Volume 3, Issue 10, October 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

# Provide the Security to a Web Service by using DES Cryptography Algorithm

**Shaymaa Mohammed Jawad Kadhim**
(*Transport Ministry, Iraq)*
*Bharati Vidyapeeth Deemed University, India*

**Manjusha Joshi, Dr.Shashank Joshi**
*Computer Engineering*
*Bharati Vidyapeeth Deemed University, India*

***Abstract— Nowadays, networks, internet applications and web services are becoming very popular. Security is the most important aspect concerning internet and web services. The valuable data stored on computers, servers and transmitted over the internet need to be secured information security features. Encryption algorithms play a major role for information, security and offers the necessary protection against data intruders' attacks by converting information from its normal form into an unreadable form. In first part of this paper, we are going to use the DES algorithm and apply it on our web service (an employing management system). In second part we will provide a comparison between two web services. One without applying on it the algorithm and the other using DES. The comparison is made on the basis of these three parameters :Response time, MTBF (mean time between failure) and MTTR (mean time to repair) .***

***Keywords— Web Service, security, DES, cryptography.***

## I. INTRODUCTION

Internet, web services, wired and wireless networks are becoming very popular day by day. A long with this data security becomes more and more important to protect privacy and commercial data. Cryptographic algorithms play a key role for providing the important data user security. There are many Encryption algorithms which are used for user data security. They are divided into two cryptographic mechanisms depending on what keys are used. We distinguish between symmetric and asymmetric encryptions. Symmetric key cryptography involves the usage of the same key for encryption and decryption. Asymmetric key cryptography uses the same key for encryption, and another, different key for decryption [1].

An encryption transforms a plain text into cipher text (secure data). Decryption is the reverse of the encryption process, it transforms a cipher text into plain text as show in figure (1) below.

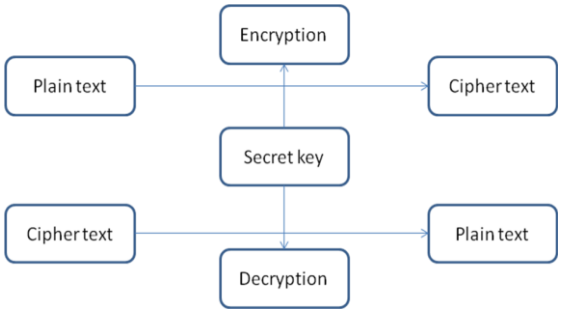


Figure (1) shows the encryption and decryption process by using the same key (symmetric key cryptographic algorithm).

For the same algorithm, an encryption using the long key is more difficult to break than with a short key. In the asymmetric encryption two keys are used, one is used to encrypt and other is used to decrypt the data [2]. In this paper, we are going to describe our work with the (Data Encryption Standard). It is an algorithm for the encryption of electronic data (data encryption standard). Here a brief introduction: DES was developed and published in the early 1970s. The encryption algorithm was also certified by ANSI (ANSI X3.92-1981) by ISO (DEA1). It has been used for over three decades. Recently DES has been proven vulnerable against brute force attacks, and, therefore, the popularity of DES has slightly decreased [decline 1]. DES is a block cipher. It encrypts data in 64- bit block. A 64-bit block of plain text goes on one end of the algorithm and a 64-bit block of cipher text comes out the other end. DES, is a symmetric algorithm: this means the same algorithm and key are used for both encryption and decryption. Over the decades, DES has been used to protect everything from databases in mainframe computers; to the communications links between ATMs and banks, to data transmissions between police cars and police stations. It is most likely that many times in your life, the security of everybody's data was protected by DES.

*© 2013, IJARCSSE All Rights Reserved* 

The encryption process is made in two permutations (P-boxes), which we call initial and final permutations, and sixteen Feistel rounds, as shown in figure (2) below:

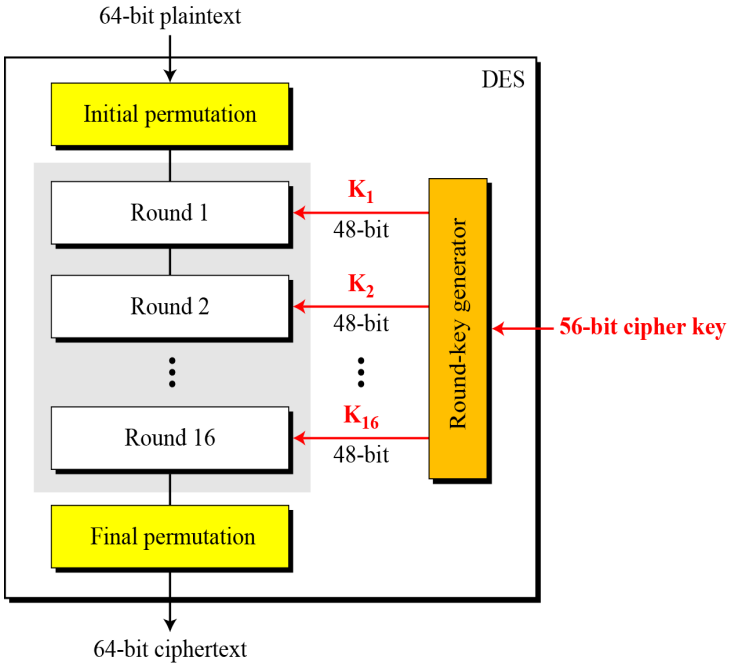


Figure (2): shows the DES algorithm.

## II. PREVIOUS WORKS

1. The paper [4] provides a description of DES algorithm.
2. The paper [5] presents notes on DES algorithm.
3. The paper [6] provides a comparison between DES, AES and Blowfish.

## III. PROPOSED ARCHITECTURE

In this paper we applied at first the DES algorithm in a web service (employing management system) that we have already to provide security for it. Second we analyse the performance of the web service depending on the parameters ( Response time, MTBF (mean time between failure) and MTTR (mean time to repair) ).

### 1) APPLYING DES ALGORITHM ON THE WEB SERVICE.

DES algorithm uses a key, it's length is about 56 bits. Actually, the initial key consists of 64 bits. However before the DES process even starts, every eighth bit of the key are discarded to produce a 56-bit key. Bit position 8, 24, 32,40,48,56 and 64 are discarded, as shown in the figure (3) below.

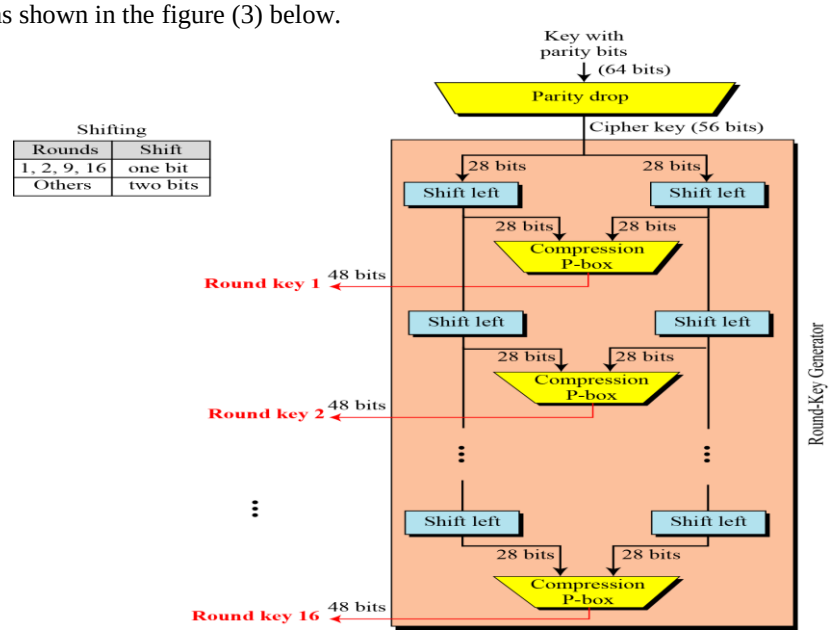


Figure (3): shows the key generation.

*© 2013, IJARCSSE All Rights Reserved* 

DES is based on the two fundamental attributes of cryptography: substitution (also called confusion) and transposition (also called as diffusion). DES consists of 16 steps, also called rounds. Each round performs the steps of substitution and transposition.

The steps of DES are:

1) In the first step, the 64-bit plain text block is handed over to an initial permutation (IP) function , The (IP) happens only once, this is before the first round.
2) Each of the 16 rounds, consists of the following steps, as shown in figure(4) below:

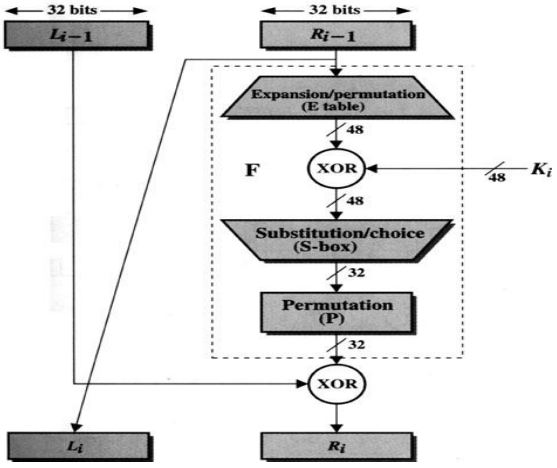


Figure (4): shows the overall steps of the round.

