Metrics of a New Symmetrical Encryption Algorithm

G. Ramesh
Research and Development Centre,
Bharathiyar University, Coimbatore-641046,
Tamil Nadu, India
mgrameshmca@yahoo.com

R. Umarani
Department of Computer Science,
Sri Sarada college for women, Salem,
Tamil Nadu 636 011, India
umainweb@gmail.com

Abstract

The new symmetrical algorithm avoids the key exchange between users and reduces the time taken for the encryption, decryption, and authentication processes. It operates at a data rate higher than DES, 3DES, AES, UMARAM, RC2 and RC6 algorithms. It is applied on a text file and an image as an application. The encryption becomes more secure and high data rate than DES, 3DES, AES,UMARAM,RC2 and RC6. A comparison has been conducted for the encryption algorithms like DES, 3DES, AES, UMARAM, RC2 and RC6 at different settings for each algorithm such as different sizes of data blocks, different data types, battery power consumption, different key size and finally encryption/decryption speed. Experimental results are given to demonstrate the effectiveness of each algorithm.

Keywords: Plaintext; Encryption; Decryption; S-Box; Key updating; Outside attack; key generation for Proposed Algorithm.

1. Introduction

Wireless Local Area Network (WLAN) is one of the fastest growing technologies. It is found in the office buildings, colleges, universities, and in many other public areas [1]. The security in WLAN is based on cryptography, the science and art of transforming messages to make them secure and immune to attacks by authenticating the sender to receiver within the WLAN.

There are a lot of symmetric-encryption algorithms used in WLAN, such as DES [2][11], TDES [3], AES [4], CAST-256,RC6 [5][7] and UMARAM[6]. In all these algorithms, both sender and receiver have used the same key for encryption and decryption processes respectively. The attacks on the security of WLAN depend on viewing the function of the computer system in WLAN as providing information (such as company title, the data type can be transferred in WLAN, and the algorithms and He will act as an evil to analysis the data-exchange to eavesdrop or act as man-in-the middle. The proposed algorithm will avoid key-exchange, the time taken for authentication process, and it will avoid the foxes[14].
The hacking is the greatest problem in the wireless local area network (WLAN). Many algorithms like DES, 3DES, AES,CAST, UMARAM[20], RC2 and RC6 have been used to prevent the outside attacks to eavesdrop or prevent the data to be transferred to the end-user correctly. The authentication protocols have been used for authentication and key-exchange processes. A new symmetrical encryption algorithm is proposed in this paper to prevent the outside attacks to obtain any information from any data-exchange in Wireless Local Area Network(WLAN)[15][21].

authentication protocol used in WLAN). Each company sends its title with each message. The outside attacks can use this fixed plaintext, company-title, and encrypted text of that title to obtain the key used in WLAN. The outside attack can also appear as a fox because he/she can lie to use a computer on the WLAN to send an important message to someone because there are some troubles in his device while his device is still open to take a copy from the encrypted message. The plaintext and encrypted text are known. He can obtain the key used for encryption and decryption processes easily. The authentication protocols have been used for authentication and key-exchange processes, such as EAP-TLS [8][13], EAP-TTLS [9], and PEAP [10]. The attacker can be authorized-user and he/she will be accepted to access the network after the success of authentication and key exchange processes.

2. Proposed Symmetrical Algorithm

This new symmetrical encryption algorithm[22] was designed by G.Ramesh etal. in the year 2010. A block encryption algorithm is proposed in this approach. In this Algorithm[22], a series of transformations have been used depending on S-BOX, XOR Gate, and AND Gate. The proposed algorithm encrypts a plaintext of size 64-bits by a key size of 64-bits. It uses eight rounds for encryption or decryption process. It overcomes some drawbacks of the other algorithms. It is more efficient and useable for the Wireless Local Area Network because it avoids the using of the same key with other packets within a message. The algorithm is simple and helpful in avoiding the hackers. S-BOX generation is the backbone of this algorithm. It has eight columns and 256 rows; each element consists of 8-bits, see Appendix A for the contents of S-boxes. It replaces the input by another code to the output. The order of the columns is changed in each round as follows:

