A NOVEL ENCRYPTION STANDARD (DES) WITH TIME HOPPING SEQUENCE FOR RFID SYSTEM

Mohamed Mostafa Abd Allah[1,2] , Gasim Alandjani[1),]
[1].Royal Commission, Yanbu Industrial Collage, EIET Department , Yanbu (KSA)
[2]. Minia University, Faculty of Engineering, Dept,. of Electrical, Communications and Electronics section, Egypt
E-mail: mmustafa@yic.edu.sa

ABSTRACT

This paper presents a novel authentication and encryption method for an RFID system. In this paper, DES (Data Encryption Standard), algorithm will be used to encrypt and verify accuracy data within an RFID tag. The new DES algorithm use permutation box with assigned a permutation pattern keeps changing in time. The proposed technique increase the difficulties for any hackers behave. The proposed algorithm measured and evaluated the time to read/write encrypted data and compared that with the time to read/write unencrypted data. Also, this study measured and analyzed the effect of encryption, on the accuracy of the read/write function in an RFID infrastructure. Experimental results show that proposed method has more efficient performance with the encryption data group, since the encryption time is less than 1 ms, it is negligible compared to the transmission time.

KEYWORDS

DES, Time hopping , RFID, Cipher, Encrypted Data.

1. INTRODUCTION

Privacy and security of RFID has become a very serious problem blocking its development [1]. Every member in RFID system should be solid enough to avoid the illegal attacks. A RFID system is made up of tag, reader, database and EPC network. Communication between tag and reader is the weakest point among all the channels. Tag suffers with unauthenticated access, tracing, eavesdropping and counterfeit [2]. For security on tag, the biggest challenge is to find an appropriate cryptography and implement it in circuit to meet power and cost requirement. How large the encryption cipher is the tag can hold with and whether a micro-processor can be accepted in a tag. This baseband performs a trial and gives out an example of a feasible secure tag. The DES algorithm implementation presented in this paper is used to as an appropriate cryptography for RFID tag. It based on the ECB mode [3],[4],[5] with 16 stages pipelining. A 16-stage pipelined DES Algorithm hardware implementation allows 16 data blocks to be processed simultaneously resulting in an impressive gain in speed. It also supports the use of different keys every clock cycle, thus improving overall security since users are not restricted to using the same key during any one session of data transfer. The DES algorithm is a private-key encryption algorithm, which is designed to encipher and decipher blocks of data consisting of 64 bits under control of a 64-bit key [6]. Deciphering must be accomplished by using the same key as for enciphering, but with the schedule of addressing the key bits altered so that the deciphering process is the reverse of the enciphering process. In this research, if encryption is required, then, in general, longer read and write time is necessary. Nevertheless, Using asymmetric encryption, one can use larger memory tags (since low memory tags can only incorporate the data increase allowed by symmetric encryption). Future research could be performed to completely characterize relationships between the use of asymmetric/symmetric encryption and tag type (low/high memory, passive/active, HF/UHF, etc) [2]. The current research establishes a

ISBN: 978-0-9853483-3-5 ©2013 SDIWC

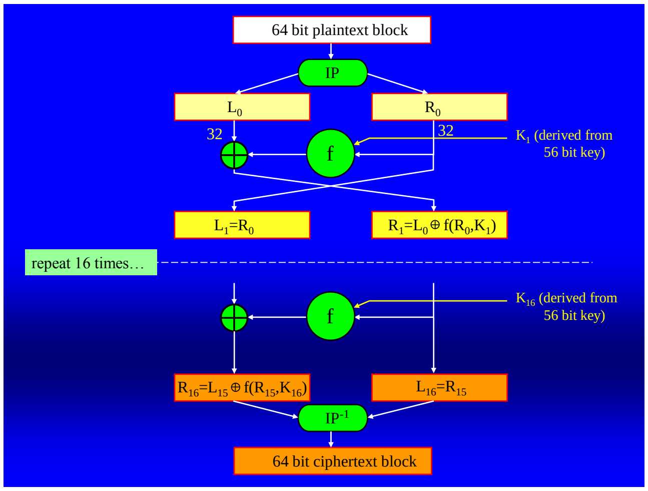

64 bit plaintext block
IP
L_0
R_0
32
32
f
K_1 (derived from 56 bit key)
$L_1=R_0$
$R_1=L_0 \oplus f(R_0,K_1)$
repeat 16 times…
f
K_{16} (derived from 56 bit key)
$R_{16}=L_{15} \oplus f(R_{15},K_{16})$
$L_{16}=R_{15}$
IP^{-1}
64 bit ciphertext block

