Rajshahi University Journal of Science & Engineering **ISSN 2309-0952**
Vol. 44:131-139, 2016

Comparison of Encryption Algorithms for Multimedia

Md. Martuza Ahamad* and Md. Ibrahim Abdullah

Department of Computer Science and Engineering, Islamic University,
Kushtia, Bangladesh.

*Corresponding author: *martuza.cse@gmail.com*

Abstract

Cryptographic techniques play crucial role when users exchange information. Multimedia plays an important role in learning and sharing experiences. When multimedia contents are shared among the users, it faces security threats. Usually multimedia contents takes much space. Encryption technique should be time efficient. In this work we consider four encryption techniques: Blowfish, AES, XOR and RSA and four types of media content: text, image, audio and video. Simulation shows that AES is time efficient than others. Comparing between symmetric and asymmetric cryptography, symmetric cryptographic techniques take less time than asymmetric technique.

Keywords: Cryptography; Security; RSA; Blowfish; AES; Rijndael; XOR.

INTRODUCTION

During recent years the telecommunication industry has made tremendous progress in their development of systems that offer more bandwidth to the end user. User share their experiences, learning and observation with their community using computer network. Shared information usually mixed of various media such as text, images, audio or video. When these media contents transmit in computer network, it faces many security threats. For example, the board of directors of a company discuss about their future marketing policies through online. An adversary heard this information and shares it with a competitor company. Arm force gets the enemy location from control room. But an adversary intercepts the transmission and alters the location. So it is necessary to protect the information from unauthorized user. The solution is encryption. There are several encryption techniques. Choice of encryption technique depends on strength of the mathematics behind the algorithm, encryption time and strength against the attack on that algorithm. Encryption techniques can be classified as Symmetric key algorithm and Asymmetric key algorithm. Symmetric-key algorithms also known as single-key and asymmetric algorithm uses two different keys Private key and a Public key to execute encryption /decryption process [1] [3].

The main limitation of cryptographic systems arises when attackers try to attack using brute force approach. For example if one use a symmetric key algorithm, so it has only one key and if attacker can guess that key using brute force approach [2] [3] then he/she can find out the original message from cipher message using that key. So a secure algorithm is that, which can construct and use a key which is hard to guess. Some algorithms are mathematically hard to implements i.e. RSA, Blowfish, AES and some of are easy to implements i.e. XOR. In RSA the key generation process is too complicated in mathematically when it was developed and hard to implement when it is use [4]. In AES implementation have a lot of repetition of cycles and having a lot of steps such as- Key Expansions for constructing key using Rijndael's key schedule, Initial Round, Rounds which have Sub Bytes, Shift Rows, Mix Columns and Add Rounds, and Final Rounds.

For each steps most common operation is XOR but the combination of all steps are not too easy to implement [5] [6]. In Blowfish algorithm also have a lot of steps such as-Key-Expansion and Data Encryption. Both of steps have some sub steps [6] [7].

Usually multimedia contents take much space [19] [20]. When these large files are encrypted it takes much time. Our objective is to find the suitable encryption technique that is less time complex. In this work we consider four most used encryption algorithms: Blowfish, AES, XOR and RSA. First three algorithms are symmetric and RSA is asymmetric. For encryption and decryption we use four types of multimedia contents: text, image, audio and video.

The rest of the paper organized as follows: section II describe about the used encryption techniques. In section III we discuss the related work. In section IV, we describe the simulation in detail and finally conclude in section V.

CRYPTOGRAPHIC ALGORITHMS

This section provides information about the cryptographic algorithms used in this work. There are two general categories of cryptographic keys: symmetric key and asymmetric key systems. The symmetric key systems use a single key. The single key is used both to encrypt and decrypt the information. Both sides of the transmission need to keep the key in secret from. The security of the transmission will depend on how well the key is protected. The biggest difficulty with this approach, of course, is the distribution of the key. Secret key cryptography schemes are generally categorized as being either stream ciphers or block ciphers. Most popular symmetric key algorithms are DES, Triple DES, AES, IDEA, TEA, Blowfish etc. [1] [2].

