

**Modification Of Des Algorithm****CRS BHARDWAJ**

Modibada, Jabalpur (Mp), India

Abstract :

This research paper discusses the modification of DES algorithm, which is the science of data encryption, a technology that provides for a safe, secure, and private information exchange. PN Generator produce the infinite random numbers which can be used to modify the DES algorithm to make it more critical to decipher. The modified encryption of data cannot be deciphered by DES algorithm. It enables you to send the secure data between two computers on private wireless link.

Keywords: *wireless communication, security, pseudorandom generator, advance cryptography*

1.Introduction

Data security plays a crucial and critical role in modern times for businesses and in military wars. Information is passed over the Internet through ecommerce and m-commerce channels. To address these security concerns, various security protocols that are of Symmetric-key and asymmetric-key type have been developed. In this paper, the modification of DES algorithm by random numbers has been produced.[1]

The organization of the paper is as follows. In section 2, we present the basics of symmetric key encryption techniques in section 3; we explain the essentials of asymmetric key encryption techniques. In section 4, Materials, apparatus and procedures has been presented. We will conclude the paper with directions for future research.

2. Symmetric-Key Encryption Techniques

Symmetric Encryption (also known as symmetric-key encryption, single-key encryption, one-key encryption and private key encryption) is a type of encryption where the same secret key is used to encrypt and decrypt information or there is a simple transform between the two keys.

A secret key can be a number, a word, or just a string of random letters. Secret key is applied to the information to change the content in a particular way. This might be as simple as shifting each letter by a number of places in the alphabet. Symmetric algorithms require that both the sender and the receiver know the secret key, so they can encrypt and decrypt all information. In any symmetric-key encryption technique, both encryption and decryption process are carried out using a single key. These algorithms are efficient, are secure, execute at high speeds, and consume less computer resources of memory and processor time. However, symmetric key cryptographic techniques suffer from the disadvantages of Key distribution problem, Key management problem and inability to digitally sign a message. Despite these drawbacks, numerous secure symmetric key encryption algorithms such as AES/Rijndael, Blowfish, CAST5, DES, IDEA, RC2, RC4, RC6, Serpent, TripleDES, Twofish, TDES, AES, have been developed. [2]

3.Asymmetric Key Encryption Techniques

The problem with secret keys is exchanging them over the Internet or a large network while preventing them from falling into the wrong hands. Anyone who knows the secret key can decrypt the message. One answer is asymmetric encryption, in which there are

two related keys--a key pair. A public key is made freely available to anyone who might want to send you a message. A second, private key is kept secret, so that only you know it. Any message (text, binary files, or documents) that are encrypted by using the public key can only be decrypted by applying the same algorithm, but by using the matching private key. Any message that is encrypted by using the private key can only be decrypted by using the matching public key.

This means that you do not have to worry about passing public keys over the Internet (the keys are supposed to be public). A problem with asymmetric encryption, however, is that it is slower than symmetric encryption. It requires far more processing power to both encrypt and decrypt the content of the message. The problems associated with symmetric-key cryptographic techniques were solved when asymmetric encryption mechanism was implemented. Here, instead of a single key, every person has a pair of keys. One key, called the public key is known to everyone and the other one, the private key is known only to the owner. There is a mathematical relationship between both these keys. Thus, if any message 'm' is encrypted using any of the key, it can be decrypted by the other portion. Various asymmetric encryption algorithms (RSA, Elgamal) have been implemented. [3]

4.Pretty Good Privacy (Pgp)

Pretty Good Privacy (PGP) is a data encryption and decryption computer program that provides cryptographic privacy and authentication for data communication. PGP is often used for signing, encrypting and decrypting texts, e-mails, files, directories and whole disk partitions to increase the security of e-mail communications. It was created by Phil Zimmermann in 1991. PGP encryption uses a serial combination of hashing, data compression, symmetric-key cryptography, and, finally, public-key cryptography; each step uses one of several supported algorithms. Each public key is bound to a user name and/or an e-mail address. The first version of this system was generally known as a web of trust to contrast with the X.509 system, which uses a hierarchical approach based on certificate authority and which was added to PGP implementations later. Current versions of PGP encryption include both options through an automated key management server.[4]

5.Data Encryption Standard (DES)