Round 1: C1C2C3C4C5C6C7C8
Round 2: C2C3C4C1C8C5C6C7
Round 3: C3C4C1C2C7C8C5C6
Round 4: C4C1C2C3C6C7C8C5
Round 5: C5C8C7C6C3C2C1C4
Round 6: C6C5C8C7C2C1C4C3
Round 7: C7C6C5C8C1C4C3C2
Round 8: C8C7C6C5C4C3C2C1

Figure (1) combines between keys generation and Data encryption. There are two external inputs for keys generation, Rni and Rv, where i is the round number, i=1, …, 8. Rv has two hexadecimal values, (00 00 00 00 00 00 00) and (FF FF FF FF FF FF FF FF). Rni has two hexadecimal values, (00 00 00 00 00 00 00 00) and the initial key value, 64-bits, used at the first time. The initial key, 64-bits, can be the same for all rounds or each round can have different initial key as the designer like.

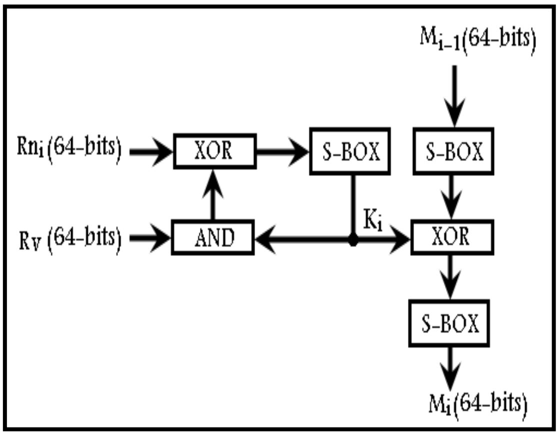

Fig .1 Proposed Algorithm for Encryption

Round-Key generation, at the first time, begins by using the first value of Rv, (00 00 00 00 00 00 00 00), to avoid any noise from the feedback of the S-BOX to the initial key. Rni equals, at the first time, to initial value of round-key. Then, the initial value of the key, 64-bits is divided into eight parts, 8-bits each. Each part will travel to a row, under the same column, having a number equals to its number plus one, it will find a code, 8-bits, that will be used instead of this part. For example, part 3=A2 , it will take the code in the column number 3,according to the columns-ordering of the S-BOX in each round, and the row number 162+1=163 and column number 3, where A2 in a hexadecimal format equals to 162 as a decimal value. The eight parts will be replaced by another eight parts, they will be used as a round-key to encrypt the message in each round, and also will feedback to update the round-key to another key by changing Rv to its second value, (FF FF FF FF FF FF FF FF), and also Rni to the other value, (00 00 00 00 00 00 00 00). So the algorithm will update its round-key by itself, and each round will choose its key randomly from 264 = 18,446,744,073,709,551,616 available keys. Thus, in each encryption process, a different key will be used for each round; it gives the impossibility to the hackers to decrypt the cipher text. The Rv initial value, (00 00 00 00 00 00 00 00), is used to make synchronization between the transmitter and receiver when there are troubles appeared in the decryption process, the receiver must send a message to the transmitter to request the reset of Rv value, in this case Rv= (00 00 00 00 00 00 00 00), and the Rni must equal to the initial key value. Otherwise, Rv=(FF FF FF FF FF FF FF FF) and Rni= (00 00 00 00 00 00 00 00).

In the data encryption, as round-key generation, the message block, 64-bits, is divided into eight parts to apply them to the eight columns of the S-BOX. The order of column depends on the round number. The output of the S-BOX will be XORed with round-key. The output of the XOR gate will be divided into to eight parts to apply to the S-BOX. The encrypted block will be the input of the next round, see figure (1), The Key generation of each round does not depend on the other round-key generation. The data decryption process is the same as the data encryption process but, the order of the round-key, K_i, used in the encryption process will be reversed to be used in the decryption process, and cipher text becomes instead of the plaintext to obtain the decrypted block as the same as the plaintext. The key will be updated by itself and the next packet will use different key. Each round will use different key because the order of columns of the S-BOX is interchanged. If there are NAK from the receiver, the sender will encrypt the packet by the initial key, default case, by applying Rni = Initial key, and Rv=, (00 00 00 00 00 00 00 00) to reset the system to the default case. If the outsider attack prevents any packet or message to reach the receiver, the next packet or message cannot be decrypted correctly because at this situation the key used for encryption is not the same as that used for decryption and these will be no synchronization between the sender and the receiver. The receiver will know that there is something wrong in the transmitted message because of virus, outside attacks, or environment noise to reach correctly. The receiver will send NAK to the sender. The NAK is a message of all 0-bits and the number of the damaged packet. The NAK length is 64-bits as the normal message. The NAK will be encrypted by the last updated-key, as the normal message will be encrypted, to avoid the traffic analysis from the outsider attacks.