Asymmetric or public key cryptosystem uses two different keys to encrypt and decrypt. Both keys are mathematically related. If A encrypts a message with his private key then B, the recipient of the message can decrypt it with A's public key. Similarly anyone who knows A's public key can send him a message by encrypting it with his public key. A will then decrypt it with his private key. Public key cryptography was developed in 1977 by Rivest, Shamir and Adleman (RSA) in the US. This kind of cryptography is more efficient than the symmetric key cryptography because each user has only one distinct key to encrypt and decrypt all the messages that he or she sends or receives [2]. Some of examples for asymmetric key cryptosystem are RSA, ELGAMAL, and ECC etc. Security of asymmetric algorithms depend on key length, key generation techniques. Asymmetric key encryption is slow compared to symmetric encryption [2] [3].

Each algorithm has own advantages and limitations. Since multimedia contents take much storage than others. We consider four algorithms AES, Blowfish, XOR and RSA [8] suitable for large file.

AES: AES stands for Advanced Encryption Standard also known as “Rijndael”. It is symmetric block cipher algorithm. In 2001 two Belgian cryptographers “Joan Daemen” and “Vincent Rijmen” first develop this algorithm at National Institute of Standards and Technology (NIST). It supports 128 bits fixed length block size and variable length key size of 128, 192 and 256 bits [9] [20].

AES is based on a design principle known as substitution-permutation network. And AES operates on a 4x4 column-major order matrix of byte. The key size used for an AES cipher specifies the number of repetitions of transformation rounds. For 128-bit keys needed 10 cycles of repetitions, for 192-bit keys needed 12 cycles of repetitions and for 256-bit keys needed 14 cycles of repetitions. Each round has some specific operations like, SubBytes, ShiftRows, MixColumns and AddRoundKey [1].

AES use four types of transformations: substitution, permutation, mixing and key-adding. In substitution, mathematical calculation is use for transforming each byte individually and only one table is use for this; it has two different processes. First is SubBytes, this transformation is use in encryption site and transformation operation involves 16 independent byte to byte transformations and another is InvSubBytes, which is use in decryption site, operation is opposite of SubBytes. In permutation, permutes the bytes; operations are:- ShiftRows- In here rows are shifted such as Row 0: no shift, Row 1: 1-byte shift, Row 2: 2-bytes shifts and so on; is used in encryption site. The opposite of ShiftRow is InvShiftRow, which is used in decryption site. Mixing: In here, changes the bytes and create four byte at a time taking four byte input; MixColumns and InvMixColumns are two different operations of Mixing, both are use in column level mixing, first one is use in encryption site and another one is use in decryption site. The last transformation is key-adding; probably it is the most important one. In here, operation AddRoundKey adds a round key word with each state column matrix. [1] [2].

Another important operation of AES is Key-Expansion. We know AES use three different key size 128-bits, 192-bits and 256-bits. AES use different method to construct different size of key [1].

Blowfish: Blowfish is one of most used encryption algorithm. In 1993 “Bruce Schneier” designed this algorithm. It is symmetric block cipher algorithm takes variable length key from 32 bits to 448 bits. It can encrypt block data of 64-bits at a time. It was the nice alternatives of DES or IDEA [3] [12].

