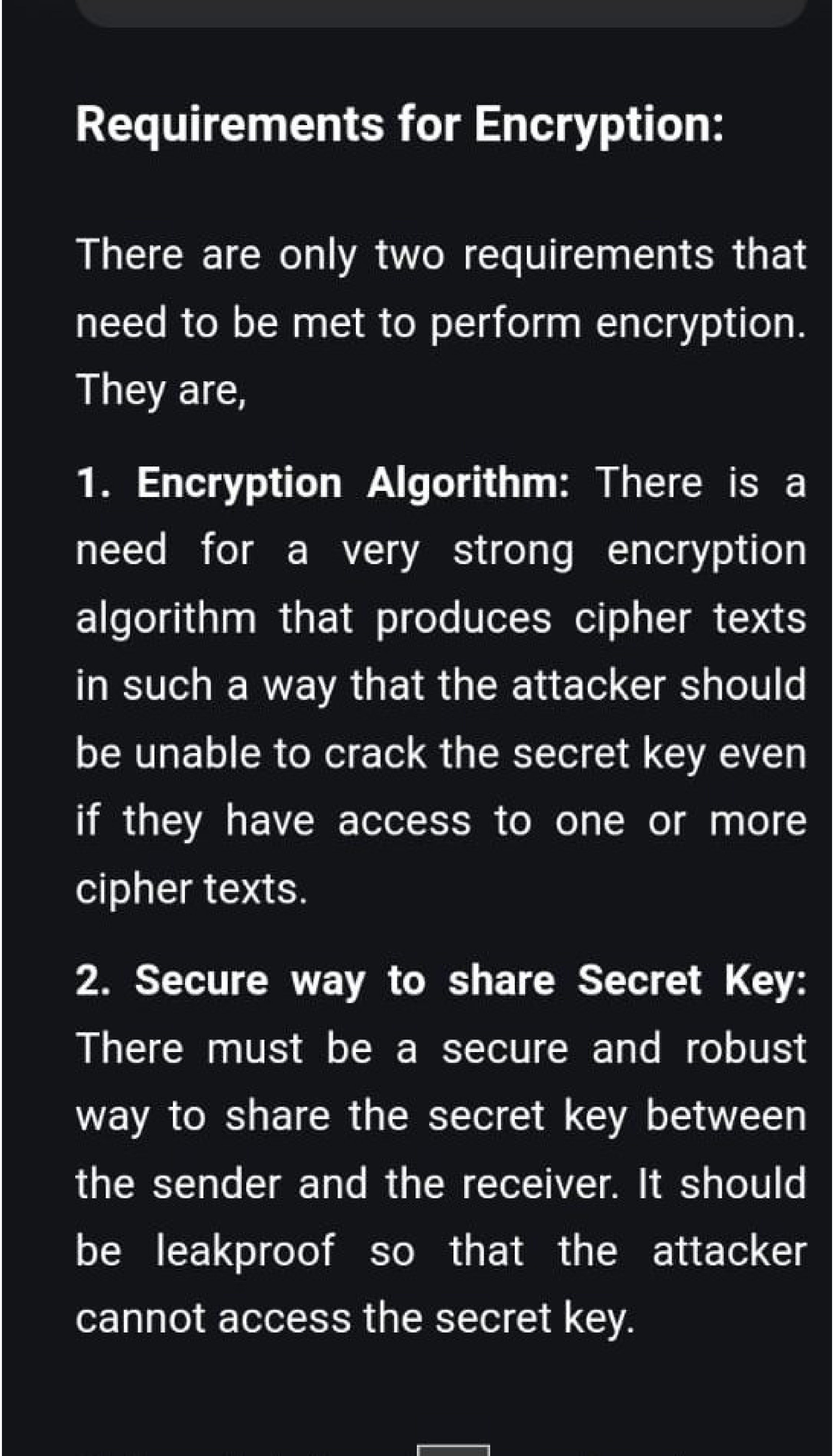
$$D(y, k) = x$$

Requirements for Encryption:

There are only two requirements that need to be met to perform encryption. They are,

1. Encryption Algorithm: There is a need for a very strong encryption algorithm that produces cipher texts in such a way that the attacker should be unable to crack the secret key even if they have access to one or more cipher texts.

2. Secure way to share Secret Key:

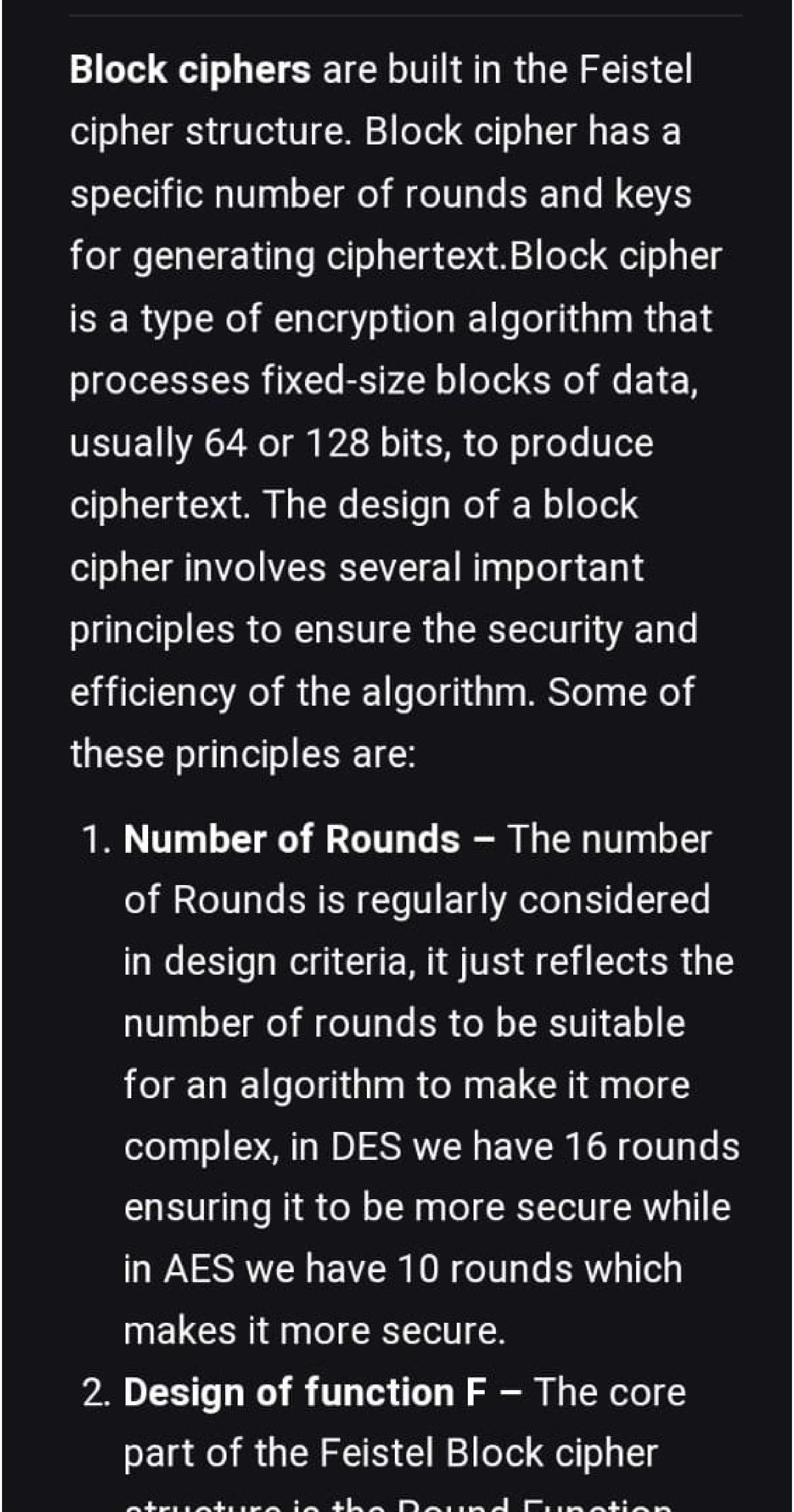


Requirements for Encryption:

There are only two requirements that need to be met to perform encryption. They are,

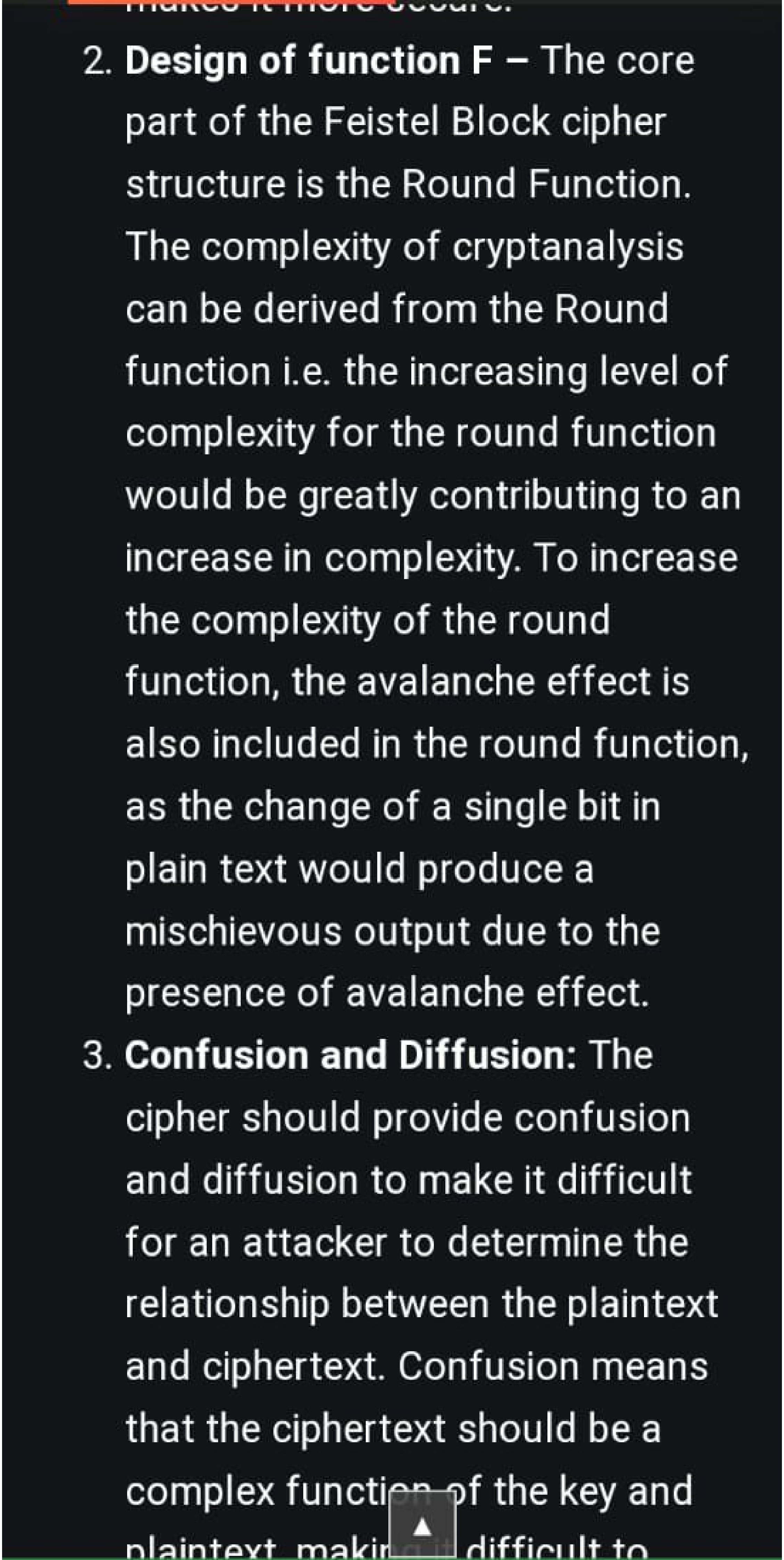
1. **Encryption Algorithm:** There is a need for a very strong encryption algorithm that produces cipher texts in such a way that the attacker should be unable to crack the secret key even if they have access to one or more cipher texts.

2. **Secure way to share Secret Key:** There must be a secure and robust way to share the secret key between the sender and the receiver. It should be leakproof so that the attacker cannot access the secret key.