Figure 1, Data Encryption Standard Outline

base line for the development of this entire genre of research. According to the results of the experiment, encryption of RFID tag data increases the transmission time and does not affect the accuracy of read/write. However, the encryption method is still potentially viable for the healthcare industry. Privacy protection and security issues are two of the main concerns of broadly applying a RFID system to healthcare, and encryption can solve most of the problems [10].The rest of this paper is organized as follows: Proposed DES architecture based on new Hopping Permutation boxes are presented in Section 2. Testing program, measured and evaluated time to read/write encrypted data are introduced in Section 3. Section 4 compares the achieved results with the previous DES implementations. Conclusions and references are given in Section 5.

2. DATA ENCRYPTION STANDARD

An outline of data encryption standard (DES) is shown in Fig. 1. It is a block cipher operating on 64- bits blocks of plaintext utilizing a 64-bits key. Every eight bit of the 64-bits key is used for parity checking and otherwise ignored. After an initial permutation, the 64-bits input is split into a right (R_0) and left half (L_0), each 32 bits in length.

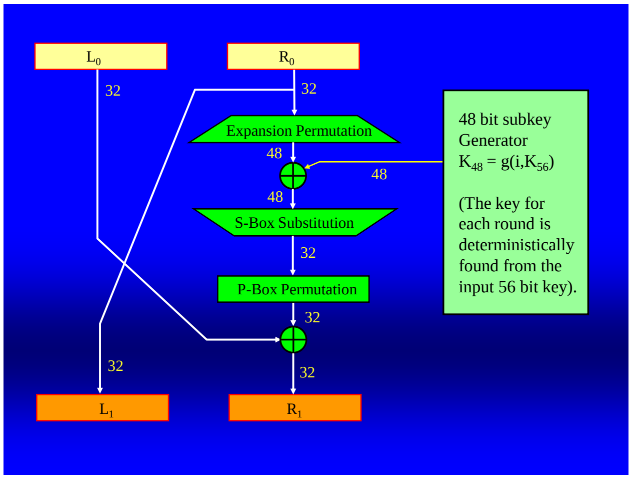


Figure 2, Function of the DES algorithm

DES has 16 iterations or rounds. In each round a function *f* is performed in which the data is combined with a 48-bits permutation of the key. After the 16th iteration, the right (R_{15}) and left (L_{15}) halves are concatenated and a final permutation, which is the inverse of the initial permutation, completes the algorithm. As shown in figure 2, the function *f* of the DES algorithm is made up of four operations. Firstly, the 32-bits right half of the plaintext R_0 is expanded to 48-bits and then XORed with a 48-bits sub-key K1. The result is fed into eight substitution boxes (s-boxes), which transform the 48-bits input to a 32-bits output. Finally, a straight permutation (P-permutation) is performed, the output of which is XORed with the initial left half, L_0 to obtain the new right half R1. The original right half R0 becomes the new left half L1.

2.1Proposed DES Algorithm

It is clear that the current algorithm shown at figure 1, can be attacked by any method of cryptanalysis linear, differential, brute and force [6],[12]. In order to make the DES algorithm more secure, the new DES shown in Figure 3 has been developed. The proposed scheme has a technique is called permutation hopping PH. As shown in Figure 3, and Figure4, the proposed Permutation box, contains several permutations patterns be used periodically with time. A timer controller has been added to this scheme, to synchronize sender and receiver. To avoid errors synchronization between the sender and receiver, an additional ciphered data has been transmitted to indicate the timer value that guide receiver to choose the correct permutation form. This behavior increases the size of transmitted bits, but it enhances the algorithm. We will look at permutation hopping. in detail in the following section.

2.2A Novel Permutation Hopping Technique.