Blowfish use a large number of subkeys, and these keys must be constructed before encryption and decryption. It uses 18 32-bit P-arrays:

$P_1, P_2, \ldots\ldots\ \ldots.., P_{18}$

And four 32-bits S-boxes have 256 entries each:

$S_{1,0}, S_{1,1}, \ldots\ldots\ldots, S_{1,255}$
$S_{2,0}, S_{2,1}, \ldots\ldots\ldots, S_{2,255}$
$S_{3,0}, S_{3,1}, \ldots\ldots\ldots, S_{3,255}$
$S_{4,0}, S_{4,1}, \ldots\ldots\ldots, S_{4,255}$

Blowfish is a Feistel network consisting of 16 round. It can take 64-bit data as input at a time. It uses a Feistel function F is as follows: Divide x_L into four eight-bit quarters: [3]

a,b,c, and d $F(x_L) = XOR\ ((S_{1,a} + S_{2,b} \bmod 2^{32}), S_{3,c}) + S_{4,d} \bmod 2^{32}$

Subkeys are calculated as: firstly initialize the P-arrays then four S-boxes using hexadecimal digits of P. XOR P_1 with first 32-bits of key then XOR P_2 with second 32-bits of key, this process continue up to P_{18}. Encrypt all-zero string using previous two steps and replace P_1 and P_2 with the output of this step. Encrypt output of the previous step; replace P_3 and P_4 with the output of this step. Continue this process for replacing all the contents of P-arrays and then all four S-boxes in order, with the output of the continuous output of Blowfish algorithm [3].

XOR: XOR is simply bitwise exclusive OR operation. Where a key stream XORed with a plain text stream. It is symmetric variable key length stream cipher algorithm. In XOR operation, if the two bits are same then output will be true otherwise output will be false. It is also known as modulus 2 addition. Same operations are held on encryption and decryption [10]. Operations of XOR: [1] [10]

Encryption: c = XOR (m, key);
Decryption: m = XOR (c, key);
(where m is plain text and c is cipher text and XOR (A, 0) = A; XOR (A, A) = 0)

RSA: RSA is one of the widely used, secure and applicable algorithm. In 1977 three mathematician "Ron Rivest", "Adi Shamir" and "Leonard Adleman" publicly describe the algorithm. It is first practical asymmetric public key algorithm where two key, encryption key and decryption key are exists, the encryption key is public and decryption key is secret which is differ from first one [2] [3] [23].

Operations of RSA:
Choose two prime number p and q. p and q should be large, chosen randomly and need to keep secret for better security.

$$n = pq$$

Now randomly choose the encryption key, e, such that e and (p-1)(q-1) are relatively prime. Finally use the extended Euclidian algorithm to compute the decryption key, d, that

$$ed \equiv 1 \bmod (p-1)(q-1)$$

In other words, $d = e^{-1} \bmod ((p-1)(q-1)$

Here, d and n are relatively prime. The number e and n (e, n) is use as public key and d (d, n) is the private key.

Encryption formula is:

$$c_i = m_i^e \bmod n$$

And decryption formula is:

$$m_i = c_i^d \bmod n$$

Since $c_i^d = (m_i^e)^d = m_i^{ed} = m_i^{k(p-1)(q-1)+1} = m_i m_i^{k(p-1)(q-1)} = m_i*1 = m_i$; all (mod n)

The formula recovers the message. Where m is the plain text and c is the cipher text.

PREVIOUS WORKS

In early most of research work carried on comparison between symmetric algorithms or asymmetric algorithms or both [10-17]. In [10], DES, 3DES, AES and Blowfish algorithms are compared on the basis of rounds block size, key size and encryption/decryption time and it has shown that, Blowfish is better than other algorithm. In [11], image files are encrypted using different key length and compare them. In [13], five algorithms: Twofish, Blowfish, IB_mRSA, RSA and RC are compared with the parameter of rounds block size, key size and encryption/decryption time and shown that IB_mRSA is the better than other algorithm. In [14] "Comparative Implementation of Cryptographic Algorithms on ARM Platform", the authors consider two algorithms AES and Blowfish and compare basis of ARM implementation and shown that Blowfish is better than AES. In the work [18] to [21], different encryption techniques are implemented for video files. Image files are encrypted and decrypted in [21-22]. In [24], authors use a novel method chaos to encrypt audio (sound) files on mobile phone.