DES is the archetypal block cipher — an algorithm that takes a fixed-length string of plaintext bits and transforms it through a series of complicated operations into another ciphertext bit string of the same length. In the case of DES, the block size is 64 bits. DES also uses a key to customize the transformation, so that decryption can supposedly only be performed by those who know the particular key used to encrypt. The key ostensibly consists of 64 bits; however, only 56 of these are actually used by the algorithm. Eight bits are used solely for checking parity, and are thereafter discarded. Hence the effective key length is 56 bits, and it is never quoted as such. Every 8th bit of the selected key is discarded, that is, positions 8, 16, 24, 32, 40, 48, 56, 64 are removed from the 64 bit key leaving behind only the 56 bit key. Like other block ciphers, DES by itself is not a secure means of encryption. The algorithm's overall structure is shown in Figure 2.4: there are 16 identical stages of processing, termed rounds. There is also an initial and final permutation, termed IP and FP, which are inverses (IP "undoes" the action of FP, and vice versa). Before the main rounds, the block is divided into two 32-bit halves and processed alternately; this criss-crossing is known as the Feistel scheme.[5]

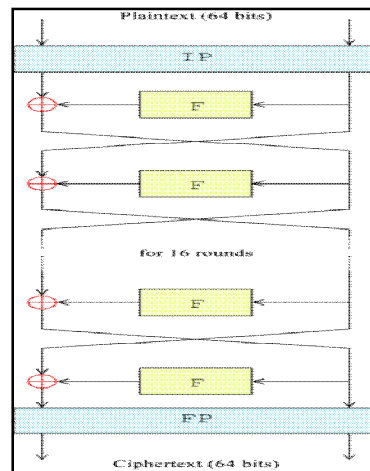


Figure 1: Feistel Structure of DES

The Feistel structure ensures that decryption and encryption are very similar processes — the only difference is that the sub keys are applied in the reverse order when decrypting. The rest of the algorithm is identical. This greatly simplifies implementation, particularly in hardware, as there is no need for separate encryption and decryption algorithms.

The \oplus symbol denotes the exclusive-OR (XOR) operation. The F-function scrambles half a block together with some of the key. The output from the F-function is then combined with the other half of the block, and the halves are swapped before the next round. After the final round, the halves are not swapped; this is a feature of the Feistel structure which makes encryption and decryption similar processes.[6]

6. Materials, Apparatus And Procedures

6.1. Implementation of DES

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
```

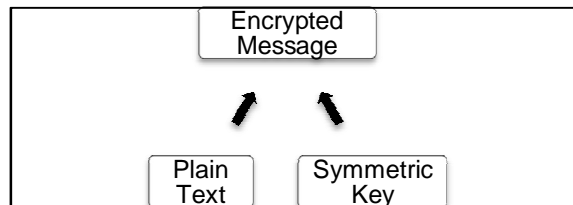


Figure2: Implementation of DES

```
#include<malloc.h>
#include<stdlib.h>
#include<math.h>
```

Figure 3: Data encryption standard

Data encryption program is implemented by using hexadecimal numbers. Program is coded in hexadecimal numbers. Implementation means install the software to the destination and make it to work there. Implementation is an ongoing process and can be achieved by one of the following methods:

- The implementation of dissertation is carried out in language “c” because of the following advantages.
- C is a building block for many other currently known languages.
- C is a compiled language versus an interpreted language.
- A lot of libraries are written in C.
- The main advantages of C language are that there is not much vocabulary to learn, and that the programmer can arrange for the program is very fast.
- C has features that allow the programmer to organize programs in a clear, easy, logical way.
- C is a portable language.

7.Modification Of Des Program

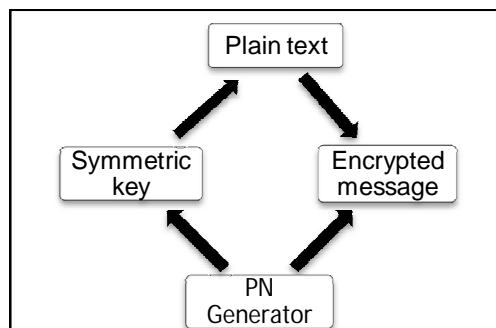
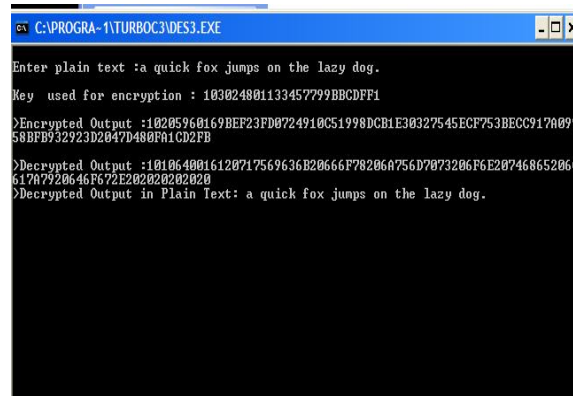


Figure 4: Modification of DES

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
#include<malloc.h>
#include<stdlib.h>
#include<math.h>
# define COMPUTATION
```