This initial key is used only in three cases, the connection in the first time, NAK, and authentication process. In authentication process, the sender and the receiver will interchange a secret message encrypted by last updated key. If this message encrypted again, the encrypted message will have a different contents than the first one. The outside attack cannot find out the key even if he knows the title of the company because the encrypted title will take other form and the key- generation of each round does not depend on each others. The designer can use different initial keys for each round to make the system more secure[12].

The new symmetrical algorithm avoids the key exchange between users and reduces the time taken for the encryption, decryption, and authentication processes. It operates at a data rate higher than DES, 3DES, AES, UMARAM, RC2 and RC6 algorithms. It is applied on a text file and an image as an application. The encryption becomes more secure and high data rate than DES, 3DES, AES, RC2, UMARAM and RC6.

This paper examines a method for evaluating performance of selected symmetric encryption of various algorithms. Encryption algorithms consume a significant amount of computing resources such as CPU time, memory, and battery power. Battery power is subjected to the problem of energy consumption due to encryption algorithms. Battery technology is increasing at a slower rate than other technologies. This causes a "battery gap" [17, 18]. We need a way to make decisions about energy consumption and security to reduce the consumption of battery powered devices.

This study evaluates seven different encryption algorithms namely; AES, DES, 3DES, RC6,UMARAM and RC2. The performance measure of encryption schemes will be conducted in terms of energy, changing data types - such as text or document, Audio data and video data power consumption, changing packet size and changing key size for the above and proposed cryptographic algorithms.

This paper is organized as follows. Section 2 gives Proposed Symmetrical Algorithm, Section 3 gives experimental design for metric of new algorithm. Section 4 gives experimental results of the proposed algorithm. Conclusions are presented in section 5.

3. Experimental Design For Metric of Proposed System:

For our experiment, the data is collected from the laptop IV 2.4 GHz CPU. In the experiments, the laptop encrypts a different file size ranges from 321 K byte to 7.139Mega Byte139MegaBytes for text data, from 33 Kbytes to 8262 Kbytes for audio data, and from 4006 Kbytes to 5073 Kbytes for video files.

Several performance metrics are collected: (1) Encryption time, (2) CPU process time, (3) CPU clock cycles and battery power, (4) Throughput, (5) Different data types, and (6)Different size of data block.

The encryption time is considered the time that an encryption algorithm takes to produce a cipher text from a plaintext. Encryption time is used to calculate the throughput of an encryption scheme. It indicates the speed of encryption. The throughput of the encryption scheme is calculated as the total plaintext in bytes encrypted divided by the encryption time [19].

Throughput = Total plaintext encrypted in bytes / Encryption time (1)

The CPU process time is the time that a CPU is committed only to the particular process of calculations. It reflects the load of the CPU. The more CPU time is used in the encryption process, the higher is the load of the CPU. The CPU clock cycles are a metric, reflecting the energy consumption of the CPU while operating on encryption operations. Each cycle of CPU will consume a small amount of energy.

The following tasks that will be performed are shown as below:

- A comparison is conducted between the results of the selected different encryption and decryption schemes in terms of the encryption time at two different encoding bases namely; hexadecimal base encoding and in base 64 encoding.
- A study is performed on the effect of changing packet size at power consumption during throughput for each selected cryptography algorithm.
- A study is performed on the effect of changing data types - such as text or document, audio file, and video file for each cryptography selected algorithm on power consumption.
- A study is performed on the effect of changing key size for cryptography selected algorithm on power consumption[16].

4. Experimental Results

4.1 Different Output Results of Encryption (Base 64, Hexadecimal)

Experimental results are given in Figures 2 and 3 for the selected seven encryption algorithms at different encoding method. Figure 2 shows the results at base 64 encoding while Figure 3 gives the results of hexadecimal base encoding. We can notice that there is no significant difference at both encoding method. The same files are encrypted by two methods; we can recognize that the two curves almost give the same results.