EXPERIMENTAL ANALYSIS

In our work, we develop a simulator using java programming language where all of implementations of the algorithms for text, image, audio and video files are takes place. Using that simulator we measure the execution (process) time of the program for various input size, then analyze the performance of the algorithms. Our PC setup was: HP 4th Gen. Probook 450, Intel ® Core™ i5-4200M CPU @ 2.50 GHz, 4 GB RAM with Ubuntu 16.04 LTS (Xenial Xerus) 64-bit Operating System. For our experiment we use four types of multimedia contents text, image, audio and video. In simulation, 20 samples of text data are considered. File size of text data varies from 1KB to 1MB. We take 14 BMP images vary from 10 KB to 2.2 MB. There are 10 samples of mp3 audio files from 100KB to 4.8MB and 10 video samples of 100 KB to 7 MB. In our simulation we use 128-bits key size for all of algorithms.

Table 1: Encryption times for different text files

Time (ms)	1 KB	10 KB	100 KB	200 KB	400 KB	600 KB	700 KB	800 KB	900 KB	1024 KB
Blowfish	1	1	1	2	5	8	9	11	17	21
AES	0	0	1	1	2	2	3	3	3	3
XOR	0	1	2	4	6	143	229	356	358	489
RSA	181	1743	17177	36814	73216	106800	126017	143206	162905	181248

Table 2: Decryption times for different text files

Time (ms)	1 KB	10 KB	100 KB	200 KB	400 KB	600 KB	700 KB	800 KB	900 KB	1024 KB
Blowfish	1	1	2	3	6	14	14	16	19	22
AES	0	1	1	1	1	2	2	2	2	2
XOR	1	1	3	5	7	9	14	14	19	24
RSA	668	6573	65841	135291	267200	396102	469120	534410	601221	681984

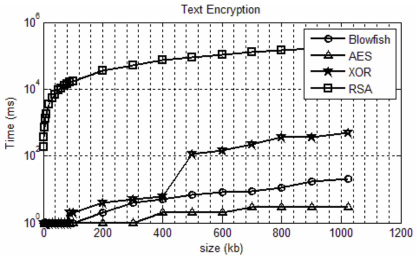

Fig. 1: File size vs encryption time for text

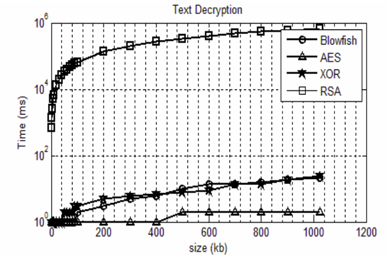

Fig. 2: File size vs decryption time for text

Table -1 and table -2 show the encryption and decryption time for different size of text files. Fig.-1 and fig.-2 illustrate the corresponding curves for the encryption and decryption time. In encryption graph (fig.-1), we saw that RSA curve consume time then other encryption techniques. XOR curve is as similar as others for small amount of load but when load is high then it take much time to encrypt data. AES and XOR curves show linearity when data load increases. In decryption graph (fig.-2), it is found that, like encryption RSA takes highest time to decrypt data. Blowfish and XOR algorithms take same amount of time to decrypt text data. AES algorithm takes less time than others. Symmetric algorithm AES shows best performance than other algorithms.

Table 3: Encryption times for different image files

Time (ms)	10 KB	100 KB	200 KB	400 KB	800 KB	1000 KB	1200 KB	1400 KB	1600 KB	1800 KB	2200 KB
Blowfish	1	2	3	6	11	15	16	19	22	24	33
AES	1	1	1	1	1	2	2	2	2	2	3
XOR	2	17	29	58	122	152	188	210	242	274	360
RSA	7	82	204	591	1910	2806	3907	5270	6946	8880	14345