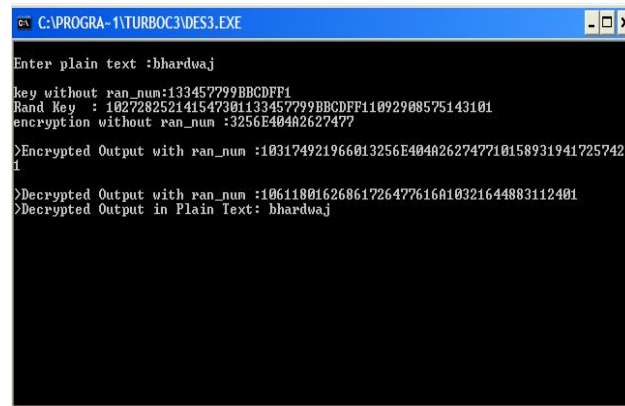
Modification of DES Program has been shown by using random numbers. In this modification random numbers are prefixed and suffixed to the main encryption. It increases the length of the encryption to hide the information. It cannot be decrypted by using the normal DES program.



```
C:\PROGRA-1\TURBOC3\DES3.EXE
Enter plain text :a quick fox jumps on the lazy dog.
Key used for encryption : 103024001133457799BBCDF1
>Encrypted Output :10205960169BEF23FD0724910C51998DCB1E38327545ECF753BECC917A099
58BFB932923D2047D480FA1CD2FB
>Decrypted Output :1010640016120717569636E20666F78206A756D7073206F6E20746865206C
61707920646F672E202020202020
>Decrypted Output in Plain Text: a quick fox jumps on the lazy dog.
```

Figure 5

8.Comparison Of Both Encryptions



```
C:\PROGRA-1\TURBOC3\DES3.EXE
Enter plain text :bhardwaj
key without ran_nun:133457799BBCDF1
Rand Key : 102728252141547301133457799BBCDF11092908575143101
encryption without ran_nun :3256E40402627477
>Encrypted Output with ran_nun :103174921966013256E40402627477101589319417257420
1
>Decrypted Output with ran_nun :10611001626861726477616A10321644083112401
>Decrypted Output in Plain Text: bhardwaj
```

Figure 6

Comparison of both encryptions of both encryptions has been shown in the diagram. Clearly the modification of DES program is not understandable.

9.Transmission Of Message

Cryptography has two main parts. First part deals with the cryptography at terminal ends. Second part deals with the secrecy during the transmission. To maintain secrecy at the

terminal ends, Hybrid Cryptography (symmetric key and PN Generator random number) is used to encrypt the plain text and then PN Generator is used to spread the message.

To encrypt the plaintext symmetric key and PN Generator random number are used.

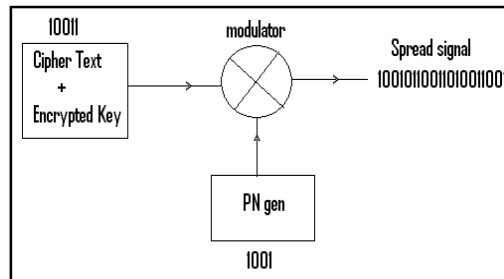


Figure 6

As shown in the diagram. The other input to the modulator is the PN gen. PN Sequence Generator block generates a sequence of pseudorandom binary numbers. Let the input to the modulator is 10011. Let the PN gen chip rate is 1001. The output of the modulator for the input bits 10011 will be 10010110011010011001 as shown in the diagram.

10.Result And Discussion

- Asymmetric Encryption is 100-1000 times slower than symmetric algorithms (RSA v. DES). Only code is transmitted. Actual message lies at the terminal ends. Code can be transmitted faster than asymmetric Key.
- The problem of distributing keys is solved because the codes are prepared locally. One person can have unlimited numbers of codes for different unlimited users. (A booklet of passwords can be prepared in advance and should be kept under lock and key. password may be changed daily by referring the serial numbers of the booklet).
- Symmetric keys are subject to a brute force attack where all keys in the key space are tried systematically to break the encryption. As we are using Customized advance cryptography, there is no chance to pick up the actual signals during the transmission by the intruders.
- Distribution of keys becomes a problem, especially if keys change frequently. Keys must be transmitted with extreme security because they allow access to all the information encrypted with them. For applications that extend throughout the world, this can be a very complex task.