In this method each permutation box is assigned a permutation pattern. Time is divided into slots. In the first time slot, a given key ciphered to the receiver using the first permutation pattern in its permutation hopping sequence. In the next time interval, it transmits using the second permutation pattern value in its permutation hopping sequence, and so on. This way, the transmit permutation pattern keeps changing in time. The proposed technique increase the difficulties for any hackers behave, linear, differential, brute and force [6]. Because the same plaintext is ciphered to different forms as a function of time, the proposed DES algorithm become stronger against any try to attack. In our design, we use six different permutations patterns. Flow chart shown in figure 4 introduces the method of changing the data permutation with time. Every time, the program checks the timer hopping sequence. According to the timer hopping sequence, the permutation pattern is selected from the permutation box. In a peer-to-peer frequency/time hopping (FH/TH) spread spectrum communication system, an attacker may try to synthesize the entire FH/TH pattern from some frequency/time slots successively observed. When the attacker observes successive twice frequency/time slots as the linear complexity (LC) [7] of the pattern, he can successfully synthesize the next frequency/time slots. Thus, from the view point of the system designers, the LCs of the FH/TH sequences in use should be as large as possible. Note that FH/TH communication systems using a few hundreds, or even a few thousands frequency/time slots are common in practice. Therefore, it is necessary to design non-binary sequences with “large” LCs, and with “little” increase in the hardware complexity.

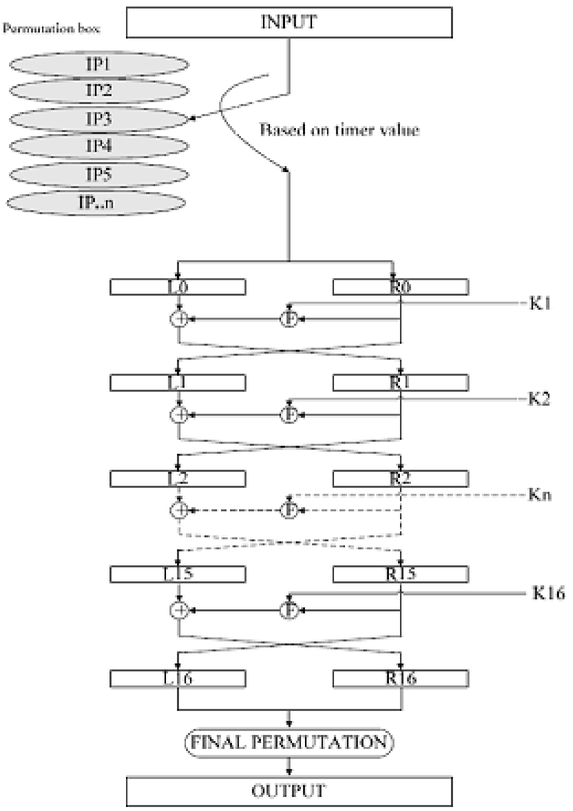


Figure (3): Proposed DES Algorithm

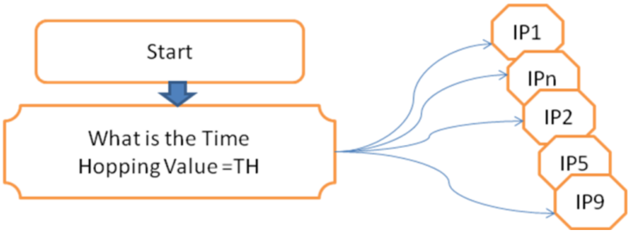
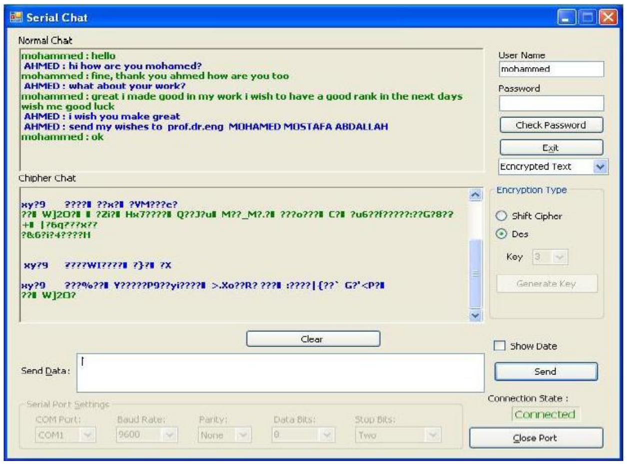