a) Key transformation: in each round a56-bit key is available and from this6-bit key, a different 48-bit sub-key is generated during each round using a process called key transformation.
For this, the 56-bit key is divided into two halves, each of 28bits. These halves are circularly shifted left by one or two position depending on the round.

b) Expansion permutation: recall the data after the permutation; we had two 32-bit plain text areas, called as left plain text (LPT) and right plain text (RPT). During the expansion permutation, the RPT is expanded from 32 bits to 48 bits. Besides increasing the bit size from 32 to 48, the bits are permuted as well.

I. The 32bit RPT is divided into 8 blocks, with each block consisting of 4 bits.
II. Each 4-bit block of the above step is then expanded to a corresponding 6-bit block. That is, per 4-bit block, 2 more bits are added. As shown in figure(5) below.

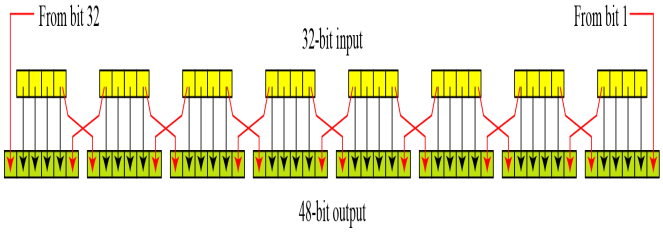


Figure (5) : Expansion permutation

c) S-box substitution: the substitution is performed by eight substitution boxes, as shown in figure (6) below. Each of the eight s- boxes has a 6-bit input and a 4-bit output. The 48-bit input block is divided into 8 sub-blocks (each containing 6 bits), and each such sub-block is given to an s-box. The s-box transforms the 6-bit input into a 4-bit output.
The output of all the s-boxes is then combined to form an 32-bit block , which is given to the next stage of a round, the p-box permutation.

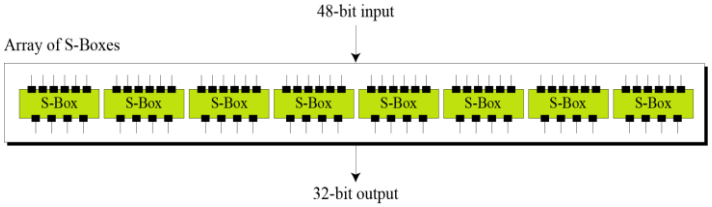


Figure (6) : S-boxes

*© 2013, IJARCSSE All Rights Reserved* 

d) P-box permutation: the output of S-box consists of 32 bits. These 32 bits are permuted using a P-box. This straightforward permutation mechanism involves simple permutation (Replacement of each bit with another bit, as specified in the p-box table, without any expansion or compression).

e) XOR and swap: at this juncture, the left half portion of the initial 64bit plain text block is XORed with the output produced by P-box permutation. The result of this XOR operation becomes the new right half. The old right half becomes the new left half, in a process of swapping.

3) Final permutation: At the end of the 16 rounds, the final permutation is performed only once. The output of the final permutation is the 64-bit encrypted block.

4) DES decryption: the same algorithm used for encryption in DES also works for decryption. The only difference between the encryption and decryption process is the reversal of key portions.

## 2) RESULT AND ANALYSIS

The score of this paper is to provide a performance analysis is comparison between a web service without security and the same web service with applied security by using the DES algorithm. The used web service is an employing management system. The web service is coded in c# the programming language C#.

The parameter that we are used to analysis the performance are :

- Response time.
- MTBF (mean time between failures).
- MTTR (mean time to repair).

A. The first set of experiments were conducted by measuring the response time from both web services, the web service with the cryptography algorithm and the web service without the algorithm. The result is shown in figure (6) below: The response time that of the encrypted web service (green bar) is higher than in the non-encrypted web service (blue bar). The results were measured in millisecond though. The difference can be considered almost be neglected compared to the gain of security to our web service.