Table 4: Decryption times for different image files

Time (ms)	10 KB	100 KB	200 KB	400 KB	800 KB	1000 KB	1200 KB	1400 KB	1600 KB	1800 KB	2200 KB
Blowfish	1	2	3	6	12	15	18	22	25	28	35
AES	1	1	1	1	2	2	3	3	3	4	6
XOR	2	17	29	58	122	152	188	210	242	274	360
RSA	117	1218	2443	5174	10542	14127	16922	20008	24023	27234	36560

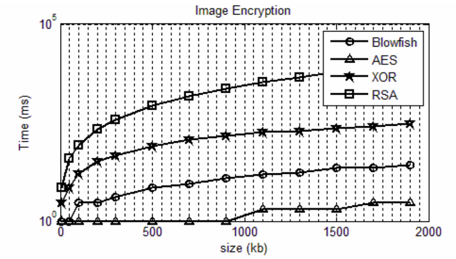

Fig. 3: File size vs encryption time for image

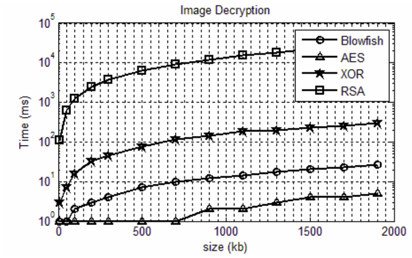

Fig. 4: File size vs decryption time for image

Table-3 and table-4 show the encryption and decryption time for different size of image files. Fig.-3 and fig.-4 represent the corresponding curves. From fig.-3 and fig.-4 it is found that the asymmetric algorithm RSA takes highest time to encrypt and decrypt images. XOR technique takes less time than RSA. Blowfish takes intermediary time between XOR and AES. Encryption and decryption time almost linearly increases with image file size increases. Among these algorithms, AES takes less time to encrypt and decrypt image files. In all cases decryption processes takes more time than encryption.

Table 5: Encryption times for different audio files

Time (ms)	100 KB	200 KB	400 KB	800 KB	1600 KB	2000 KB	2400 KB	3200 KB	4000 KB	4800 KB
Blowfish	3	9	16	26	34	64	72	84	88	90
AES	1	1	2	3	3	4	9	12	15	16
XOR	1	2	3	5	6	9	12	15	16	18
RSA	84	195	498	1629	6707	11070	16787	31972	52236	88469

Table 6: Decryption times for different audio files

Time (ms)	100 KB	200 KB	400 KB	800 KB	1600 KB	2000 KB	2400 KB	3200 KB	4000 KB	4800 KB
Blowfish	2	5	8	16	26	33	38	51	66	81
AES	1	2	3	4	4	5	8	10	11	12
XOR	1	2	3	5	6	9	12	15	16	18
RSA	1165	2294	4255	8968	20471	27712	35512	55397	80718	106491

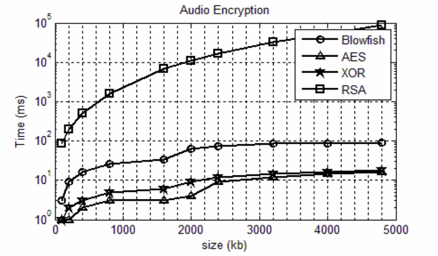

Fig. 5: File size vs encryption time for audio

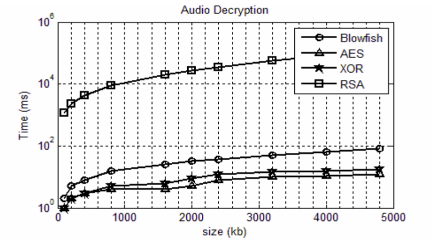

Fig. 6: File size vs decryption time for audio

Table -5 and table -6 show the encryption and decryption time for different size of audio files. Fig.-5 and fig.-6 illustrate the corresponding curves for the encryption and decryption time. From fig.-5 it is observe that RSA takes considerably much time to encrypt audio data than other algorithms. Blowfish encryption time is slightly high than XOR and AES. XOR and AES encryption time are almost same. Decryption times (fig.-6) are very close for symmetric algorithms Blowfish, XOR and AES. Asymmetric algorithm RSA has highest decryption time. In our study it is found that AES algorithm take less time than others.