Figure (4): Flow chart of data permutation



Normal Chat

```
mohammed : hello
 AHMED : hi how are you mohamed?
mohammed : fine, thank you ahmed how are you too
 AHMED : what about your work?
mohammed : great i made good in my work i wish to have a good rank in the next days
wish me good luck
 AHMED : i wish you make great
 AHMED : send my wishes to  prof.dr.eng  MOHAMED MOSTAFA ABDALLAH
mohammed : ok
```

Chipher Chat

```
xy?9    ????▮ ??x?▮ ?VM???c?
??▮ W]2O?▮ ▮ ?Zi?▮ Hx7????▮ Q??J?u▮ M??_M?.?▮ ???o???▮ C?▮ ?u6??f?????:??G?8??
+▮ |?6q???x??
?&6?i?4????H


 xy?9   ????WI????▮ ?}?▮ ?X

xy?9    ???%??▮ Y?????P9??yi????▮ >.Xo??R? ???▮ :????|{??` G?'<P?▮
??▮ W]2O?
```

User Name: mohammed

Password:

Ecncrypted Text

Encryption Type: ○ Shift Cipher ● Des

Key: 3

☐ Show Date

Send Data:

Connection State : Connected

Serial Port Settings: COM Port: COM1; Baud Rate: 9600; Parity: None; Data Bits: 8; Stop Bits: Two

Figure 5: Graphical User Interface of Proposed Program

3.DESIGNED TESTING PROGRAM

The proposed technique encrypt data with assigned encryption methods (e.g. DES), write data to the tag, read data from the RFID tag, decrypt data, automatically measure the accuracy and R/W time in assigned test cycles and generate a test report. As shown at figure 5, the Graphical User Interface of the program is designed for the user to manipulate and set parameters test of multiple functions, including reading tag ID, encrypting and decrypting data, writing the data into the tag memory, reading the data from the tag memory, clearing the tag memory, measuring the R/W time, testing sequentially and creating test record files. The following parameters and functions are set by the user:

1. Encryption methods
2. Key values (like encryption key)
3. Number of test cycles
4. Delay time between cycles
5. Accuracy of write tag
6. Encryption function

3.1 Write / Read Verify Algorithm

The proposed write/read algorithm measured and evaluated the time to read/write encrypted data and compared that with the time to read/write unencrypted data. Before the original data is sent to the tag, the program first generates the hash code of this data and stores it in its memory[7]. The program writes the original data to the tag after splitting it into blocks of 4 bytes each. If the reader returns a success flag for the write operation, then the program requests the reader to get the actual tag data. The program generates the tag data's hash code and compares it with the original data's hash code to determine if two sets of data match. If the two hash codes match, the write cycle is deemed as a success. Comparison of the hash-codes allows detection of errors due to a variety of reasons including errors in transmission or reception modes (due to air interface and noise issues), errors due to defective tags, or errors due to problems in readers. Write accuracy is still limited by the accuracy of the read step, and this limitation can not be eliminated in our approach. The logic flow chart of the writing process is shown in Figure 6.

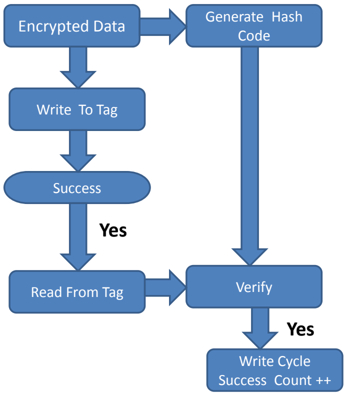


Figure6: Flow chart of the writing process

4. EXPERIMENT DESIGN & RESULTS

According to the test results, the performance of write and read cycles is a function of the distance between the reader and the tag and the delay time. However, encryption time is less than 1ms, it does not affect read and write accuracy significantly, even though the data is larger. To study how the RFID system would measures the average writing/ reading time, and the write / read accuracy, it is required to set some parameters such as encryption method, encryption keys, delay time between cycles, and enabling 100% write success function to tag. By using such variables, the experiment is separated into two major experiment groups: the unencrypted data group and the encrypted data group. Each major group has four subgroups, and each subgroup will be tested 1000 cycles. Each cycle is defined as a read and a write from each tag once. The settings of each parameter of the four subgroups are as follows [7]:

1. Group with Fail write function and NO delay function in each cycle (called group F_ND)
2. Group with Success write function and NO delay function in each cycle (called group S_ND)
3. Group with Fail write function and delay function in each cycle (called group F_D)
4. Group with Success write function and enable delay function in each cycle (called group S_D)

As shown before, since the encryption time is less than 1 millisecond, the additional time is negligible compared to the transmission time. Therefore, in the field of RFID for low-memory tags, the effect of encryption computing time on the performance can be neglected.

4.1.Unencrypted Data Group

Comparing the UF_ND group to the US_ND group, the write time of the US_ND group is 30 ~ 90 ms greater than the UF_ND group. Comparing the UF_D group to the US_D group, the write time of US_D is 8 ~ 64 ms greater than the UF_D group. It seems that, delay time between each cycle affects the result significantly. The amount of variation depends upon how many writes fail, since the failed write time is shorter than the successful write time. Comparing the differences between US_D&US_ND and UF_D&UF_ND, the trends are very similar. Therefore, the delay time function does not affect write time, but the write success function significantly affect on write time. The comparison chart is shown in Figure 7. In the read time comparison, most of the distance tests show similar times. However, at a distance of 10 inches, UF_ND takes a longer time to read. For a distance of 6 inches, the US_D group requires a longer read time. The two peak time value discrepancies are probably due to errors in the equipment. However, the trends of the four groups are similar. The comparison charts of read time are shown in Figure 8.

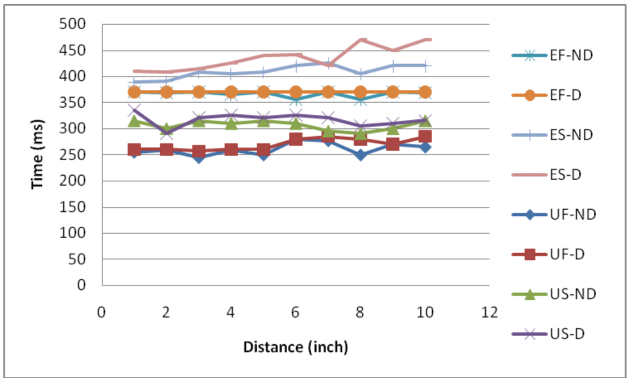


Figure 7. Write time comparasion of ecrypted data(E)& ucrypted data(U)

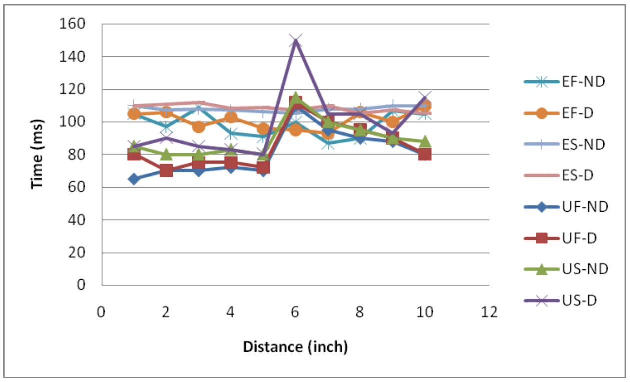


Figure 8. Read time comparasion of ecrypted data(E)& ucrypted data(U)

4.2.Encrypted Data Group

The second part of the test studies the data group encrypted by the proposed method of DES. When encrypting the data, the data size increases to 128 bytes in order to fit the maximum memory capacity of tag. As shown in figure 7, comparing the EF_ND and ES_ND groups, the write time difference is small and the trend difference is similar to that of the unencrypted data group. The write time of ES_ND in the encrypted data group is 25 ~ 118 ms more than the EF_ND group, and the difference in level is related to the measuring distance. In the comparison of EF_D and ES_D, the write time of ES_D has 14 ~ 95ms more than EF_D. The write time trend of write success groups, ES_ND and ES_D, seems to increase with read distance as shown at figure 7. By comparing the read time between delay groups and non-delay groups, delay groups take more time to read the data from the tag. The delay group, ES_D, is 20 ms slower than ES_ND at read distance of 7 inches, from 109 milliseconds to 89 milliseconds. The difference between EF_ND and EF_D is not as large as the differences in the write success function groups. The write Fail function group also has a saw tooth shaped distribution in the read time test as shown at figure 8.