4.2 Effect of Changing Packet Size for Cryptographic Algorithms on Power Consumption

4.2.1 Encryption of Different Packet Size

Encryption time is used to calculate the throughput of an encryption scheme. The throughput of the encryption scheme is calculated by dividing the total plaintext in Megabytes encrypted on the total encryption time for each algorithm. As the throughput value is increased, the power consumption of this encryption technique is decreased

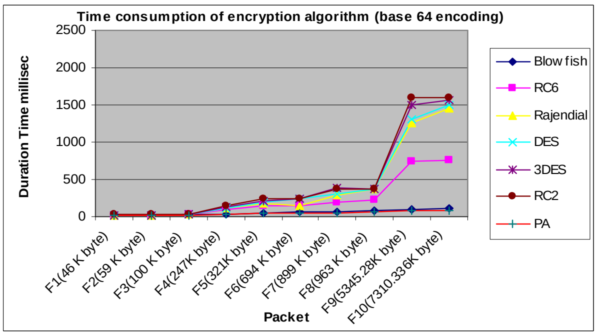

Fig. 2 Time consumption of encryption algorithm

Experimental results for this compassion point are shown Fig. 4 at encryption stage. The results show the superiority of Proposed algorithm over other algorithms in terms of the processing time. Another point can be noticed here; that RC6 requires less time than all algorithms except Proposed Algorithm. A third point can be noticed here; that AES has an advantage over other 3DES, DES and RC2 in terms of time consumption and throughput. A fourth point can be noticed here; that 3DES has low performance in terms of power consumption and throughput when compared with DES. It always requires more time than DES because of its triple phase encryption characteristics. Finally, it is found that RC2 has low performance and low throughput when compared with other six algorithms in spite of the small key size used.

4.2.2 Decryption of Different Packet Size

Experimental results for this compassion point are shown Figure 5 decryption stage. We can find decryption that Proposed Algorithm is better than other algorithms in throughput and power consumption. The second point should be noticed here that RC6 requires less time than all algorithms except Proposed Algorithm. A third point that can be noticed that AES has an advantage over other 3DES, DES, RC2.The fourth point that can be considered is that RC2 still has low performance of these algorithm. Finally, Triple DES (3DES) still requires more time than DES.

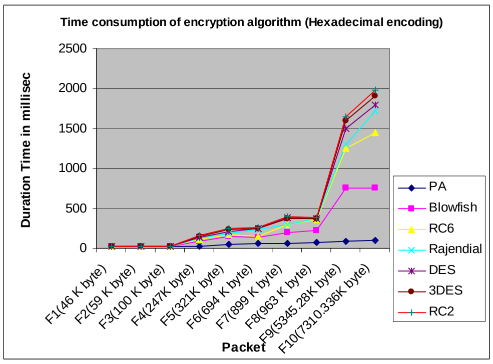

Fig. 3 Time consumption of encryption algorithm (Hexadecimal encoding)

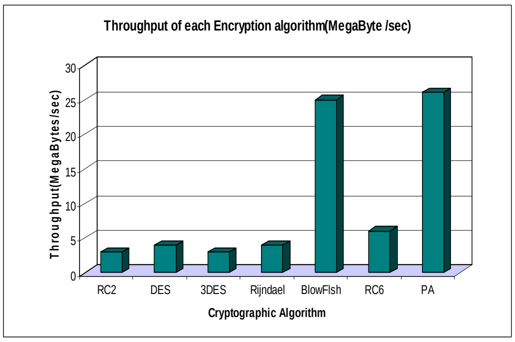

Fig. 4 Throughput of each encryption algorithm

4.3 The Effect of Changing File Type (Audio Files) for Cryptography Algorithm on Power Consumption

4.3.1 Encryption of Different Audio Files (Different Sizes) Encryption Throughput

In the previous section, the comparison between encryption algorithm has been conducted in text and document data files. Now we will make a comparison between other types of data (Audio file) to check, which one can perform better in this case. Experimental results for audio data types are shown in the Figure 6.

