Optimal Keyless Algorithm for Security

Neha
M.Tech. Research Scholar
GZS PTU Campus,
Bathinda, Punjab

Paramjeet Singh, PhD
Associate Professor
GZS PTU Campus,
Bathinda, Punjab

Shaveta Rani, PhD
Associate Professor
GZS PTU Campus,
Bathinda, Punjab

ABSTRACT
In Modern era, every business is dependent on the Internet. The Network is growing so quickly that now at this stage no one can ever imagine anything without use of internet. But at the same time security over network is very important because of the vulnerability of data to eavesdropping. To protect the data from eavesdropping, it must be appropriately encrypted before sending over the network. There are two types of algorithms, keyed and keyless, exist to protect data. The keyed algorithms are efficient but to avoid the overhead of key generation and key management, keyless algorithms are getting popularity now days. The proposed algorithm "Optimal Keyless Algorithm for Security" represents a new way of using data itself to create a protective shield. The algorithm provides security at both character level as well as bit level. The number of rounds and the number of shifts applied at bit level are made data dependent to increase the security level, is a major advantage of the algorithm. The system is proposed with the motive to provide highest security level with minimum execution time in terms of encryption and decryption. This paper presents simulation results of proposed algorithm and its comparison with the commonly used JS keyless algorithm.

General Terms
Security, Cryptography, Key, Keyless, Encryption, Decryption

Keywords
Network, Security Algorithm, Block Cipher, Stream Cipher, Optimal, Data

1. INTRODUCTION
To make the transmission of data secure over the network, Cryptography is the best solution [1]. It is a transformation process in which original message is converted into a non readable form to make data secure and protective over the network. Cryptography is the combination of encryption and decryption process. Encryption is a process in which data is transformed in a format called cipher text or encrypted text. Decryption is a process of transforming the cipher text back to the original form. To decrypt the cipher text, one must have secret key (for key oriented algorithm) or decryption algorithm (for keyless algorithm). Encryption is precisely necessary when transmitting important data over unsecure mediums like the Internet.

In present time, Security of data has become the topmost priority for any business environment. To provide privacy to the data sending over network, encryption is must. It can be done in two ways:

Block Cipher: In Block cipher is a method of encryption, operation is applied on a fixed length group of bits known as blocks of data. The block may be of any length i.e. 8/16/32…/256 or more according to the requirement. The operation applied on every block remains same. [2]

Stream Cipher: In stream cipher method of encryption, cryptographic key (in keyed algorithm) and algorithm are applied to each bit in a data stream but one at a time [3]. Stream cipher is fast as compared to block cipher but it does not provide integration security and authentication.

Both techniques are useful but block cipher is used most commonly in modern cryptography.

Mainly there are two types of cryptographic algorithms on the basis of key management:

Key-Oriented algorithms use keys depending on the need i.e. public or private [4].

When the keys used at sender side and receiver side are the same. It is said to be as **Symmetric Key Cryptography**. Various symmetric cryptographic algorithms exists each one having their own unique advantages. For example: Data Encryption Standard (DES) [5], Blowfish algorithm, Advanced Encryption Standard (AES), Rivest Cipher (RC), International Data Encryption Algorithm (IDEA).On the other hand, if keys used at both sides are different then it is said to be as **Asymmetric Key Cryptography**. The algorithms that use asymmetric key cryptography are Digital signature Algorithm, XTR, Diffie Hellman.

Keyless algorithms are becoming more popular now days. Because there is no overhead of key generation and sharing. Only simple available logical operations are applied in such a manner so that data cannot be interpreted by the intruder while sending over an insecure media. Keyless algorithms use different ways like QUANTUM CRYPTOGRAPHY [3]. Keyless algorithms can also be categorized as symmetric and asymmetric.

2. LITRRATURE SURVEY
- **Jiwan Pokharel, N. Saisumanth, Dr. Ch. Rupa,T. Vijaya Saradhi [2]** described a commonly used keyless Algorithm which uses keyless symmetric cryptography. The algorithm is keyless and basically works on bit level. Various simple operations like one's complement, grey code conversion etc. are used.
- **M.Lakshmi, S.Kavitha [3]** described KUDOS algorithm (Keyless User Defined Optimal Security Encryption) which is based on the concept of user customization**.**
- **Jiwan Pokharel [7]** has presented the analysis of paper Keyless JS Algorithm encryption technique in. The main issue found in this algorithm is related to number of rounds. The problem lies only in specifying the number

of rounds in the header field while sending data because intruder can creep into it.