4.3 Comparison between Unencrypted Data & Encrypted Data

The encrypted data group, compared to the unencrypted data group, the write and read times is also longer. By comparing F_ND of each group for R/W time difference, the write time of the encrypted data group takes 68 ms ~ 116 ms more than the unencrypted data. The read time of the encrypted data group is 36 ms slower than the unencrypted data group. However, at 6, 7, 10 inches, the unencrypted data group unexpectedly takes longer to read than the encrypted data group. The unencrypted group takes up to 14 ms, at 6 inches, longer than the encrypted group. The comparison is shown in Figure 9 and Figure 10. Focusing on the accuracy test of F_ND of the two groups, the write accuracy of the encrypted data is better than the unencrypted data for read distance of 1 ~ 5 inches. The biggest difference between the two data groups is 11%, from 93% to 82%, at a read distance of 2 inches. From a distance of 6 ~ 11 inches, the unencrypted data group has better write accuracy than the encrypted data group, and the difference can go as high as 24%, from

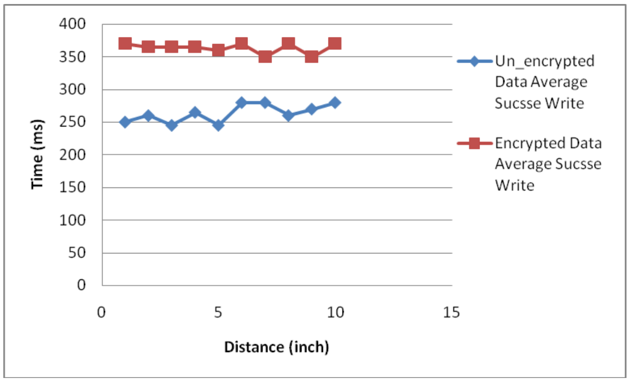


Figure 9. Write time comparasion of uncrypted & encrypted Data

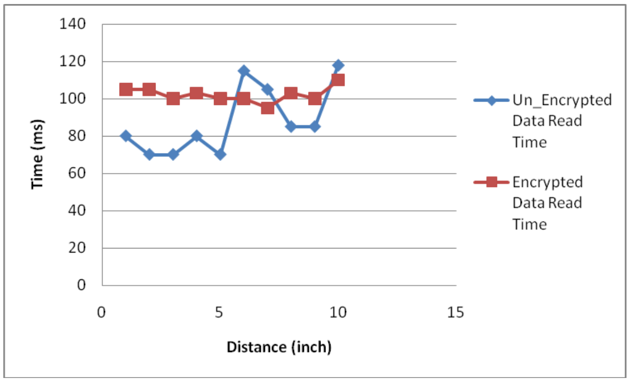


Figure 10. Read time comparasion of uncrypted & encrypted Data

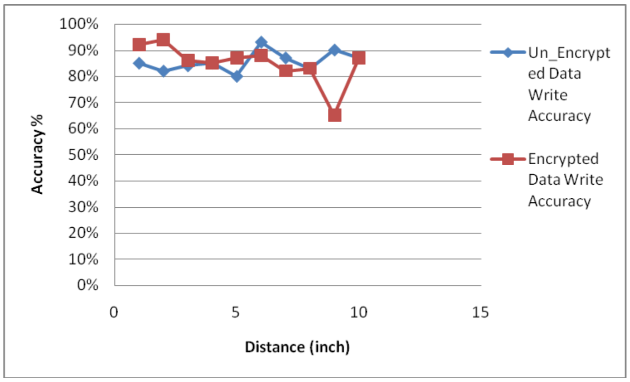


Figure 11 Write accuracy comparision of encrypted and Uncrypted data

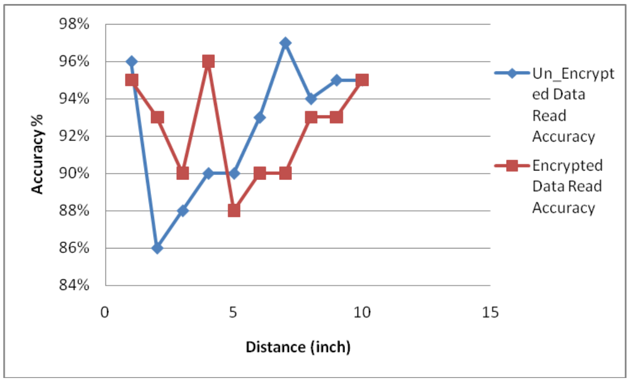


Figure 12 Read accuracy comparision of encrypted and Uncrypted data

92% to 68 %, at 9 inches. The write accuracy comparison chart is shown in Figure 11. In the comparison of read accuracy, the biggest difference between the two groups is the unencrypted group having 10% higher accuracy than encrypted group, from 98% to 88%, at a read distance of 7 inches. However, by comparing the trends of the two groups, distinguishing the effect of larger data length is not possible, shown in Figure 12.

5.CONCLUSION

In these tests, conclusions that can be made are that read accuracy and write accuracy are not significantly affected by encryption. However, the trend of the write accuracy is that it reduces as the distance increases. The data indicates that read accuracy does not drop with distance as significantly as does write accuracy. The delay time between cycles only affects read accuracy and does not show obvious affects on write accuracy. By adding the delay time, the results show that the reader is able to finish the previous reading instruction