CPU Work Load

In Fig. 7, we show the performance of cryptographic algorithms in terms of sharing the CPU load. With a different audio block size Results show the superiority of Proposed algorithm over other algorithms in terms of the processing time (CPU work load) and throughput. Another point can be noticed here; that RC6 requires less time than all algorithms except Proposed Algorithm. A third point can be noticed here; that AES has an advantage over other 3DES, DES and RC2 in terms of time consumption and throughput especially in small size file.

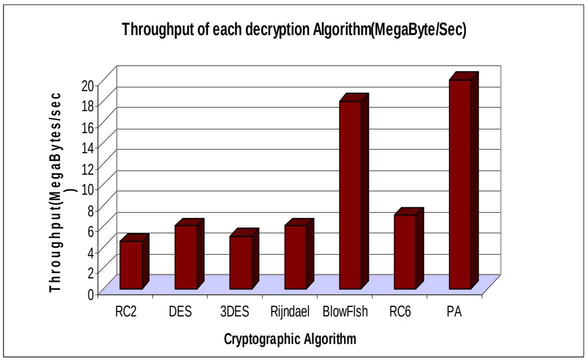

Fig. 5 Throughput of each decryption algorithm (Megabyte/Sec)

A fourth point can be noticed here; that 3DES has low performance in terms of power consumption and throughput when compared with DES. It always requires more time than DES. Finally, it is found that RC2 has low performance and low throughput when compared with other seven algorithms in spite of the small key size used.

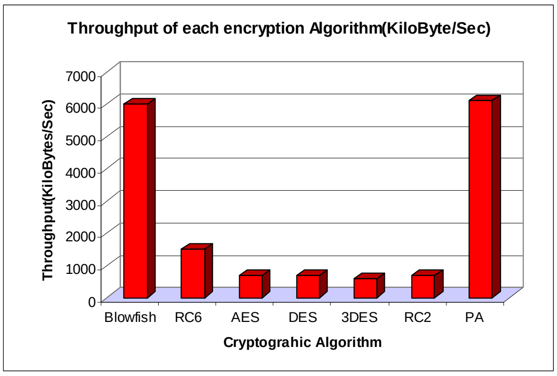

Fig. 6 Throughput of each encryption algorithm (Kilo-bytes/Second)

Decryption of Different Audio files (Different Sizes)

Decryption Throughput Experimental results for this compassion point are shown Figure 8.

CPU Work Load

Experimental results for this compassion point are shown Figure 9.From the results we found the result as the same as in encryption process for audio files.

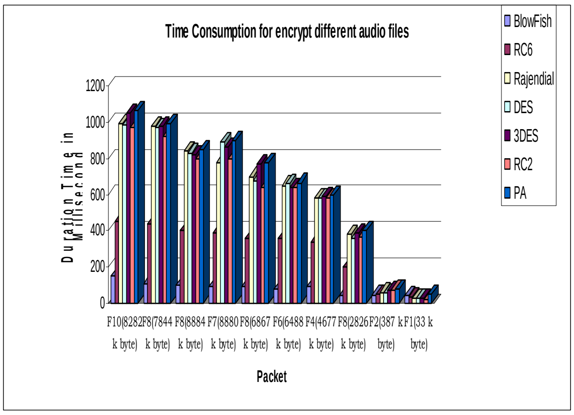

Fig. 7 Time consumption for encrypt different audio files

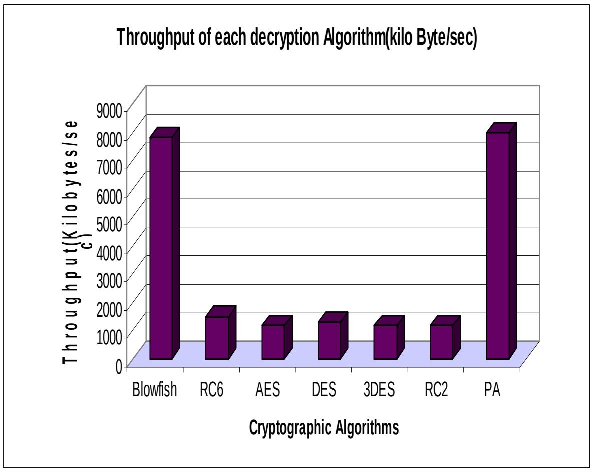

Fig. 8 Throughput of each Decryption algorithm (Kilobytes / Second)

4.4 The Effect of Changing File Type (Video Files) for Cryptography Algorithm on Power Consumption

4.4.1 Encryption of different video files (different sizes)

Encryption Throughput

Now we will make a comparison between other types of data (video files) to check which one can perform better in this case. Experimental results for video data type are shown in the Figure 10.