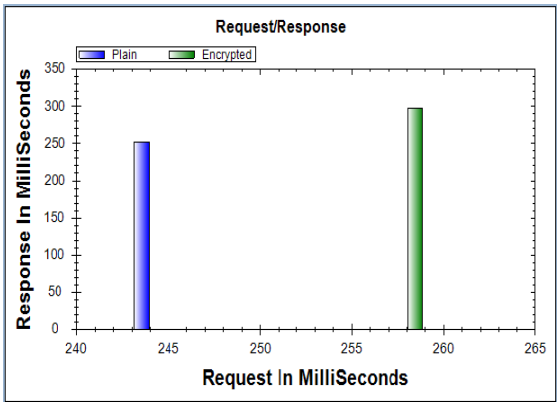


Figure (6) shows response time in encrypted and Non-encrypted systems

B. The second set of experiments was conducted measuring the MTBF (mean time between failures) from web services, the web service using the cryptography algorithm and the web service without the algorithm. The result is displayed in figure (7) below. For the MTBF we notice a similar result as for the response time: The value for the encrypted web service is slightly higher but the difference is only a few milliseconds

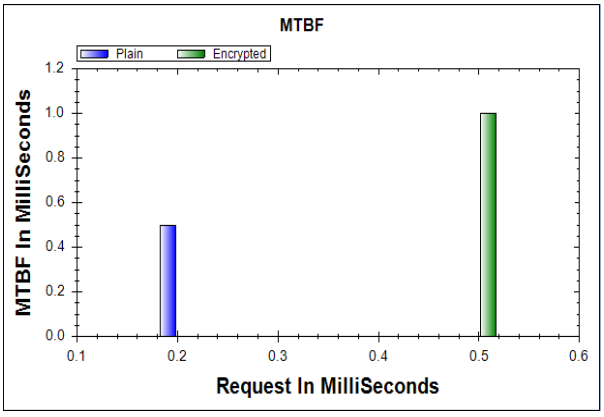


Figure (7) shows mean time between failures in Encrypted and non encrypted systems

*© 2013, IJARCSSE All Rights Reserved* 

C. The third set of experiments was conducted by measuring the MTTR (mean time to repair) from web services, the web service with cryptography algorithm and the web service without the algorithm. The result is shown in figure (8) below. The time required by the encrypted web service is almost the time that was required by the non-encrypted web service.

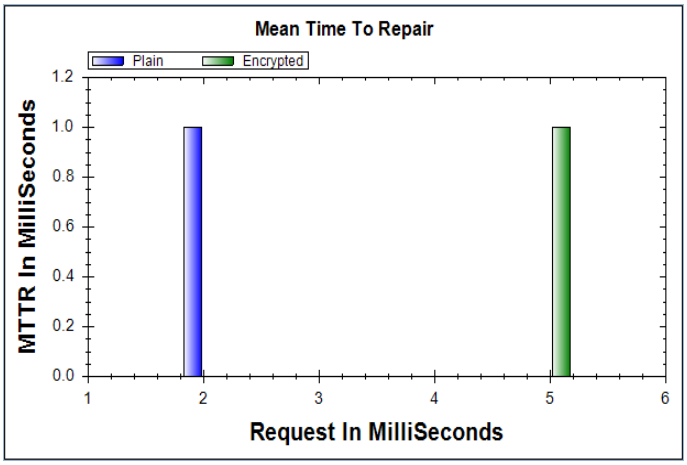


Figure (8) shows mean time to repair in Encrypted and Non-encrypted systems

## IV. CONCLUSION

In our project we have created our own web service, an employing management system. We applied a the DES cryptography algorithm, to the web service. After that we compared the performance of the web service without encryption and the web service with applied data encryption for the comparison three parameters were measured: (Response time, MTBF (mean time between failure) and MTTR (mean time to repair)) the results of the comparison show that a web service whose data are encrypted with the DES algorithm is almost as performing as a web service without any data encryption. The encryption showed only a little affect on the performance of our test system. At the same time encryption brings a big gain on the security of our web service.

### FUTURE WORK

Customer satisfaction is the main aim of every system. We should provide better and faster systems to the customer.
In our system we provide better security on web service (Employing management system) with minimal effect on its performance, all aims should be reached at minimal expenses.
Our future work will focus on necessary hardware improvements, with improving the performance and security at the same time.

## REFERENCE

[1] Atul Kahat (Cryptography and Network Security).
[2] Daemen, J., and Rijmen, V. "Rijndael: The Advanced Encrption Standard. "Dr. Dobb's Journal, March 2001,PP.137-139.
[3] https://www.schneier.com.
[4] http://www.facweb.iitkgp.ernet.in/~sourav/DES.pdf .
[5] www.site.uottawa.ca.
[6] Jawahar thakur, Nagesh kumar (DES,AES and blowfish :symmetric key cryptography algorithms simulation based performance analysis).

*© 2013, IJARCSSE All Rights Reserved* 