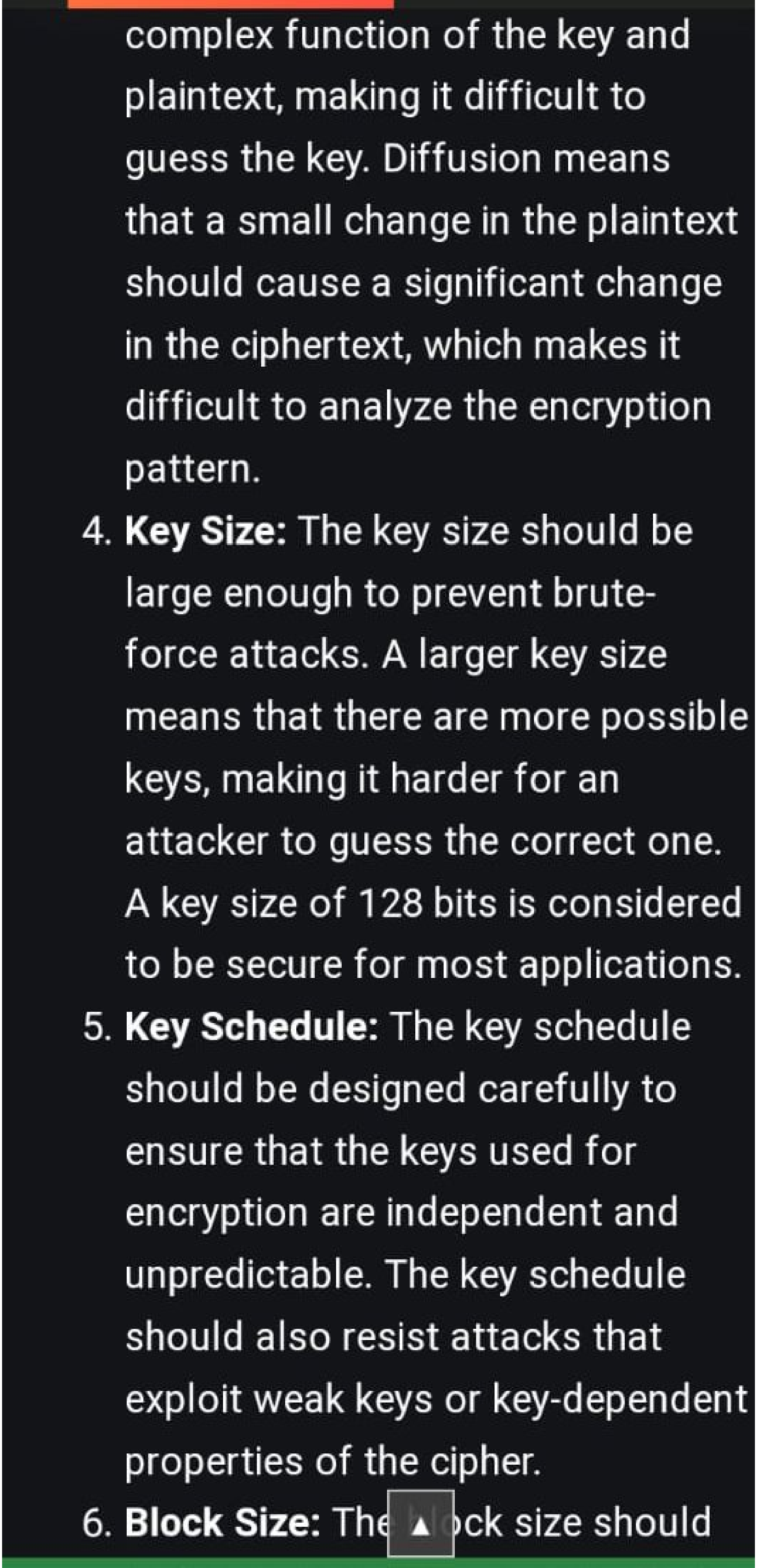
Block ciphers are built in the Feistel cipher structure. Block cipher has a specific number of rounds and keys for generating ciphertext.Block cipher is a type of encryption algorithm that processes fixed-size blocks of data, usually 64 or 128 bits, to produce ciphertext. The design of a block cipher involves several important principles to ensure the security and efficiency of the algorithm. Some of these principles are:

1. **Number of Rounds –** The number of Rounds is regularly considered in design criteria, it just reflects the number of rounds to be suitable for an algorithm to make it more complex, in DES we have 16 rounds ensuring it to be more secure while in AES we have 10 rounds which makes it more secure.
2. **Design of function F –** The core part of the Feistel Block cipher structure is the Round Function



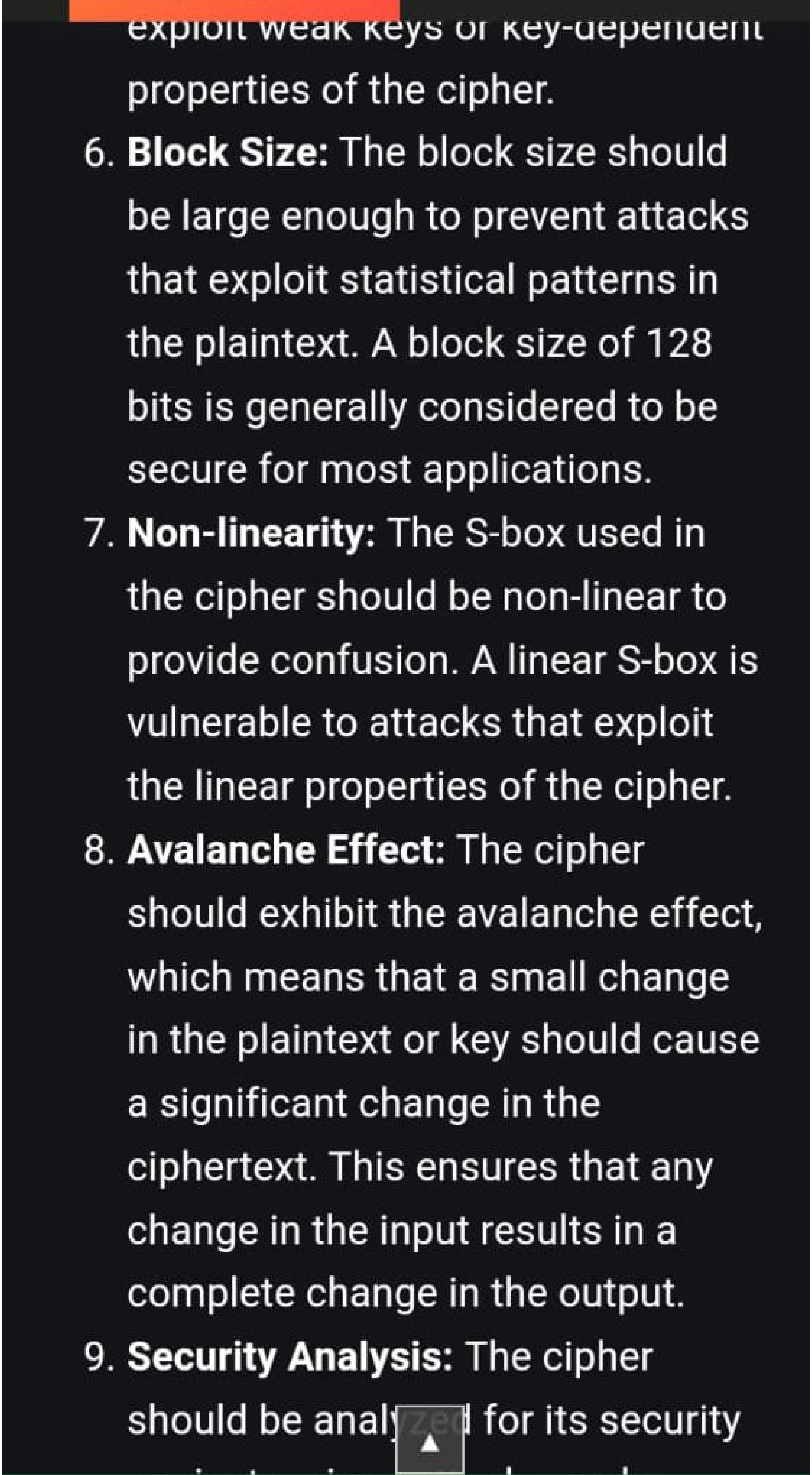
makes it more secure.