before starting the next one. The encrypted data test group that was 128 bytes has 40 bytes more than its unencrypted counterparts – an increase of 45%. The encryption time is negligible (< 1ms.) compared to the increase in the transmission time of the larger data size. In our comparison, the encrypted data group takes 70 ~ 120 milliseconds more for the transmission than the unencrypted data group for the write function which is an increase of 24% ~ 49%. In the read time comparison tests, most encrypted data groups take longer to read than their unencrypted counterparts. However, the tests also show that the unencrypted data group takes longer than the encrypted data group for certain distances.

6.REFERENCES

1. J. Al-Kassab, and W.-C. Rumsch, Challenges for RFID cross-industry standardization in the light of diverging industry requirements, IEEE systems journal, vol 2, no. 2, pp. 170-177, June (2008).
2. S. R. Aroor, and D. D. Deavours, Evaluation of the state of passive UHF RFID: An experimental approach, IEEE System Journal, vol. 1, no. 2, pp. 168-176, december (2007).
3. National Bureau of Standard (U.S.), “DES modes of operation,” Federal Information Processing Standard Publication 81, National Technical Information Service, Springfield, VA, December (1980).
4. K. M. A. Abd El-Latif, E. A. M. Hasaneen and H. F. A. Hamed, “Improved DES Algorithm Based On Variable Time Data Permutation, “International Conference for Advanced Computer Theory and Engineering (ICACTE), vol. 2, pp. 1381-1388, Cairo,September, (2009).
5. S. L. Garfinkel, A. Juels, and R. Pappu, RFID privacy: an overview of problems and proposed solutions, IEEE Security and Privacy, vol. 3, no. 3, pp.34-43, (2005).
6. Read / Write Performance for low memory passive HF RFID tag-reader system “Journal of Theoretical and Applied Electronic Commerce Research” VOL 4 / ISSUE 3 / pp 1-16 DECEMBER (2009)
7. S.-M. Lee, Y. J. Hwang, D. H. Lee, and J. I. L. Lim, Efficient authentication for low-cost RFID systems, in ICCSA, (2005).
8. M. Ohkubo, K. Suzuki, and S. Kinoshita, Cryptographic approach to privacy-friendly tags, in RFID Privacy Workshop, (2003).
9. S. Prabhu, C. Qiu, B. Schmitt, C.-C. Chu, and R. Gadh, SpecimenTrak: an RFID system for tagging and tracking anatomical specimens, in The 25th Annual Scientific Session of the American Association of Clinical Anatomists, Toronto, (2008).
10. M. R. Rieback, B. Crispo, and A. S. Tanenbaum, The evolution of RFID security, IEEE Pervasive Computing, vol. 5, no. 1, pp. 62-69, (2006).