- **Chandra Prakash Dewangan,Shashikant Agrawal [9]** The research has been done in the field of improving security. The implemented encryption technique is analyzed by using a parameter called Avalanche effect.
- **T. Nie at el. [10]** presented that Encryption algorithms are essential for providing secure network but along with this power consumption issues should be sorted out. This algorithm considers the power consumption of encryption algorithm.
- **Suparna Karmakar, Sayani Chandra [11]** proposed an algorithm that uses various calculations to make the data secure and unpredictable. Use of various mathematical tools like Petri net and analysis technique of reach ability tree has made the algorithm very secure.

3. PROPOSED KEYLESS ALGORITHM

The proposed algorithm falls under the symmetric keyless cryptographic algorithm. In symmetric keyless cryptography decryption algorithm follows the same technique of encryption algorithm but in reverse order. The real advantage of proposed algorithm is the number of rounds applied in algorithm is made data dependent which are calculated independently at both sender side and receiver side. Only encrypted data is sent over the network which cannot be understood by the intruder as no clue is available to decipher the text. In the proposed algorithm, encryption is done at two levels i.e. character level and binary level.

Character Level: At character level, columnar transposition is applied. In this technique, first, the data is arranged in columns and then columns are rearranged using some methodology. [6]

Binary Level: After character level encryption, data is converted in to binary and various logical bit level operations are performed. Binary level is the lowermost level of security. Bit level calculation provides more security because its effects are visible to the character level also. [3]

3.1 Encryption Algorithm

Step 1: The input data is arranged in column matrix i.e. characters are positioned in matrix form and columnar transposition is applied.

Step 2: The characters are converted into ASCII values and then ASCII to binary.

Step 3: The size of data is noted and divided by 128 to get the total number of rounds. In this way algorithm is made data dependent.

Step 4: The binary plain text is spitted into 128 bit blocks.

Step 5: Divide 128 bit block further into 60-8-60 sub blocks.

Step 6: For even round

Perform 1's complement of the middle 8 bits

For odd round

Get the grey code of the middle 8 bits

Step 7: Perform XOR operation of last 8 bits of left 60 bit block and middle 8 bits.

Step 8: Right circular shift is performed on middle 8 bits and last 60 bit sub-block. Shift size is equal to the no of rounds calculated. Then grey code is obtained of the last 68 bits.

Step 9: Left Circular shift is performed between middle 8 bits and the first 60 bit sub block. The shift size is equal to the number of rounds calculated. Then grey code is obtained of first 68 bits.

Step 10: first 60 bits are appended to last 68 bits.

Step 11: Repeat steps 4-10 for the total number of rounds as calculated in step 3.

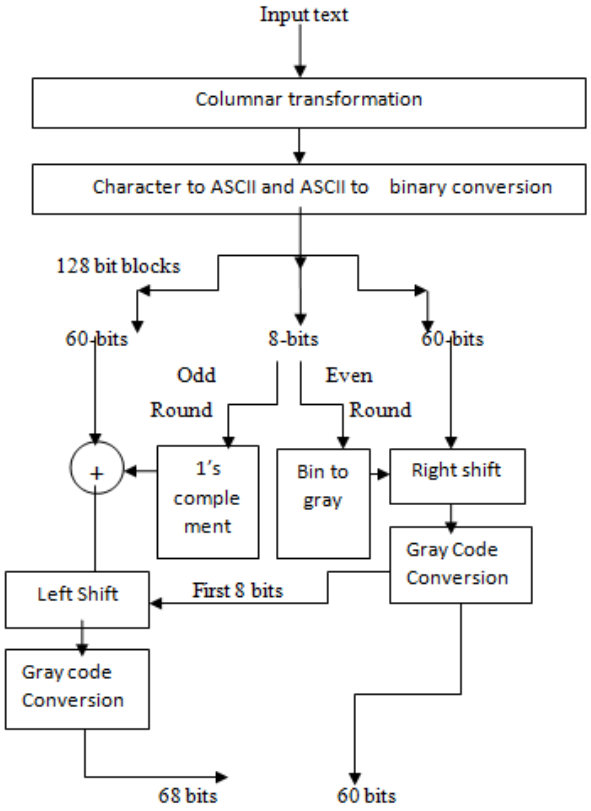

Figure 1: Flowchart of Encryption Algorithm

3.2 Decryption Algorithm

The decryption algorithm follows the same steps that are used in encryption algorithm but in reverse order.