Table 7: Encryption times for different video files

Time (ms)	100 KB	300 KB	500 KB	1000 KB	2000 KB	3000 KB	4000 KB	5000 KB	6000 KB	7000 KB
Blowfish	3	5	9	18	39	56	78	94	108	144
AES	1	2	2	3	5	5	7	8	14	19
XOR	1	2	2	4	7	13	14	20	22	26
RSA	83	335	789	2527	11653	39703	54328	88396	129695	187856

Table 8: Decryption times for different video files

Time (ms)	100 KB	300 KB	500 KB	1000 KB	2000 KB	3000 KB	4000 KB	5000 KB	6000 KB	7000 KB
Blowfish	2	5	8	15	33	51	66	84	100	114
AES	1	3	3	3	5	7	11	14	21	24
XOR	1	2	2	4	7	13	14	20	22	26
RSA	1144	3493	5904	12137	30451	54467	84162	120830	165364	212529

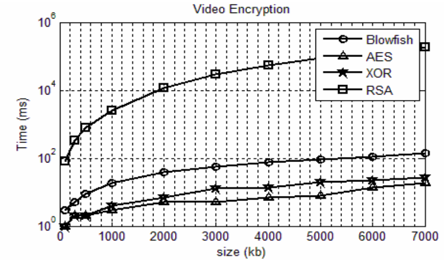

Fig. 7: File size vs encryption time for video

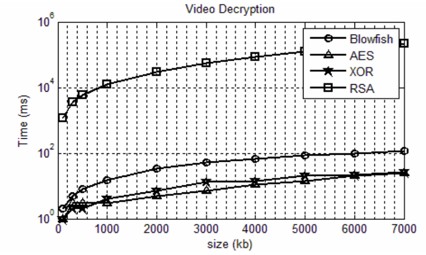

Fig. 8: File size vs decryption time for video

Table-7 and table-8 represent the encryption and decryption time for different size of video files. Fig.-7 illustrates the encryption time of table-7. RSA takes much time than other algorithms. Encryption time of Blowfish, XOR and AES are close enough. Fig.-8 presents the corresponding curves of table-8. Decryption curve (fig.-8) looks similar to encryption curve (fig.-7) for video data. In all cases AES has best performance to encrypt and decrypt video data.

CONCLUSIONS

In our study, it is found that, AES is the best performed algorithm than other algorithms. Blowfish and XOR has average rate of performance. Blowfish is the second best performed algorithm for multimedia contents text and image. The XOR algorithm shows good performance for audio and video. Asymmetric algorithm RSA is the most time consuming algorithm for encryption and decryption. In our study Symmetric key algorithms take less time than asymmetric key. We can suggest that, for multimedia content transmission symmetric key algorithms should be use.

References

[1] Forouzan A.B., “Cryptography and Network Security”, 2nd Edition, Tata McGraw Hill, 2012.

[2] Stallings W., “Cryptography and Network Security: Principles and Practice”, 6th Edition, Pearson, 2013.

[3] Schneier B., “Applied Cryptography”, 2nd Edition, John Wiley & Sons, 1996.

[4] Barret P., “Implementing the Rivest Shamir and Adleman Public Key Encryption Algorithm on a Standard Digital Signal Processor”, Advances in Cryptology, Springer, vol. 263, pp. 311-323, August 1986.

[5] Akkar M. L., Giraud C., “An Implementation of DES and AES, Secure against Some Attacks”, Cryptographic Hardware and Embedded System, Springer, vol. 2162, pp. 309-318, 2001.

[6] Manral V., “Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH)”, April 2007, Available from URL: https://tools.ietf.org/html/rfc4835.