- Face-to-face key exchange is done. One booklet may contain passwords for one month. After one month another booklet of password is issued.
- An alternate solution of hybrid cryptography has been found to increase the speed and to decrease the memory size requirement. Customized Advance Cryptography is the better solution which ensures secrecy at terminal ends and during the transmission of the text.
- Customized Advance Cryptography is very important during the wars where each message has its own value. It is also important in business transactions.
- Customized advance cryptography ensures user authentication, bandwidth sharing, and security from eavesdropping and immunity to interference, and difficulty in detection, data encryption and key management.

10.1.Limitations

- Using Customized Advance Cryptography can be a complex process and its concept is often difficult for some people to grasp.
- Both parties must be able to use Customized Advance Cryptography. It is impossible to use Customized Advance Cryptography unless people at both ends of the connection are capable of using this.

11. Conclusion

This research paper discusses the modification of DES algorithm, which is the science of data encryption, a technology that provides for a safe, secure, and private information exchange. PN Generator produce the infinite random numbers which can be used to modify the DES algorithm to make it more critical to decipher. The modified encryption of data cannot be deciphered by DES algorithm. It enables you to send the secure data between two computers on private wireless link.

Customized cryptography ensures user authentication, bandwidth sharing, and security from eavesdropping and immunity to interference, and difficulty in detection. Thus advance cryptography is better than other cryptographic techniques because it is efficient, fast and reliable technique.

12.Acknowledgement

I take great pleasure in recording a sincere gratitude and immense heartfelt thanks to the people who helped me in completing this dissertation by giving me their valuable time and guidance. I would like to acknowledge all of them.

I would like to thank Prof. Rajesh Kumar Pandey for his guidance and help. His constant inspiration and timely suggestions helped me in collection all the information I needed for my dissertation without much difficulty.

I also want express my feelings towards all the staff members of Computer Science and Engineering Department and towards all those who have continuously motivated during preparations of the dissertation.

Finally, I am highly obliged to all my family members for their support and blessings.

13.Reference

1. Bamford, J. (1983). *The Puzzle Palace: Inside the National Security Agency, America's most secret intelligence organization*. New York: Penguin Books.
2. Bamford, J. (2001). *Body of Secrets: Anatomy of the Ultra-Secret National Security Agency from the Cold War through the Dawn of a New Century*. New York: Doubleday.
3. Barr, T.H. (2002). *Invitation to Cryptology*. Upper Saddle River, NJ: Prentice Hall.
4. Bauer, F.L. (2002). *Decrypted Secrets: Methods and Maxims of Cryptology*, 2nd Ed. New York: Springer Verlag.
5. Belfield, R. (2007). *The Six Unsolved Ciphers: Inside the Mysterious Codes That Have Confounded the World's Greatest Cryptographers*. Berkeley, CA: Ulysses Press.
6. Denning, D.E. (1982). *Cryptography and Data Security*. Reading, MA: Addison-Wesley.
7. Diffie, W., & Landau, S. (1998). *Privacy on the Line*. Boston: MIT Press.
8. Electronic Frontier Foundation. (1998). *Cracking DES: Secrets of Encryption Research, Wiretap Politics & Chip Design*. Sebastopol, CA: O'Reilly & Associates.
9. Federal Information Processing Standards (FIPS) 140-2. (2001, May 25). *Security Requirements for Cryptographic Modules*. Gaithersburg, MD: National Institute of Standards and Technology (NIST). Retrieved from <http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf>
10. Overview of Cryptography by Alfred J. Menezes
11. M. Blaze, J. Feigenbaum, J. Ioannidis, A. Keromytis. *Cryptography—G. Julius Caesar*
12. Schneier, Bruce (1995-10-09). *Applied Cryptography*. New York: 11.
13. Schneier B. *Applied Cryptography*. John Wiley & Sons Inc., New York, New York, USA, 2nd edition, 1996.
14. Stallings W. *Cryptography and Network Security*. Prentice Hall, Upper Saddle
15. Garfunkel, Simson and Beth Rosenberg (2005). *RFID: Applications, Security and Privacy*. Addison-Wesley.
16. Garfunkel, Simson and Gene Spafford and Alan Schwartz (2003). *Practical UNIX and Internet Security*, 3rd Edition. O'Reilly and Associates.