CPU Work Load

In Figure 11, we show the performance of cryptography algorithms in terms of sharing the CPU load with a different audio block size.

The results show the superiority of Proposed algorithm over other algorithms in terms of the processing time and throughput as the same as in Audio files. Another point can be noticed here; that RC6 still requires less time has throughput greater than all algorithms except Proposed Algorithm. A third point can be noticed here; that 3DES has low performance in terms of power consumption and throughput when compared with DES. It always requires more time than DES. Finally, it is found that RC2 has low performance and low throughput when compared with other six algorithms.

4.4.2 Decryption of Different Video Files (Different Sizes)

Decryption Throughput

Experimental results for this compassion point are shown Fig. 12.

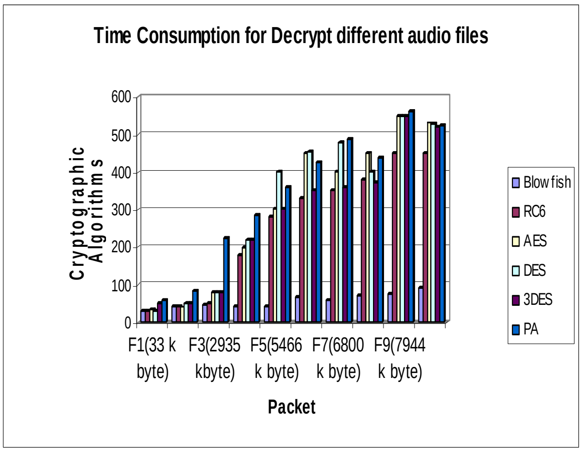

Fig. 9 Time consumption for decrypt different audio files

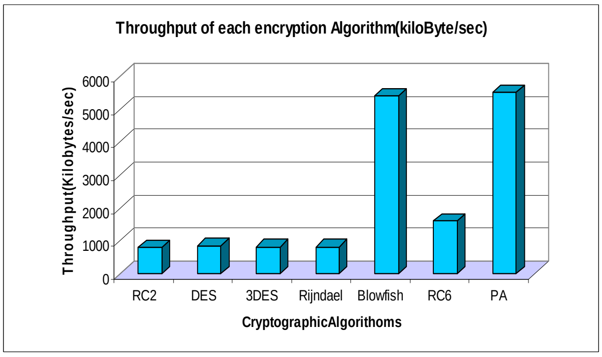

Fig. 10 Throughput of each encryption algorithm (Kilobytes/sec)

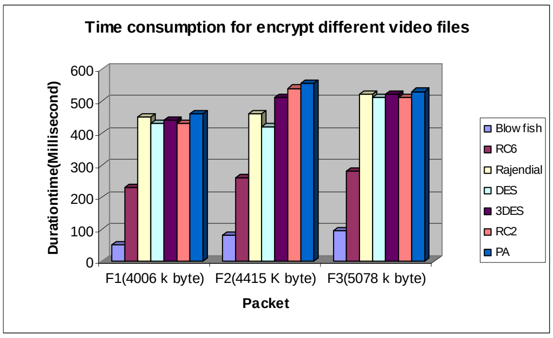

Fig. 11 Time consumption for encrypt different video files

CPU Work Load

Experimental results for this compassion point are shown Fig. 13. From the results we found the result as the same as in encryption process for video and audio files.

