Data Encryption Standard (DES), Double DES and Triple DES

Last updated on Jan, 2025

We live so much of our lives today on the internet. Whether it's for storing our personal information, finding entertainment, making purchases, or doing our jobs, our society relies increasingly on an online presence.

This increased dependence on the internet means that information security is more important than ever. The stakes are too high now. Users need to know that their sensitive data is kept confidential, unmodified, and readily available to authorized readers.

Data encryption is just one weapon in the cybersecurity arsenal, but it's one of the oldest and most used. And since no discussion about data encryption is complete without talking about DES, here we are!

Data Encryption Standard (DES)

DES stands for Data Encryption Standard. There are certain machines that can be used to crack the DES algorithm. The DES algorithm uses a key of 56-bit size. Using this key, the DES takes a block of 64-bit plain text as input and generates a block of 64-bit cipher text.

The DES process has several steps involved in it, where each step is called a round. Depending upon the size of the key being used, the number of rounds varies. For example, a 128-bit key requires 10 rounds, a 192-bit key requires 12 rounds, and so on.

Take a look at the video below which explains steps for encryption and decryption in detail, future of the Data Encryption Standard in cryptography and live example to further highlight the characteristics of DES encryption.

After having gone through and understanding what is DES, let us look into ways to improve our cybersecurity skills.

What is the DES Algorithm in Cyber Security?

The DES (Data Encryption Standard) algorithm is a symmetric-key block cipher created in the early 1970s by an IBM team and adopted by the National Institute of Standards and Technology (NIST). The algorithm takes the plain text in 64-bit blocks and converts them into ciphertext using 48-bit keys.

Since it's a symmetric-key algorithm, it employs the same key in both encrypting and decrypting the data. If it were an asymmetrical algorithm, it would use different keys for encryption and decryption.

History of DES Algorithm

DES is based on the Feistel block cipher, called LUCIFER, developed in 1971 by IBM cryptography researcher Horst Feistel. DES uses 16 rounds of the Feistel structure, using a different key for each round.

DES became the approved federal encryption standard in November 1976 and was subsequently reaffirmed as the standard in 1983, 1988, and 1999.

DES's dominance came to an end in 2002, when the Advanced Encryption Standard (AES) replaced the DES encryption algorithm as the accepted standard, following a public competition to find a replacement. The NIST officially withdrew FIPS 46-3 (the 1999 reaffirmation) in May 2005, although Triple DES (3DES), remains approved for sensitive government information through 2030.

Initial Permutation (IP)

The plain text is divided into smaller chunks of 64-bit size. The IP is performed before the first round. This phase describes the implementation of the transposition process. For example, the 58th bit replaces the first bit, the 50th bit replaces the second bit, and so on. The resultant 64-bit text is split into two equal halves of 32-bit each called Left Plain Text (LPT) and Right Plain Text (RPT).

Step 1: Key Transformation

We already know that the DES process uses a 56-bit key, which is obtained by eliminating all the bits present in every 8th position in a 64-bit key. In this step, a 48-bit key is generated. The 56-bit key is split into two equal halves and depending upon the number of rounds the bits are shifted to the left in a circular fashion.

Due to this, all the bits in the key are rearranged again. We can observe that some of the bits get eliminated during the shifting process, producing a 48-bit key. This process is known as compression permutation.

Step 2: Expansion Permutation

Let's consider an RPT of the 32-bit size that is created in the IP stage. In this step, it is expanded from 32-bit to 48-bit. The RPT of 32-bit size is broken down into 8 chunks of 4 bits each and extra two bits are added to every chunk, later on, the bits are permutated among themselves leading to 48-bit data. An XOR function is applied in between the 48-bit key obtained from step 1 and the 48-bit expanded RPT.

Now in our understanding of what is DES, let us next look into the DES algorithm steps.