2. **Design of function F –** The core part of the Feistel Block cipher structure is the Round Function. The complexity of cryptanalysis can be derived from the Round function i.e. the increasing level of complexity for the round function would be greatly contributing to an increase in complexity. To increase the complexity of the round function, the avalanche effect is also included in the round function, as the change of a single bit in plain text would produce a mischievous output due to the presence of avalanche effect.
3. **Confusion and Diffusion:** The cipher should provide confusion and diffusion to make it difficult for an attacker to determine the relationship between the plaintext and ciphertext. Confusion means that the ciphertext should be a complex function of the key and plaintext, making it difficult to



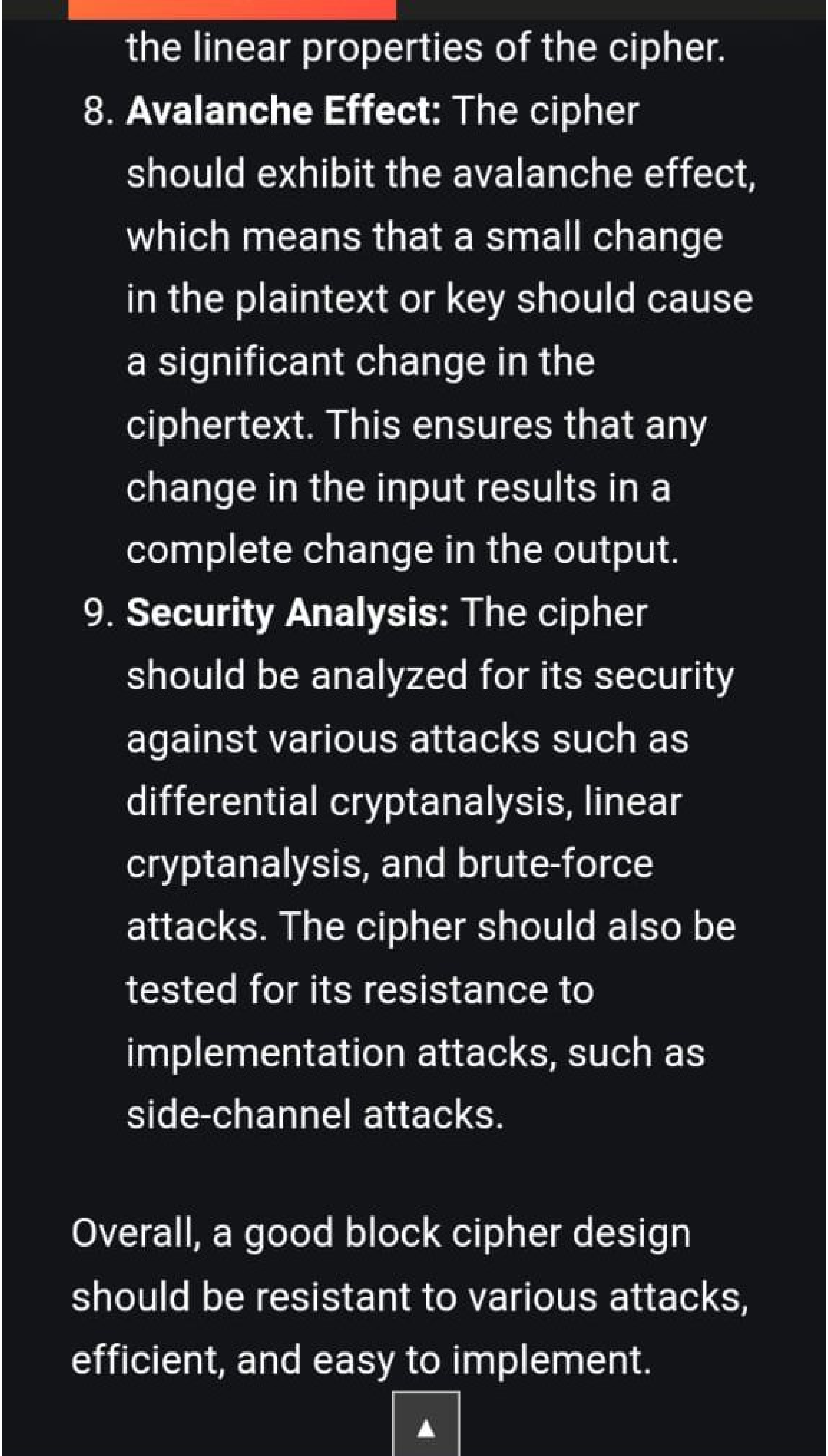
complex function of the key and plaintext, making it difficult to guess the key. Diffusion means that a small change in the plaintext should cause a significant change in the ciphertext, which makes it difficult to analyze the encryption pattern.

4. **Key Size:** The key size should be large enough to prevent brute-force attacks. A larger key size means that there are more possible keys, making it harder for an attacker to guess the correct one. A key size of 128 bits is considered to be secure for most applications.
5. **Key Schedule:** The key schedule should be designed carefully to ensure that the keys used for encryption are independent and unpredictable. The key schedule should also resist attacks that exploit weak keys or key-dependent properties of the cipher.
6. **Block Size:** The block size should



exploit weak keys or key-dependent properties of the cipher.

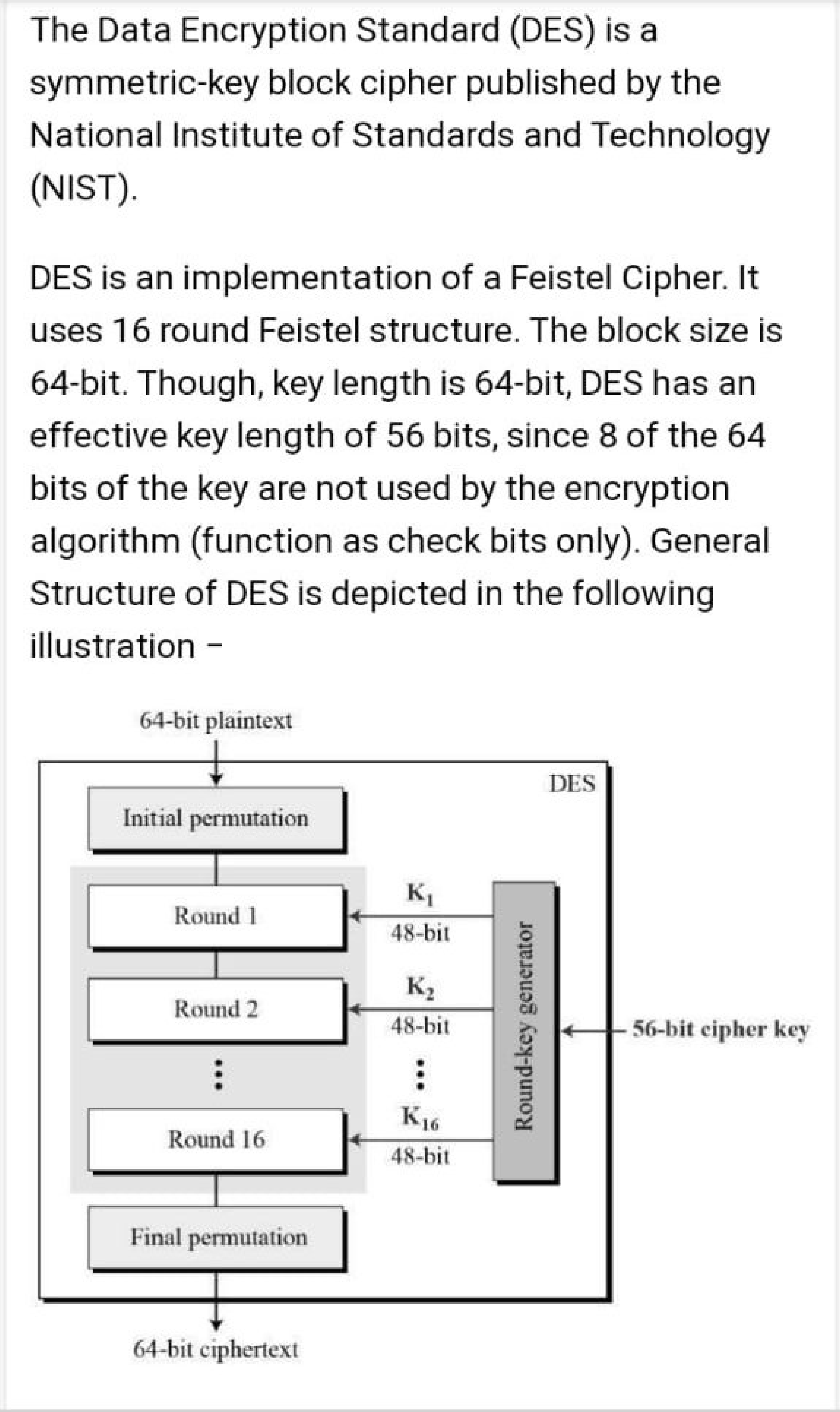
6. **Block Size:** The block size should be large enough to prevent attacks that exploit statistical patterns in the plaintext. A block size of 128 bits is generally considered to be secure for most applications.
7. **Non-linearity:** The S-box used in the cipher should be non-linear to provide confusion. A linear S-box is vulnerable to attacks that exploit the linear properties of the cipher.
8. **Avalanche Effect:** The cipher should exhibit the avalanche effect, which means that a small change in the plaintext or key should cause a significant change in the ciphertext. This ensures that any change in the input results in a complete change in the output.
9. **Security Analysis:** The cipher should be analyzed for its security