Step 1: The total number of bits in the cipher text is divided by 128 to get the total number of rounds in the same way of encryption algorithm.

Step 2: Divide the encrypted bit stream into 128 bit blocks.

Step 3: Then divide each 128 bit cipher text block into 60-8-60 bit sub blocks.

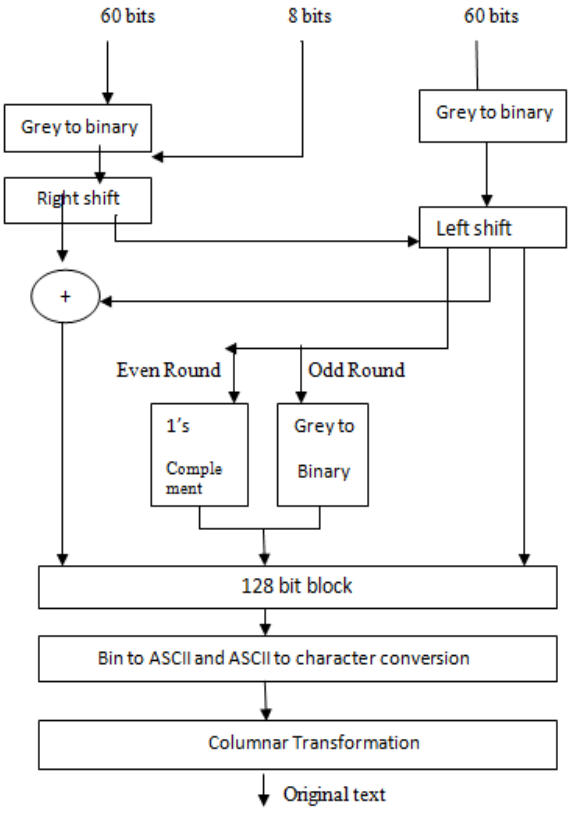

Figure 2: Flowchart of decryption Algorithm

Step 4: For left 68 bits, grey to binary code conversion is performed and then right circular shift is done on this data. The shift size is equal to the number of rounds calculated in step 1.

Step 5: For last 68 bits, grey to binary code conversion is performed and then circular left shift is done on this data. The shift size is equal to the number of steps calculated in step 1.

Step 6: Leftmost 8 bits of sub block obtained in step-3 are performed XOR with the 68 bit sub block obtained in step-2.

Step 7: For even round

Perform 1's complement of the middle 8 bits

For odd round

Get the grey code of the middle 8 bits.

Step 8: Repeat the steps 3-7 for number of rounds calculated by the algorithm.

Step 9: From the binary data, binary to ASCII and ASCII to Decimal conversion is done.

Step 10: Character level decryption is performed to get the original data.

4. IMPOERTANCE OF PROPOSED ALGORITHM

1. This algorithm is keyless so no need of key generation and key management and also no need of key transfer through third party. Because key sharing mainly causes a problem in security.

2. The number of rounds is made data dependent so the value keeps on changing according to the data which is the major advantage of this algorithm because on looking at cipher text number of rounds used in encryption cannot be estimated and thus data cannot be deciphered until the exact number of rounds used in the encryption are not known where as in JS algorithm number of rounds are fixed and need to be specified in the header section and sent over the network.

3. The number of rounds in proposed algorithm is calculated in both encryption and decryption algorithm independently by using length of text which provides more security as compared to commonly used JS algorithm because there number of rounds applied in encryption algorithm was sent along with the cipher text which may prove a clue for intruder.

4. The algorithm has been implemented over 128 bit blocks where as commonly used algorithm is implemented for 64 bit block size. Hence, block size has been increased which has directly reduced the encryption and decryption time along with increasing security level.

5. Encryption is done at character level and bit level in the proposed algorithm whereas in commonly used JS algorithm only bit level encryption is performed. So proposed algorithm provides double security.

5. EXPERIMENTAL RESULTS AND COMPARISONS

The algorithms are implemented using MATLAB. Different text inputs of different sizes are taken and results are calculated and compared by using three parameters:-

1. Encryption Time: It is the time consumed by encryption algorithm to convert original text to cipher text.

2. Decryption Time: It is the time consumed by decryption algorithm to convert cipher text back to the original text.

3. Avalanche Effect: Avalanche effect [9] is a powerful metric for measuring the security of an encryption algorithm. It is measured as little change in the input caused how much change in output. In other words it describes the bit shuffling obtained after applying the encryption algorithm. More is the change in the bit positions implies more randomness and difficulty in deciphering text and hence more security level.