DES has the exact structure of a Feistel cipher

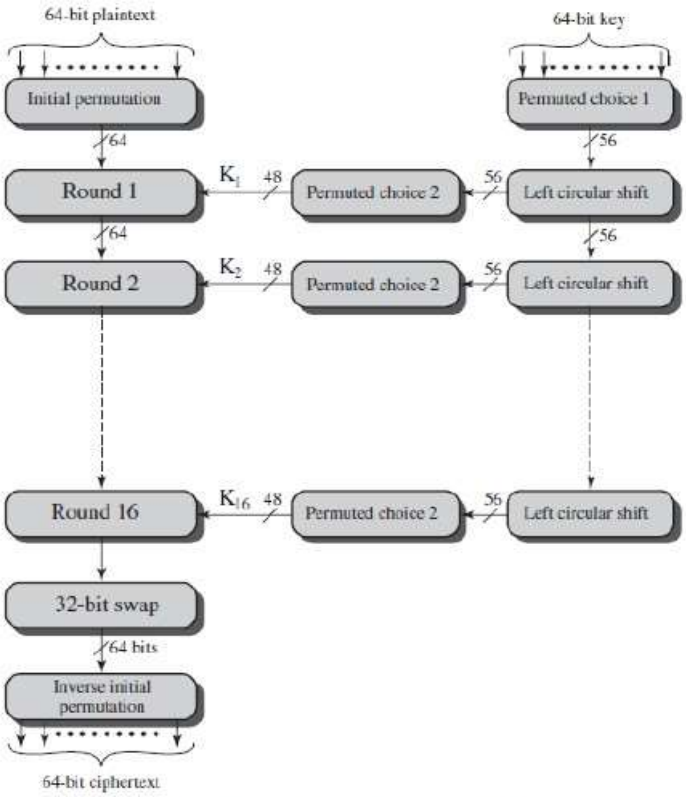


Figure 3.5 General Depiction of DES Encryption Algorithm

DES Algorithm Steps

To put it in simple terms, DES takes 64-bit plain text and turns it into a 64-bit ciphertext. And since we're talking about asymmetric algorithms, the same key is used when it's time to decrypt the text.

The algorithm process breaks down into the following steps:

1. The process begins with the 64-bit plain text block getting handed over to an initial permutation (IP) function.
2. The initial permutation (IP) is then performed on the plain text.
3. Next, the initial permutation (IP) creates two halves of the permuted block, referred to as Left Plain Text (LPT) and Right Plain Text (RPT).
4. Each LPT and RPT goes through 16 rounds of the encryption process.
5. Finally, the LPT and RPT are rejoined, and a Final Permutation (FP) is performed on the newly combined block.
6. The result of this process produces the desired 64-bit ciphertext.

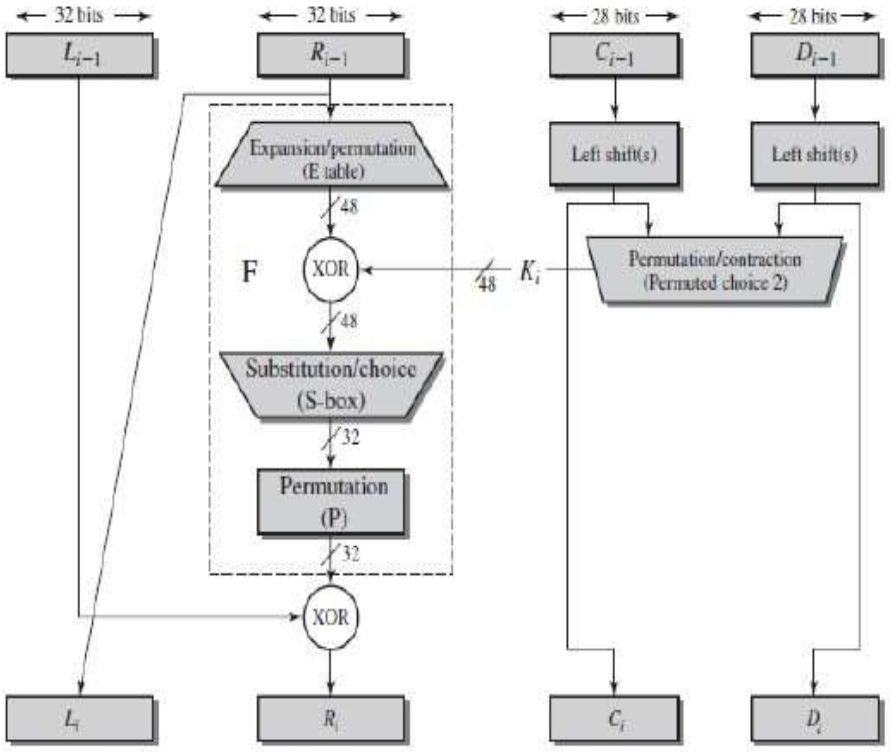


Figure 3.6 Single Round of DES Algorithm

The encryption process step (step 4, above) is further broken down into five stages:

1. Key transformation
2. Expansion permutation
3. S-Box permutation
4. P-Box permutation
5. XOR and swap

For decryption, we use the same algorithm, and we reverse the order of the 16 round keys.

Next, to better understand what is DES, let us learn the various modes of operation for DES.

- The round function f can be summarized as follows:

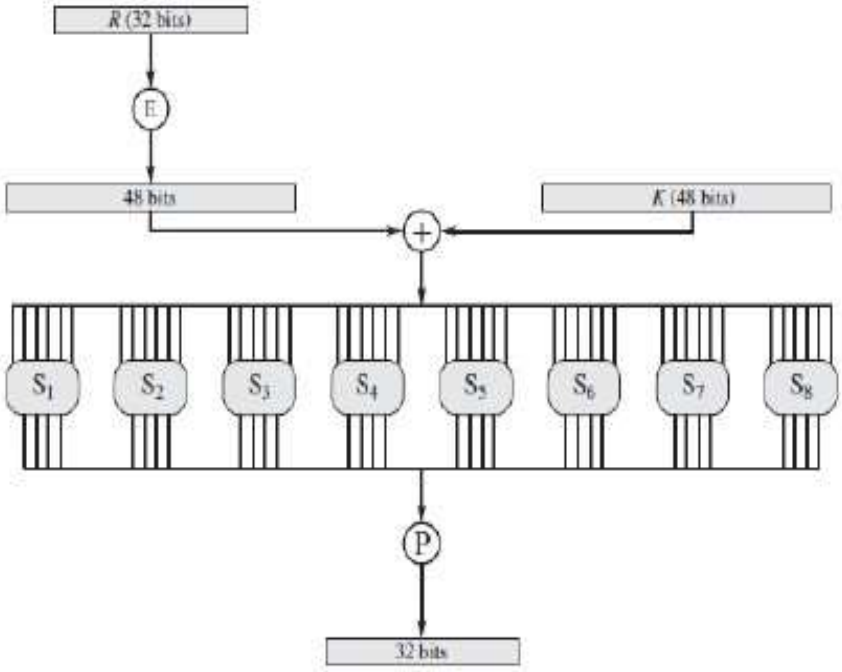


Figure 3.7 Calculation of F(R, K)

DES Modes of Operation

Experts using DES have five different modes of operation to choose from.

- Electronic Codebook (ECB). Each 64-bit block is encrypted and decrypted independently
- Cipher Block Chaining (CBC). Each 64-bit block depends on the previous one and uses an Initialization Vector (IV)
- Cipher Feedback (CFB). The preceding ciphertext becomes the input for the encryption algorithm, producing pseudorandom output, which in turn is XORed with plaintext, building the next ciphertext unit
- Output Feedback (OFB). Much like CFB, except that the encryption algorithm input is the output from the preceding DES
- Counter (CTR). Each plaintext block is XORed with an encrypted counter. The counter is then incremented for each subsequent block

We will next improve our understanding of what DES is, let us look into the DES implementation and testing.

DES Implementation and Testing

DES implementation requires a security provider. However, there are many available providers to choose from, but selecting one is the essential initial step in implementation. Your selection may depend on the language you are using, such as Java, Python, C, or MATLAB.

Once you decide on a provider, you must choose whether to have a random secret key generated by the KeyGenerator or create a key yourself, using a plaintext or byte array.

It's also essential to test the encryption to make sure it is properly implemented. You can find a testing procedure that will do the trick using the recurrence relation found on GitHub.