the linear properties of the cipher.

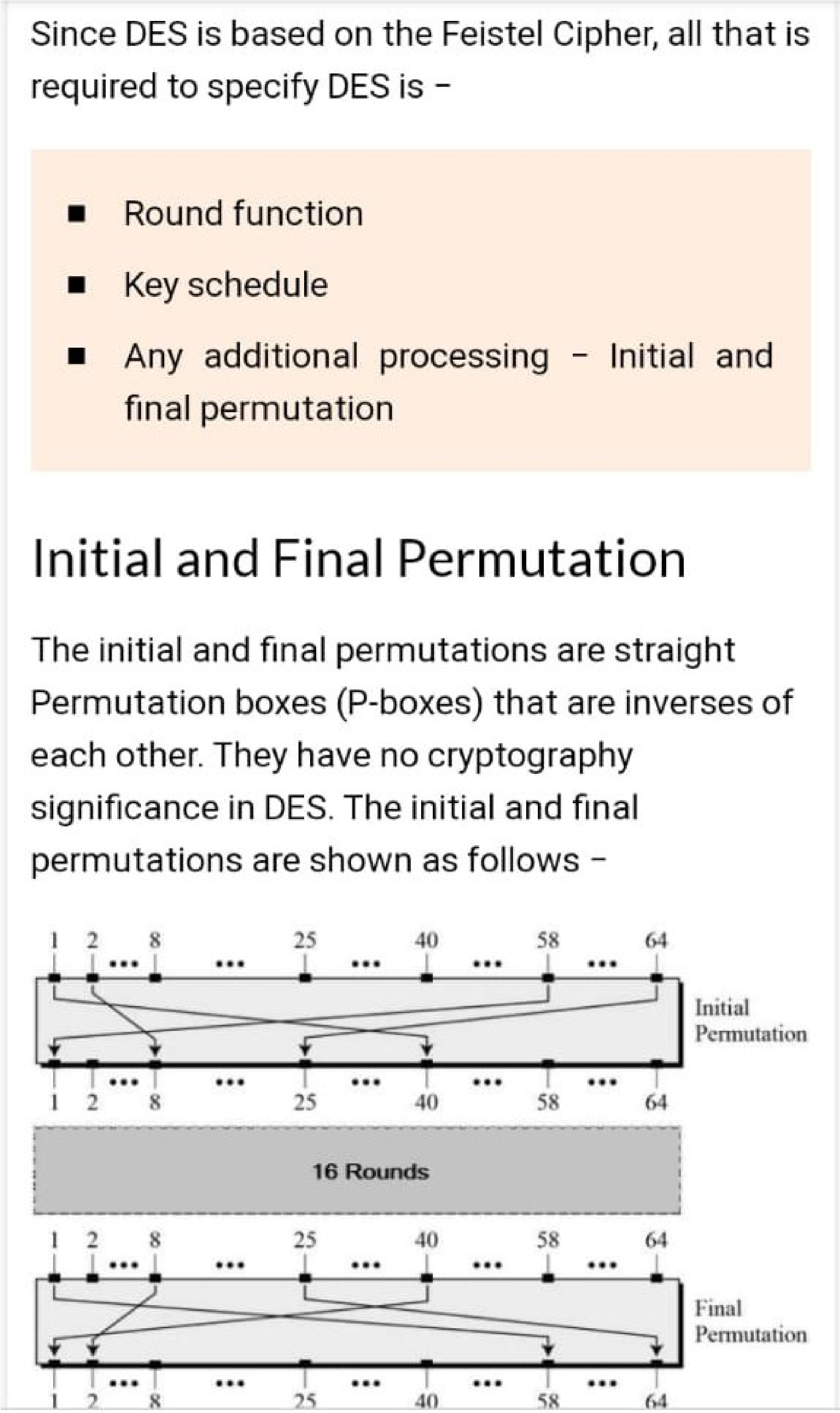
8. **Avalanche Effect:** The cipher should exhibit the avalanche effect, which means that a small change in the plaintext or key should cause a significant change in the ciphertext. This ensures that any change in the input results in a complete change in the output.
9. **Security Analysis:** The cipher should be analyzed for its security against various attacks such as differential cryptanalysis, linear cryptanalysis, and brute-force attacks. The cipher should also be tested for its resistance to implementation attacks, such as side-channel attacks.

Overall, a good block cipher design should be resistant to various attacks, efficient, and easy to implement.



The Data Encryption Standard (DES) is a symmetric-key block cipher published by the National Institute of Standards and Technology (NIST).

