Data Encryption Standard (DES)

http://www.vocal.com

VOCAL Technologies, Ltd. encryption software libraries include DTCP, DES, AES, CCMP, RC4, and SHA algorithms in ANSI C and optimized assembly language for ADI, AMD-Alchemy, ARM, DSP Group, LSI Logic ZSP, MIPS and TI. This software is modular and can be executed as a single task under a variety of operating systems or it can execute standalone with its own kernel

The Data Encryption Standard (DES), is the name of the Federal Information Processing Standard (FIPS) 46-3, which describes the data encryption algorithm (DEA). The DEA is also defined in the ANSI standard X3.92. DEA is an improvement of the algorithm Lucifer developed by IBM in the early 1970s. IBM, the National Security Agency (NSA) and the National Bureau of Standards (NBS now National Institute of Standards and Technology NIST) developed the algorithm. The DES has been extensively studied since its publication and is the most widely used symmetric algorithm in the world.

The DES has a 64-bit block size and uses a 56-bit key during execution (8 parity bits are stripped off from the full 64-bit key). DES is a symmetric cryptosystem, specifically a 16-round Feistel cipher. When used for communication, both sender and receiver must know the same secret key, which can be used to encrypt and decrypt the message, or to generate and verify a Message Authentication Code (MAC). The DES can also be used for single-user encryption, such as to store files on a hard disk in encrypted form

NIST has re-certified DES (FIPS 46-1, 46-2, 46-3) every five years. FIPS 46-3 reaffirms DES usage as of October 1999, but single DES is permitted only for legacy systems. FIPS 46-3 includes a definition of triple-DES (TDES); TDES is "the FIPS approved symmetric algorithm of choice." Within a few years, DES and triple-DES will be replaced with the Advanced Encryption Standard (AES).

DES Encryption Modes of Operation:

- **ECB (Electronic Code Book)** . This is the regular DES algorithm. Data is divided into 64-bit blocks and each block is encrypted one at a time. Separate encryptions with different blocks are totally independent of each other. This means that if data is transmitted over a network or phone line, transmission errors will only affect the block containing the error. It also means, however, that the blocks can be rearranged, thus scrambling a file beyond recognition, and this action would go undetected. ECB is the weakest of the various modes because no additional security measures are implemented besides the basic DES algorithm. However, ECB is the fastest and easiest to implement, making it the most common mode of DES.

- **CBC (Cipher Block Chaining)**. In this mode of operation, each block of ECB encrypted ciphertext is XORed with the next plaintext block to be encrypted, thus making all the blocks dependent on all the previous blocks. This means that in order to find the plaintext of a particular block, you need to know the ciphertext, the key, and the ciphertext for the previous block. The first block to be encrypted has no previous ciphertext, so the plaintext is XORed with a 64-bit number called the Initialization Vector, or IV for short. So if data is transmitted over a network or phone line and there is a transmission error, the error will be carried forward to all subsequent blocks since each block is dependent upon the last. This mode of operation is more secure than ECB because the extra XOR step adds one more layer to the encryption process.

Custom Product Design Division
200 John James Audubon Parkway
Buffalo, New York 14228
716-688-4675

- **CFB (Cipher Feedback)**. In this mode, blocks of plaintext that are less than 64 bits long can be encrypted. Normally, special processing has to be used to handle files whose size is not a perfect multiple of 8 bytes, but this mode removes that necessity (Stealth handles this case by adding several dummy bytes to the end of a file before encrypting it). The plaintext itself is not actually passed through the DES algorithm, but merely XORed with an output block from it, in the following manner: A 64-bit block called the Shift Register is used as the input plaintext to DES. This is initially set to some arbitrary value, and encrypted with the DES algorithm. The ciphertext is then passed through an extra component called the M-box, which simply selects the left-most M bits of the ciphertext, where M is the number of bits in the block we wish to encrypt. This value is XORed with the real plaintext, and the output of that is the final ciphertext. Finally, the ciphertext is fed back into the Shift Register, and used as the plaintext seed for the next block to be encrypted. As with CBC mode, an error in one block affects all subsequent blocks during data transmission. This mode of operation is similar to CBC and is very secure, but it is slower than ECB due to the added complexity.

- **OFB (Output Feedback)**. This is similar to CFB mode, except that the ciphertext output of DES is fed back into the Shift Register, rather than the actual final ciphertext. The Shift Register is set to an arbitrary initial value, and passed through the DES algorithm. The output from DES is passed through the M-box and then fed back into the Shift Register to prepare for the next block. This value is then XORed with the real plaintext (which may be less than 64 bits in length, like CFB mode), and the result is the final ciphertext. Note that unlike CFB and CBC, a transmission error in one block will not affect subsequent blocks because once the recipient has the initial Shift Register value, it will continue to generate new Shift Register plaintext inputs without any further data input. However, this mode of operation is less secure than CFB mode because only the real ciphertext and DES ciphertext output is needed to find the plaintext of the most recent block. Knowledge of the key is not required.

DES Encryption Standard