Now that we have come so far in our understanding of what is DES, let us next look into the reasons to learn DES.

Applications of DES Algorithm

In this section, we are going to learn about some of the applications of the DES Algorithm.

1. It is used in random number generation
2. It is deployed when not-so-strong encryption is needed
3. It is used to develop a new form of DES, called Triple DES (using a 168-bit key formed using three keys)

Difference Between DES and AES algorithms

DES	AES
Used to encrypt plain text of 64-bit	Used to encrypt plain text of 128-bit
The key is of 56-bit size.	The key is of different sizes such as 128-bits, 192-bits, and so on
Less secure than AES	More secure than DES
It can be broken by brute force attacks	To date, AES has not been attacked
It is based on Feistel network	It is based on permutation and substitution network

Advantages and Disadvantages of DES Algorithm

The advantages of the DES algorithm:

1. It is set as a standard by the US government.
2. When compared to the software, it works faster on hardware.
3. Triple DES, used a 168-bit key which is very hard to crack.

The disadvantages of the DES algorithm:

1. Weakly secured algorithm.
2. There is a threat from Brute force attacks.
3. A DES cracker machine known as Deep Crack is available in the market.

Steps for Encryption

There are multiple steps involved in the steps for data encryption. They are:

1. Permutate the 64-bits in the plain text and divide them into two equal halves.
2. These 32-bit chunks of data will undergo multiple rounds of operations.
3. Apply XOR operation in between expanded right plain text and the compressed key of 48-bit size.
4. The resultant output is sent to the further step known as S-box substitution.
5. Now apply the XOR function to the output and the left plain text and store it in the right plain text.
6. Store the initial right plain text in the left plain text.
7. Both the LPT and RPT halves are forwarded to the next rounds for further operations.
8. At the end of the last round, swap the data in the LPT and RPT.
9. In the last step, apply the inverse permutation step to get the cipher text.

Steps for Decryption

The steps involved in the steps for data decryption are:

1. The order of the 16 48-bit keys is reversed such that key 16 becomes key 1, and so on.

2. The steps for encryption are applied to the ciphertext.

Q) If DES is Becoming Irrelevant, Why Learn It?

Despite DES losing the lofty position of being the go-to data encryption standard algorithm, it's still worth learning. There will always be room for the DES algorithm in cryptography because it was the foundation for subsequent encryption algorithms. If you understand the origins of data encryption, you will consequently have an easier time grasping the basics of current encryption methods.

Strength of Data encryption standard (DES)

Data encryption standard (DES) is a symmetric key block cipher algorithm. The algorithm is based on Feistel network. The algorithm uses a 56-bit key to encrypt data in 64-bit blocks. There are mainly two categories of concerns about the strength of Data encryption standard. They are:

1. Concerns about the particular algorithm used.
2. Concerns about the usage of key of size 56-bit.

The first concern regarding the algorithm used addresses the possibility of cryptanalysis by making use of the DES algorithm characteristics. A more severe concern is about the length of secret key used. There can be