[7] Meyers R. K., Desoky H. A., “An Implementation of the Blowfish Cryptosystem,” IEEE International Symposium on Signal Processing and Information Technology, pp. 346-351, Sarajevo, Dec, 2008.

[8] Bradford C., “5 Common Encryption Algorithms and the Unbreakables of the Future” Available from URL: http://www.storagecraft.com/blog/5-common-encryption-algorithms/ (Last Access 28/02/2016).

[9] Pahal R., Kumar V., “Efficient Implementation of AES”, International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3 issue 7, pp. 290-295, July 2013.

[10] Kumar A., Sinha S, Chaudhary R., “A comparative Analysis of Encryption Algorithms for Better Utilization,” International Journal of Computer Application, vol. 71, pp. 19-23, May 2013.

[11] Ali E., El-Deen T., El-Sayed, El-Badawy A., Gobran S. N., “Digital Image Encryption Based on RSA Algorithm,” IOSR Journal of Electronics and Communication Engineering, vol. 9 issue 1, pp. 69-73, Jan. 2014.

[12] Ravali S.V.K, Neelima P., Dileep. P.S., Manasa. B., “Implementation of Blowfish Algorithm for Efficient Data Hiding in Audio,” International Journal of Computer Science and Information Technologies, vol. 5(1), pp. 748-750, 2014.

[13] Singh L, Bharti R.K, “Comparative Performance Analysis of Cryptographic Algorithms”, International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3 issue 11, pp. 563-568, Nov. 2013.

[14] Pallavi H. D., Uttam D. L. B., Patil V. B., “Comparative Implementation of Cryptographic Algorithms on ARM Platform,” International Journal of Innovation Research in Science, Engineering and Technology, vol. 2 issue 10, pp. 5505-5510, Oct. 2013.

[15] Masram R., Shahare V., Abraham J., Moona R., “Analysis and Comparison of Symmetric Key Cryptographic Algorithms Based on various File Features”, International Journal of Network Security & Its Applications, vol. 6(4), pp. 43-52, July 2014.

[16] Nadeem A. and Javed M. Y., “A Performance Comparison of Data Encryption Algorithms,” in International Conference on Information and Communication Technologies, IEEE, pp. 84-89, Aug. 2005.

[17] Salama D., Kader H. A., Hadhoud M., “Studying the Effects of Most Common Encryption Algorithms.” International Arab Journal of e-Technology, Vol. 2, No. 1, pp. 1-10, Jan. 2011.

[18] Lian S., Liu Z., Ren Z. and Wang Z., “Secure Advanced Video Coding Based on Selective Encryption Algorithms,” IEEE Transactions on Consumer Electronics, vol. 52(2), pp. 621-629, May 2006.

[19] Socek D., Magliveras S., Culibrk D., Marques O., Kalva H. and Furht B., “Digital Video Encryption Algorithms Based on Correlation-Preserving Permutations”, EURASIP Journal on Information Security, pp 1-15, 2007.

[20] Dhananjay M. D., Janwe J. N., “Video encryption using AES algorithm”, 2nd International Conference on Current Trends in Engineering and Technology (ICCTET), IEEE, pp. 332-337, July 2014.

[21] Yang M., Bourbakis N., Li S., “Data-image-video encryption,” IEEE Potentials, vol. 23(3), pp. 28-34, September 2004.

[22] Landge I., Contractor B., Patel A., and Choudhary R., “Image encryption and decryption using Blowfish algorithm,” World Journal of Science and Technology, vol. 2(3), pp. 151-156, 2012.

[23] Menezes A., Oorschot P. van, Vanstone S., “Handbook of Applied Cryptography,” CRC Press, 1996.

[24] Prabu A.V., Srinivasarao S. H., Apparao T., M. J., K. Babu Rao, “Audio Encryption in Handsets,” International Journal of Computer Applications, vol. 40 No. 6, pp. 40-45, Feb. 2012.