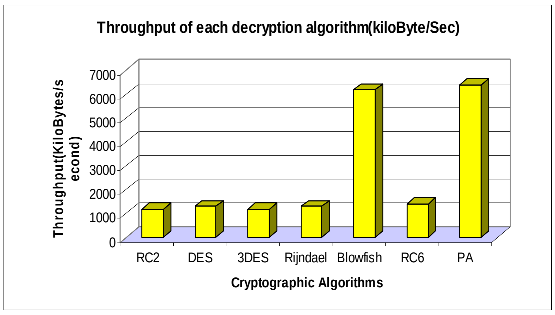

Fig. 12 Throughput of each decryption algorithm

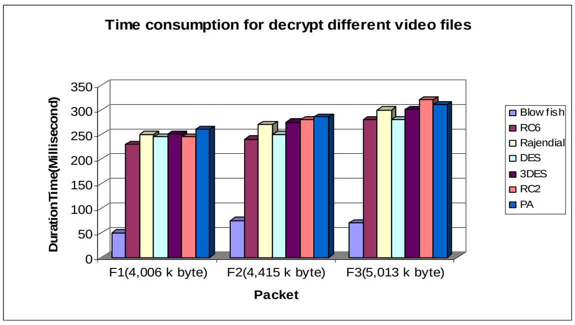

Fig. 13 Time consumption for decrypt different video files

4.5 The Effect of Changing Key Size of AES, And RC6 on Power Consumption

The last performance comparison point is changing different key sizes for AES and RC6 algorithm. In case of AES, we consider the three different key sizes possible i.e., 128-bit, 192-bit and 256-bit keys. The Experimental results are shown in Fig.14 and 15.

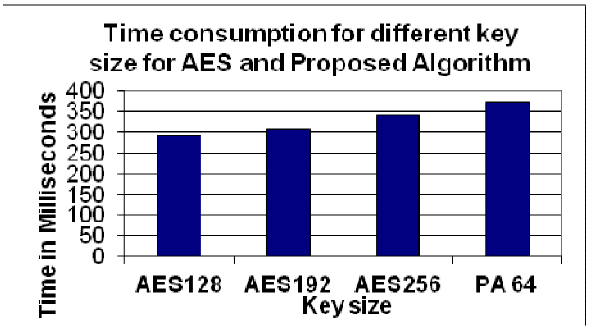

Fig. 14 Time consumption for different key size for AES and PA

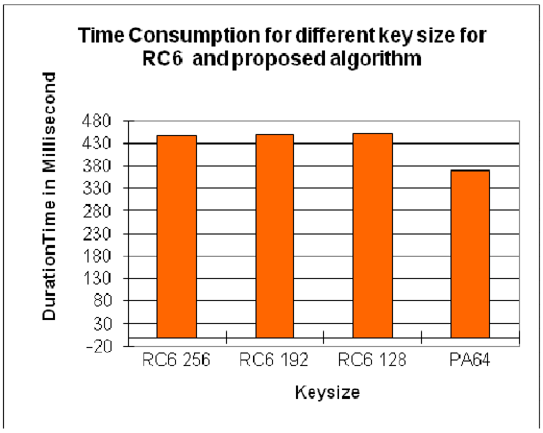

Fig. 15 Time consumption for different key size for RC6 and PA

In case of AES it can be seen that higher key size leads to clear change in the battery and time consumption. It can be seen that going from 128-bit key to 192-bit causes increase in power and time consumption about 8% and to 256-bit key causes an increase of 16% [12]. Also in case of RC6, we consider the three different key sizes possible i.e., 128-bit, 192-bit and 256-bit keys. In case of RC6 it can be seen that higher key size leads to clear change in the battery and time consumption.

5. Conclusion

The selected algorithms are AES, DES, 3DES, RC6, RC2,UMARAM and Proposed Algorithm were tested .Several points can be concluded from the Experimental results. Firstly; there is no significant difference when the results are displayed either in hexadecimal base encoding or in base 64 encoding. Secondly; in the case of changing packet size, it was concluded that proposed Algorithm has better performance than other common encryption algorithms used, followed by RC6. Thirdly; we find that 3DES still has low performance compared to algorithm

DES. Fourthly; wend RC2, has disadvantage over all other algorithms in terms of time consumption. Fifthly; we find AES has better performance than RC2, DES, and 3DES. In the case of audio and video files we found the result as the same as in text and document. Finally in the case of changing key size - it can be seen that higher key size leads to clear change in the battery and time consumption.

References

[1] William Stallings, "Network Security Essentials (Applications and Standards)", Pearson Education, 2004.

[2] National Bureau of Standards, "Data Encryption Standard", FIPS Publication 46, 1977.