Different text inputs have been taken to analyze the performance of proposed system and existing system. The results are compared and analyzed on the basis of above mentioned metrics.

Table 1: Comparison between commonly used algorithm and proposed algorithm in terms of encryption time

Input Size (in bytes)	Encryption Time (Existing System)	Encryption Time (Proposed System)
328	0.115416	0.045728
561	0.198228	0.076743
899	0.264142	0.168722
1126	0.300371	0.275297
1535	0.6257507	0.470493

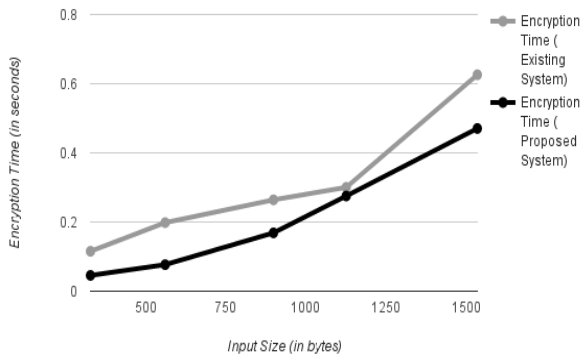

Figure 3: Graphical representation of Table 1

The graph clearly shows the improvement in encryption time of the proposed algorithm. The reason behind the optimality of the parameter is reduction in number of rounds. To achieve highest level of security with commonly used JS algorithm the number of rounds can be assigned a maximum value 256 with a cost of increase in encryption time. But in proposed algorithm the number of rounds is not fixed rather they depend on length of text and also the block size has been made 128 bit whereas it was 64 bit in commonly used algorithm. These modifications are the causes for improvement in encryption time. After encryption, Decryption is done at the receiver side. The results for decryption time parameter have been shown below:

Table 2: Comparison between commonly used algorithm and proposed algorithm in terms of decryption time

Input Size (in bytes)	Decryption Time (Existing System)	Decryption Time (Proposed System)
328	0.112998	0.029471
561	0.158887	0.072092
899	0.218725	0.158875
1129	0.26492	0.242684
1535	0.595787	0.451762

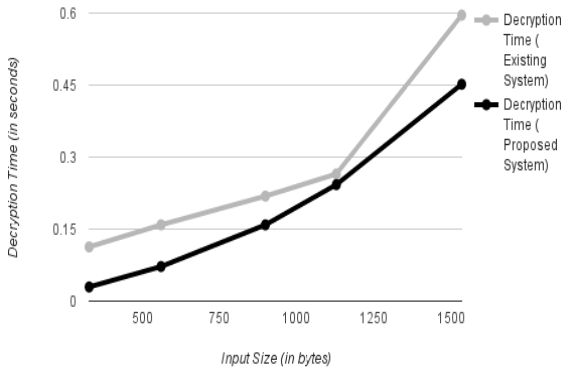

Figure 4: Graphical representation of Table 2

The above results show the improvement in decryption time parameter of the proposed algorithm as compared to commonly used JS algorithm. The reason of this improvement is similar to that of encryption. Although the number of rounds has been reduced in proposed algorithm, which has drastically improved the encryption time and decryption time parameters but along with fast execution proposed algorithm is proved more secure. To find out the security level of algorithm, a parameter avalanche effect is calculated. Avalanche effect describes the randomness produced by the encryption algorithm in terms of change in bit positions. More randomness implies more shuffling among bits and hence it is difficult to decipher the data and it shows the security level provided by the algorithm.

Table 3: Comparison between commonly used algorithm and proposed algorithm in terms of avalanche effect

Input Size (in bytes)	Avalanche Effect (Existing system)	Avalanche Effect (Proposed system)
328	22	48
561	15	56
899	20	70
1129	21	62
1535	12	64

Figure 5: Graphical representation of Table 3

It is clear from the graph that avalanche effect is found more in proposed system as compared to commonly used system. So proposed algorithm is more secure.