DES is an implementation of a Feistel Cipher. It uses 16 round Feistel structure. The block size is 64-bit. Though, key length is 64-bit, DES has an effective key length of 56 bits, since 8 of the 64 bits of the key are not used by the encryption algorithm (function as check bits only). General Structure of DES is depicted in the following illustration –



Since DES is based on the Feistel Cipher, all that is required to specify DES is –

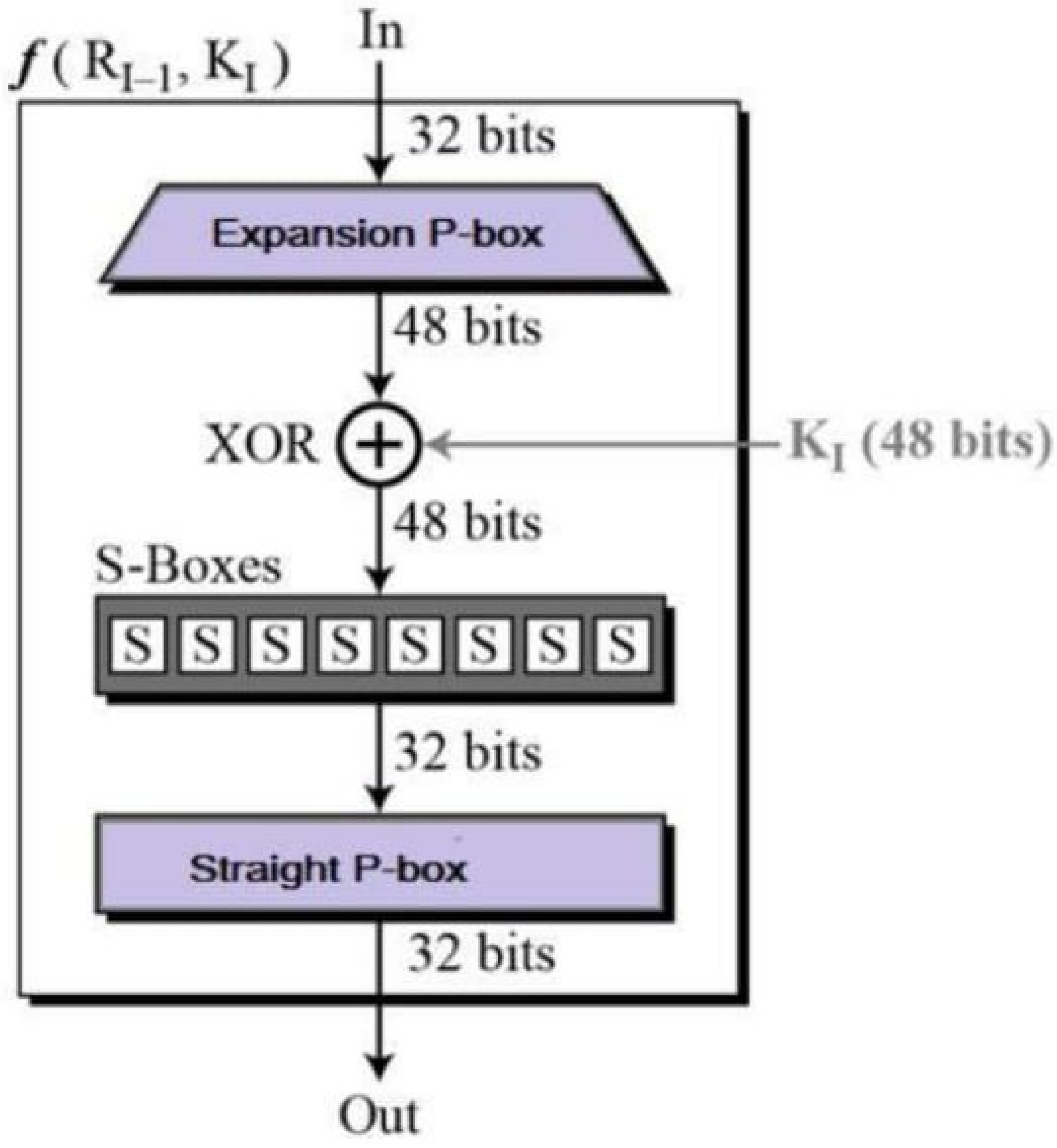
- Round function
- Key schedule
- Any additional processing – Initial and final permutation

Initial and Final Permutation

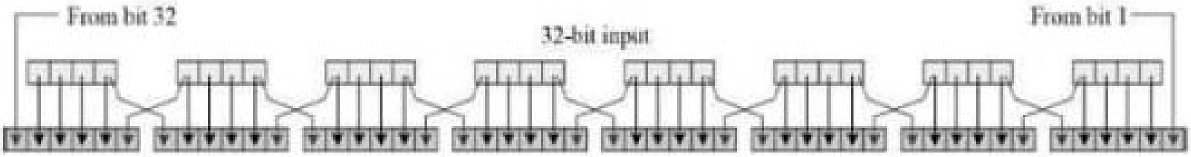
The initial and final permutations are straight Permutation boxes (P-boxes) that are inverses of each other. They have no cryptography significance in DES. The initial and final permutations are shown as follows –

Round Function

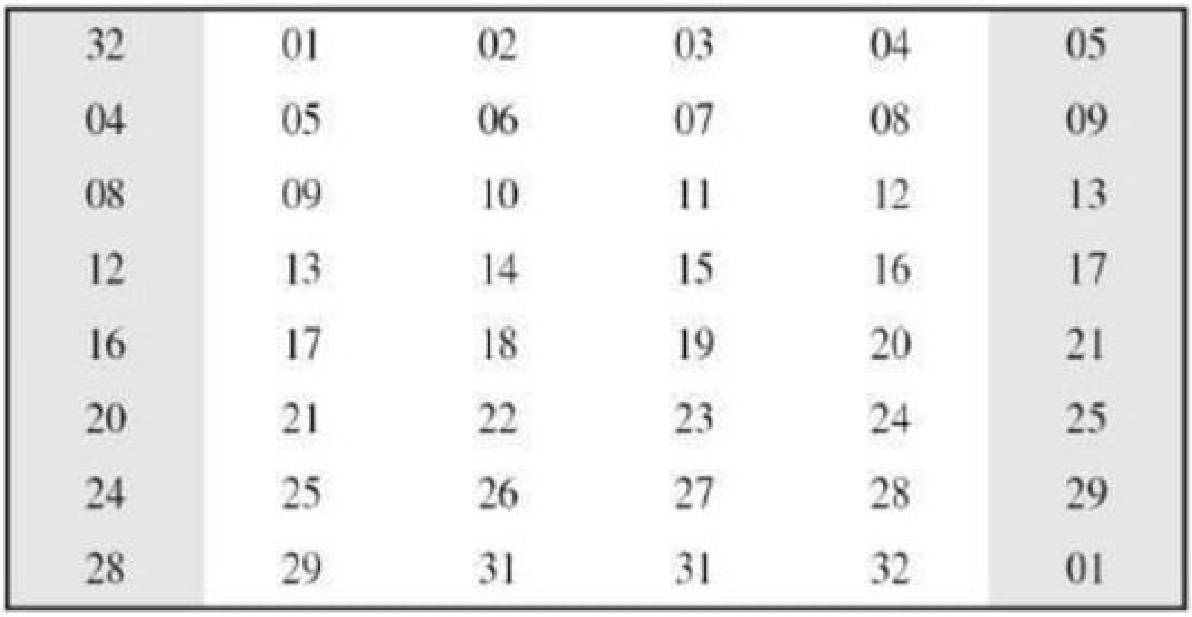
The heart of this cipher is the DES function, f. The DES function applies a 48-bit key to the rightmost 32 bits to produce a 32-bit output.