[3] Jose J. Amador, Robert W.Green, "Symmetric-Key Block Ciphers for Image and Text Cryptography", International Journal of Imaging System Technology, 2005. Vol3. Pp.31-36

[4] Daemen, J., and Rijmen, V. Rijndael, "The Advanced Encryption Standard", Dr. Dobb's Journal, March 2001.
Vol.26, No.3, March 2001, pp.137—139

[5] Adams,C. " Constructing Symmetric Ciphers Using the CAST Design." , Design, Codes, and Cryptography, 1997. Pp. 2-15 May 1997

[6] Ramesh G, Umarani. R, " Data Security In Local Area Network Based On Fast Encryption Algorithm", International Journal of Computing Communication and Information System(JCCIS) Journal ,Page 85-90. 2010.

[7] S. Contini, R.L. Rivest, M.J.B. Robshaw and Y.L. Yin. "The Security of the RC6 Block Cipher. Version 1.0 ". August 20, 1998. Pp.8-16 Vol2.No.3

[8] Simon, D., Aboba, B., and R. Hurst, "The EAP-TLS Authentication Protocol",RFC 5216, March 2008. Pp.1-32.

[9] P. Funk and S. Blake-Wilson, "EAP Tunneled TLS Authentication Protocol Version 1 (EAP-TTLSv1)", The Internet Society, Mar. 2006. draft-funk-eap-ttls-v1-01.txt

[10] Palekar, A., Simon, D., Zorn, G., Salowey, J., Zhou, H., and S. Josefsson, "Protected EAP Protocol (PEAP) Version 2", work in progress, October 2004. Pp.28-33,Vol 4.

[11] ANSI3.106, "American National Standard for Information Systems—Data Encryption Algorithm—Modes of Operation", American National Standards Institute, 1983. Pp.3-5.

[12] Bruce Schneider, John Wiley & Sons, Inc., "Applied Cryptography, Second Edition", New York, NY, 1996. Pp.84-98

[13] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, "Extensible Authentication Protocol (EAP)", RFC 3748, June 2004. Pp. 21-40

[14] Simpson, W., "The Point-to-Point Protocol (PPP)", STD 51, RFC 1661, July 1994. Pp.1-16.

[15] Institute of Electrical and Electronics Engineers, "Local and Metropolitan Area Networks: Overview and Architecture", IEEE Standard 802, 1990. December 31, 1990 pp.1-31

[16] Aamer Nadeem, Dr M. Younus Javed, " A Performance Comparison of Data Encryption Algorithms ", IEEE International Conference on Networking, 2009. Pp.84-89.

[17] R. Chandramouli, "Battery power-aware encryption", ACM Transactions on Information and System Security (TISSEC), vol. 9, no. 2, pp. 162-180,May 2006.

[18] K. McKay, "Trade-o®s between Energy and Security in Wireless Networks Thesis", Worcester Polytechnic Institute, Apr. 2005. Pp.12-13.

[19] A. A. Tamimi, "Performance Analysis of Data Encryption Algorithms", Oct. 1, 2008. Pp. 20-27, Vol2.

[20] G. Ramesh and R. Umarani , "UMARAM: A novel fast encryption algorithm for data security in local area network", pp.758–768. 2010

[21] G. Ramesh and R. Umarani "Data Security in Local Area Network Based on Fast Encryption Algorithm", Communications in Computer and Information Science, Vol. 89, Part 1, pp.11-26, 2010.

[22] G. Ramesh and R. Umarani, " A Novel Symmetrical Encryption Algorithm with High Security Based on Key Updating", pp.57-69. 2010

G. Ramesh is working as Scholar in Research and development Centre, Bharathiyar University, Coimbatore,India. He has 11 years of experience in both Industrial and academic fields. He has published 14 Papers in International and national journals and 23 papers in national and international conferences. His area of Interest includes information security and Wireless Networks.

R. Umarani is working as Associate Professor in the Department of Computer Science, Sri Sarada College for Women, Salem. She has 22 years of experience in both industrial and academic fields. She has completed Ph.D in Computer Science from Periyar University, Salem. She has published 37 Papers in international and National Journals and 56 papers in National and International Conferences. Her area of interest includes data mining, information security, wireless sensor networks and neural networks.