6. CONCLUSION

The "Optimal Keyless Algorithm For Security" provides a mechanism to encrypt and send data over network without producing external key. Data itself is used to make a protective shield. The simple logical operations have been applied on the data to transform it into another form which can be understood only by the recipient by using decryption algorithm. It is keyless algorithm so time and memory gets saved which is consumed in key generation and management mechanism. The results shown in the last section proves better performance of the proposed algorithm in terms of encryption time, decryption time and avalanche effect. Number of rounds is made data dependent, only encrypted data is sent over the network and by looking at the cipher text it is not possible to decrypt the data so it provides more security over commonly used algorithm because in that number of rounds applied in the encryption algorithm are specified in the header field and sent over the network so that data can be properly decrypted at receiver side. But it creates a breech into the security because round number can be attacked by intruder in any way. Encryption is done at both character level and bit level so more security level is achieved. High level of avalanche effect produced in the proposed algorithm clarifies that it is highly secure algorithm along with quick execution. The algorithm

can be enhanced in future to provide encryption at block level along with character level and binary level according to the security level needed. Encryption time and decryption time can be optimized by increasing the block size and for further optimizing the security level, number of transformations can be increased at both character level and bit level. In addition, algorithm can be enhanced to make it suitable for all kind of data i.e. audio, video, images etc.

7. ACKNOWLEDGMENTS

I would like to thank my guide Prof (Dr.) Paramjeet Singh and Prof (Dr.) Shaveta Rani for their consistent guidance and support in the accomplishment of this research work entitled "Optimal Keyless Algorithm for Security". I am also thankful to my family and friends who keep encouraging me to complete my work in time. My acknowledgements would not be complete without expressing my personal belief in and gratitude towards God, our creator. None of this would have been possible without His blessings.

8. REFERENCES

[1] Nishika, Rahul Kumar Yadav, "Cryptography on Android Message Applications – A Review", IJCSE, Volume 5, May 2013.

[2] J.Pokharel, N.Saisumanth, C.T.Rupa, V.Saradhi, "A KEYLESS JS ALGORITHM", International Journal of Engineering Science & Advanced Technology, Volume 2 No.05, Sep-Oct 2012, ISSN: 2250-3667, pp. 1397 – 1401.

[3] M.Lakshmi,S.Kavitha, "Keyless User Defined Optimal Security Encryption", International Journal Of Engineering And Computer Science, Volume 2 , June 2013, ISSN: 1788-1793.

[4] E. Thambiraja, G. Ramesh, Dr. R. Umarani, "A survey on various most common encryption techniques", International Journal of advanced Research in Computer Science and Software Engineering, Volume 2, Issue 7, July 2012, ISSN: 2277

[5] A.Kumar, S.Jakhar, S.Makkar, "Comparative Analysis between RSA and DES," International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2 No.07, 2012, ISSN: 2277 128X, pp. 386-391

[6] Md. Mizanur Rahman,"Any file encryption by translating ASCII value of characters", International Journal of advanced Research in Computer Sciences(IJARCS), Volume 2, Issue 3, May- June 2012, ISSN: 0976-5697.

[7] Jiwan Pokharel, "Analysis of Keyless JS Algorithm", ResearchGate Publications, Dec 2013

[8] 128XUdepal Singh, Upasna Garg, "An ASCII value based text data encryption system", International Journal of Scientific and research Publications (IJSRP), Volume 3, Issue 11, November 2013, ISSN 2250-3153.

[9] Chandra Prakash Dewangan,Shashikant Agrawal, "A Novel Approach to Improve Avalanche Effect of AES Algorithm" , International Journal of Advanced Research in Computer Engineering & Technology, Volume 1, Issue 8, Oct 2012.

[10] T.Nie, L.Zhou, Z.MingLu, "Power evaluation methods for data encryption algorithms," IET Software IEEE, Volume 8 No.01, 2014, pp. 12–18.

[11] S.Karmakar, S.Chandra, "An Approach for Ensuring Security and its Verification," International Journal of Computer Science Engineering, Volume 2 No.03, 2013, ISSN: 2319-7323, pp.42-49.

[12] V. Gupta, G. Singh, R. Gupta, " Advanced cryptography Algorithm for improving data security", International Journal of Advanced Research in Computer Science and software Engineering, Volume 2, Issue 1, January 2012, ISSN: 2277 128X.

[13] Malik S., "A novel key based transposition scheme for text encryption", 2011 IEEE Frontiers of Information Technology (FIT), DOI: 10.1109/FIT.2011.44,pp. 201-205.