- **Expansion Permutation Box** – Since right input is 32-bit and round key is a 48-bit, we first need to expand right input to 48 bits. Permutation logic is graphically bits. Permutation logic is graphically depicted in the following illustration –



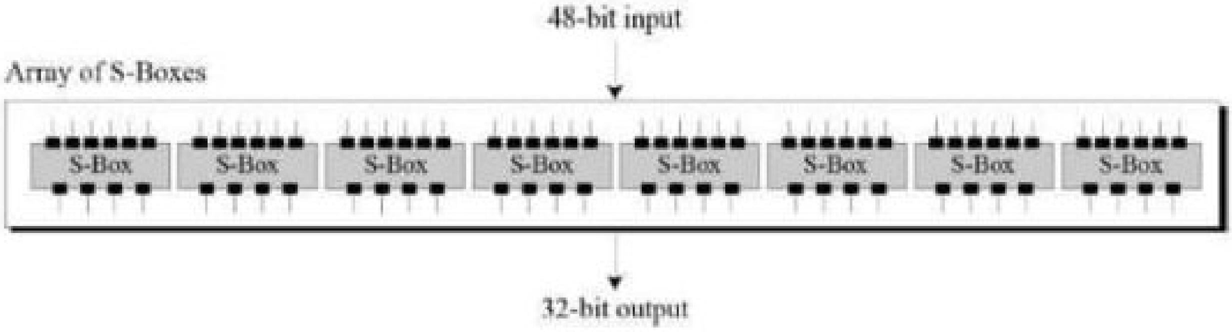
- The graphically depicted permutation logic is generally described as table in DES specification illustrated as shown –



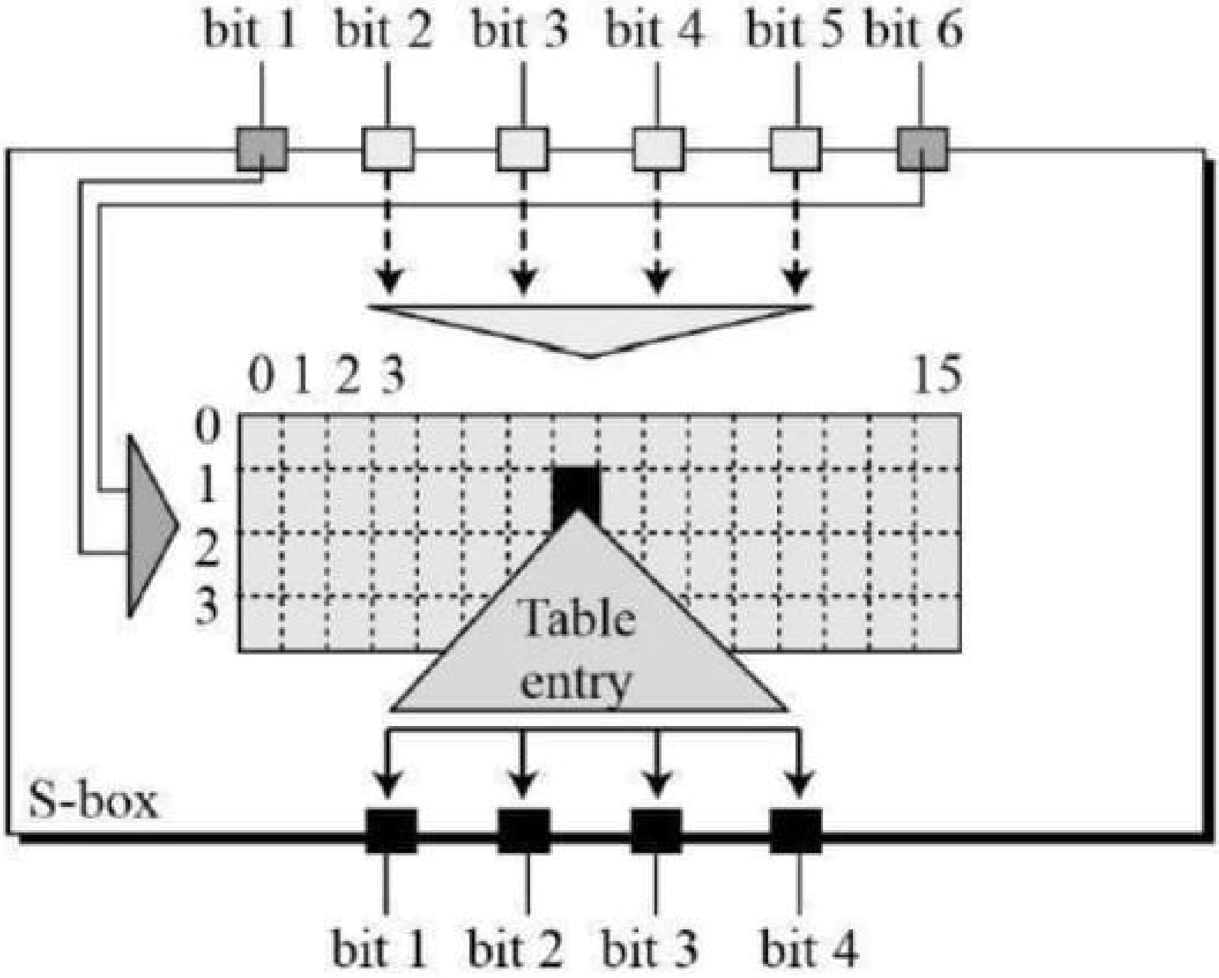
32	01	02	03	04	05
04	05	06	07	08	09
08	09	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	31	31	32	01