2^{56} (approximately 7.2×10^{16}

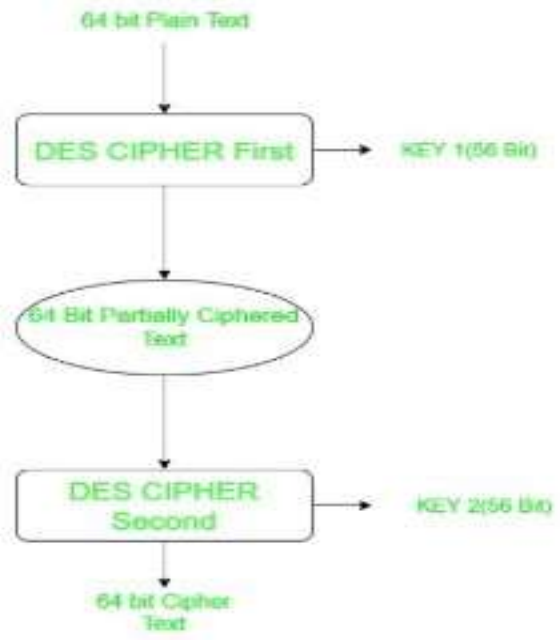
keys) possible keys with a key length of 56 bits. Thus, a brute force attack appears to be impractical. Assuming that on an average one has to search half the key space, to break the cipher text, a system performing one DES encryption per microsecond might require more than thousand years. But, the assumption of one DES encryption per microsecond is too conservative. In July 1998, DES was finally proved to be insecure when the Electronic Frontier Foundation (EFF) had broken a DES encryption. The encryption was broken with the help of a special-purpose “DES cracker” machine. It was reported that the attack took less than 3 days. Simply running through all possible keys won’t result in cracking the DES encryption. Unless known plain text is given, the attacker must be able to differentiate the plain text from other data. Some degree of knowledge about the target plain text and some techniques for automatically distinguishing plain text from garble are required to supplement the brute-force approach. If brute force attack is the only means to crack the DES encryption algorithm, then using longer keys will obviously help us to counter such attacks. An algorithm is guaranteed unbreakable by brute force if a 128- bit key is used. The differential cryptanalysis, linear cryptanalysis, are examples for statistical attacks on DES algorithm. Few of the important alternatives for DES are AES (Advanced Encryption Standard) and triple DES.

Double DES and Triple DES

As we know the Data encryption standard (DES) uses 56 bit key to encrypt any plain text which can be easily be cracked by using modern technologies. To prevent this from happening double DES and triple DES were introduced which are much more secured than the original DES because it uses 112 and 168 bit keys respectively. They offer much more security than DES.

Double DES:

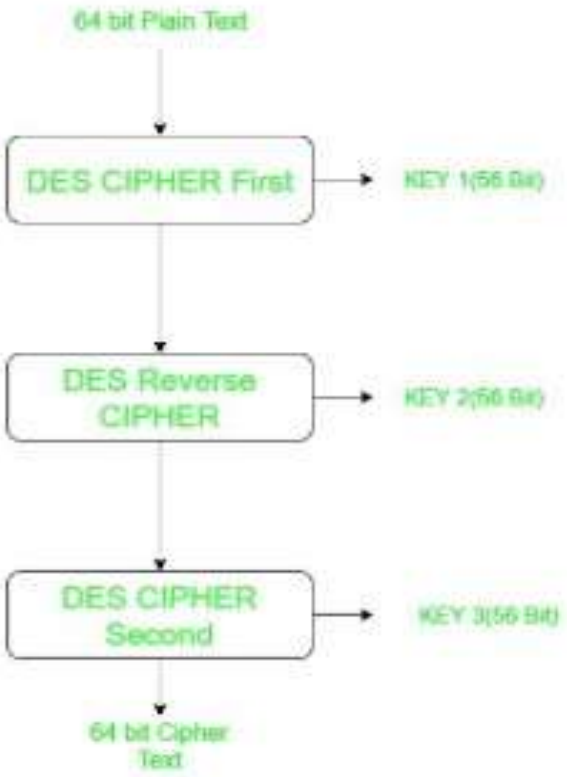
Double DES is an encryption technique which uses two instance of DES on same plain text. In both instances it uses different keys to encrypt the plain text. Both keys are required at the time of decryption. The 64 bit plain text goes into first DES instance which then converted into a 64 bit middle text using the first key and then it goes to second DES instance which gives 64 bit cipher text by using second key.



However double DES uses 112 bit key but gives security level of 2^56 not 2^112 and this is because of meet-in-the middle attack which can be used to break through double DES.

Triple DES:

Triple DES is an encryption technique which uses three instance of DES on same plain text. It uses three different types of key choosing technique in first all used keys are different and in second two keys are same and one is different and in third all keys are same.



Note that just because DES is no longer the NIST federal standard, it doesn't mean that it's no longer in use. Triple DES is still used today, but it's considered a legacy encryption algorithm. Note that NIST plans to disallow all forms of Triple-DES from 2024 onward.

Triple DES is also vulnerable to meet-in-the middle attack because of which it gives total security level of 2^112 instead of using 168 bit of key. The block collision attack can also be done because of short block size and using same key to encrypt large size of text. It is also vulnerable to sweet32 attack.