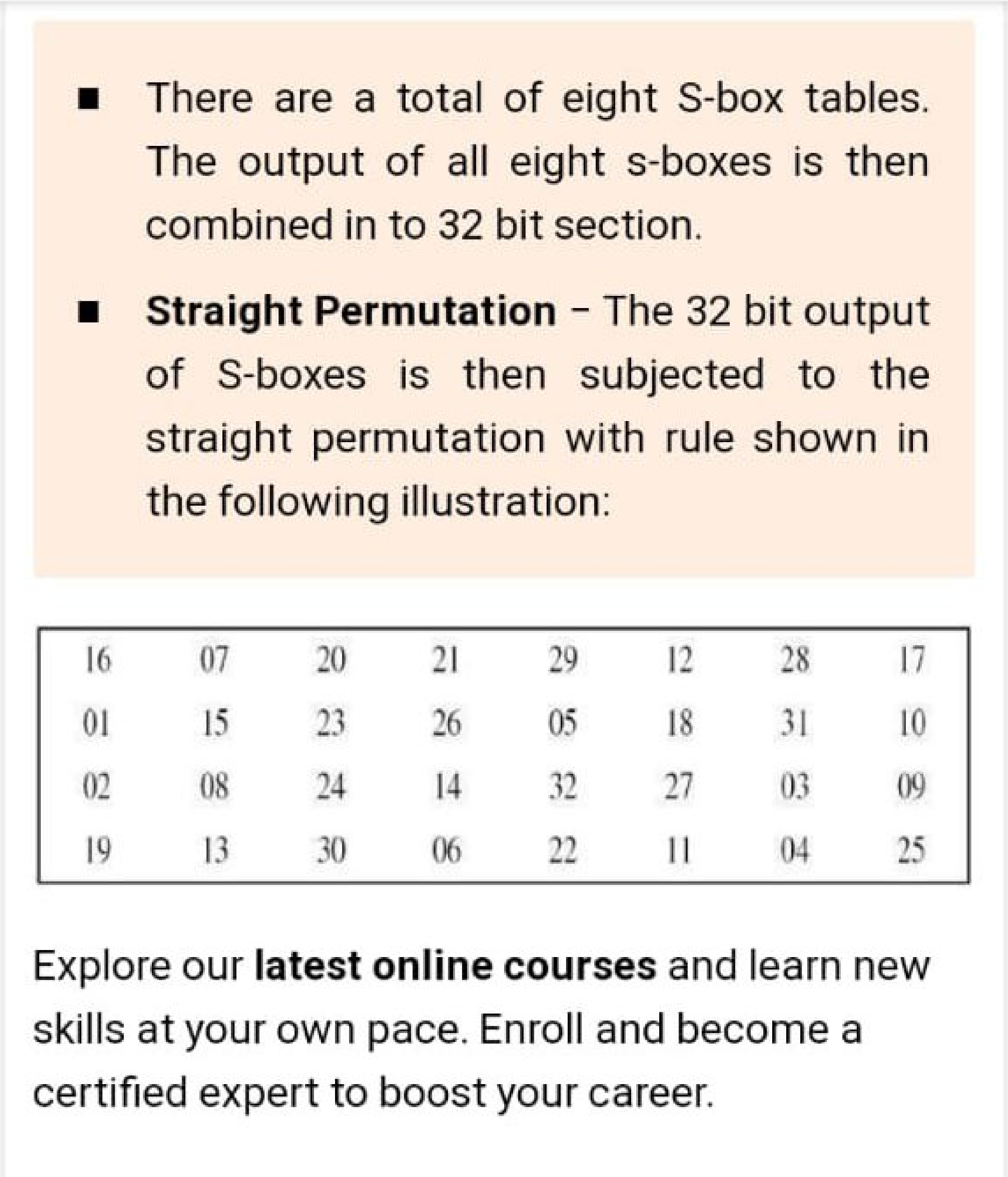
- **XOR (Whitener).** – After the expansion permutation, DES does XOR operation on the expanded right section and the round key. The round key is used only in this operation.
- **Substitution Boxes.** – The S-boxes carry

out the real mixing (confusion). DES uses 8 S-boxes, each with a 6-bit input and a 4-bit output. Refer the following illustration −



- The S-box rule is illustrated below −



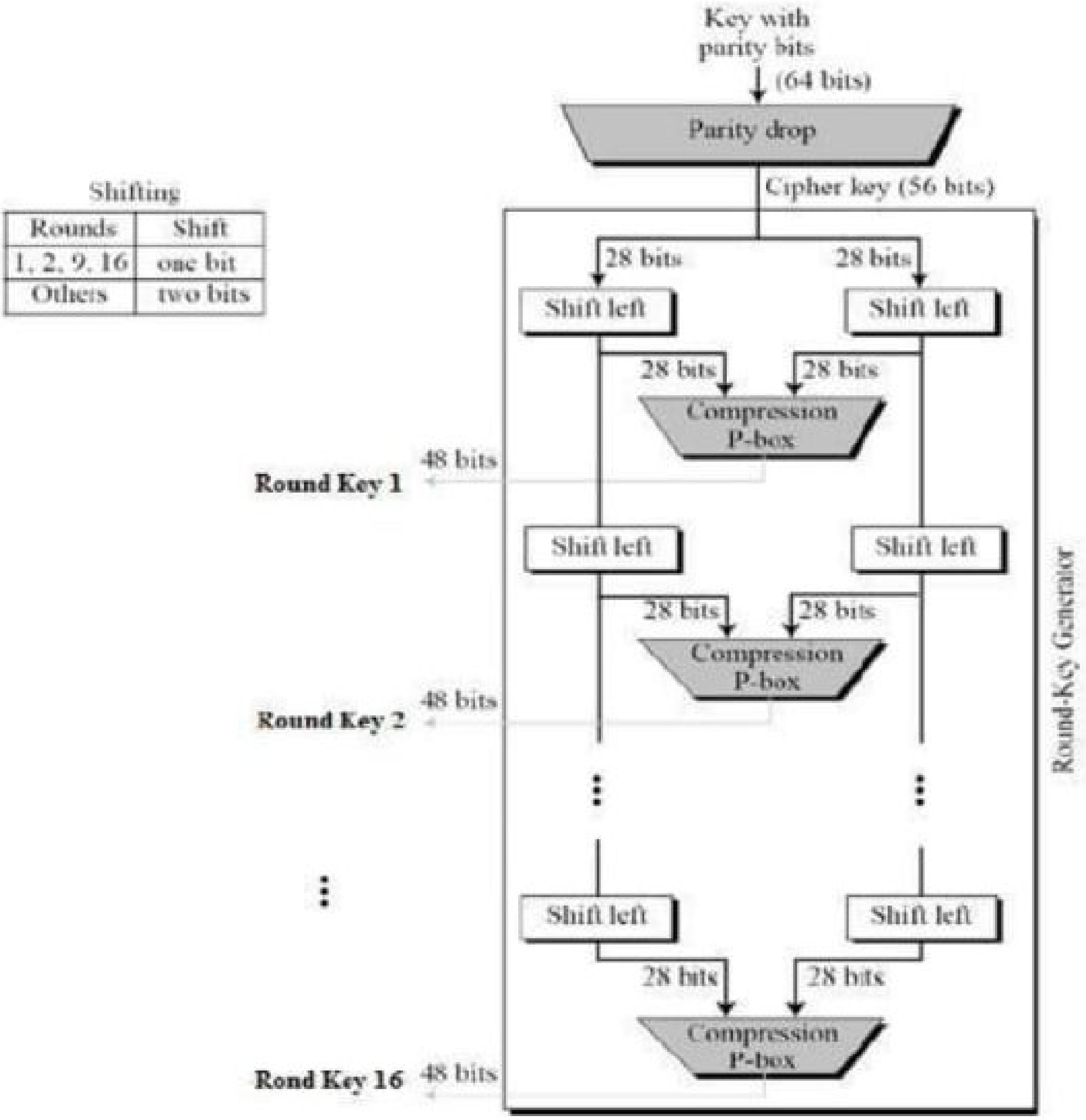


- There are a total of eight S-box tables. The output of all eight s-boxes is then combined in to 32 bit section.
- **Straight Permutation** – The 32 bit output of S-boxes is then subjected to the straight permutation with rule shown in the following illustration:

16	07	20	21	29	12	28	17
01	15	23	26	05	18	31	10
02	08	24	14	32	27	03	09
19	13	30	06	22	11	04	25

Key Generation

The round-key generator creates sixteen 48-bit keys out of a 56-bit cipher key. The process of key generation is depicted in the following illustration –



Shifting

Rounds	Shift
1, 2, 9, 16	one bit
Others	two bits

The logic for Parity drop, shifting, and Compression P-box is given in the DES description.