Keyless User Defined Optimal Security Encryption

Akhil Kaushik, Satvika, Manoj Barnela, and Anant Kumar

Abstract—Cryptography and encryptions, these terms are now-a-days have an unseen impact in the emerging field of network and its security. As security of data on any kind of system has become the first priority for the organization, the methods which are used to ensure security not only need to be strong and efficient, but should also be easy to execute and implemented. With progress in technology, encryption came up with a big boom, taken as a weapon of ultimate security. It is an earliest art and it is defined as the science of writing in secret code. Basically, the cryptography algorithms are categorized into two types on the basis of key management, which are key-oriented and keyless encryption algorithms. The prime goal leading the design of an encryption algorithm must provide security against unauthorized attacks. Key-oriented algorithms are very efficient but they were very bulky to manage as key handling must be done. Due to the great overhead, keyless algorithms seem an attractive option. But what about security by keyless, here is the solution- KUDOS encryption, which is a keyless security algorithm to provide ultimate security at the one level above of key- oriented.

Index Terms—Cryptography, decryption, encryption, keyless, kudos, sequence counter, user defined.

I. INTRODUCTION

In computer and communication systems security issues play a crucial role and must be addressed before hand to guard against illicit attacks[1]. In the global communal world, where faster access to precise information is the most basic need; security of confidential data during transfer from one place to another place is a major concern. There are a lot of sheltered approaches that can be applied inside the organization's premises to keep the data safe and sound. But when this confidential data or information comes out of the company's premises, it becomes susceptible to the unauthorized attacks by hackers or opponent companies [2]. There can be various techniques that can be used to attain secure transfer of data like firewalls, proxy servers, and steganography, data security plans against worms, viruses or denial-of-service attacks [3].

But cryptography proves itself as a central tool for achieving data and software protection. Cryptography can be defined as the art or science of altering information or change

it to a chaotic state, so that the real information is hard to extract during transfer over any unsecured channel [4]. Latest advancements in technology and new concepts like quantum cryptography have added a complete new dimension to data security. The strength of this cryptographic technique comes from the fact that no one can read (or steal) the information without altering its content. This alteration alerts the communicators about the possibility of a hacker and thus promising a highly secure data transfer [5]. Due to this advantage, quantum cryptography has grasped a great deal of attention and huge amount of research is being carried out on it for safeguarding of business-critical data.

During the course of time, various encryption algorithms have been developed to achieve the ultimate aim of safe environment for information transmission. However, the principal objective guiding the design of an encryption algorithm must be security against all possible unauthorized attacks. However for all practical applications, performance and the cost of implementation are also important concerns. The best cryptographic algorithm is the one that strikes a good balance between security and performance [6].

Apart from the well known division of private-key and public-key cryptography, there is another way to categorize the encryption process depending upon the input size. One approach encrypts data bit wise i.e. one unit at a time. The second approach takes a block of data and encrypt in it one go. The former is called stream cipher and latter is called block cipher. Block encryption is the faster way but produces same ciphertext for the same plaintext and encryption key [7]. However stream ciphers does not have any such pitfall and is considered more secure. Stream ciphers have lower hardware complexity as compared to block ciphers. An important point to note is not to use the same starting state twice. Though stream ciphers are comparatively take more time to process and hence are not widely used in real time applications[8].

Fig. 1. Working of a stream cipher

Most of the cryptographic algorithms fall into either of these two categories, but the algorithm proposed here uses a combination of stream and block ciphers. The proposed algorithm is known as Keyless User Defined Optimal Security (KUDOS). As the name suggests, KUDOS algorithm is

Manuscript received February 14, 2012; revised March 14, 2012.

A. Kaushik is with the Department of Computer Science and Engineering, Technological Institute of Textile and Sciences, Bhiwani, Haryana, India (e-mail: akhil.kaushik@yahoo.com).

Satvika is with the Department of Information Technology, Technological Institute of Textile & Sciences, Bhiwani, Haryana, India (e-mail: satvika16oct@gmail.com).

M. Barnela is with the Department of Electronics, Technological Institute of Textile and Sciences, Bhiwani, Haryana, India (e-mail: m.barnela@gmail.com).

A. Kumar is a passed out student of Computers Engineering, Technological Institute of Textile and Sciences, Bhiwani, Haryana, India (e-mail: anant.berawal@gmail.com).

keyless because there is no key involved in the encryption process. Only sequence counters are used. The sequence counters have a definite start point and a particular increment value. The sequences are merged with the actual data at a particular level (described later) and encrypted data then replaces original data.

As the KUDOS is user defined, the user can select the starting of the sequence. There are sequence counters on each and every level with different type of increment value. User can define his/ her own counter number from which the counter will start, and then this information will be packed in a packet and appended to the encrypted data. If user does not define anything about the start point of sequence counter, then default values will be taken and there will be no need to append anything to the encrypted data. That's why it is called user defined algorithm.

II. PROPOSED SYMMETRIC KEY ALGORITHM

The KUDOS cryptographic algorithm basically falls under the symmetric encryption i.e. the same key is used at both ends to encrypt and decrypt the data. However, KUDOS actually depends on the sequence counter instead of the encryption key. The major benefit of using KUDOS over other encryption algorithms is its power of customization. The user can manipulate the sequence counter according to his needs; whether he wants it to be simple and faster or hard to crack and secure.

Sequence counter- The first thing which is new in the encryption scheme is sequence counter, which is used to provide the dynamic behaviour while converting the characters from one form to another. Sequence counters are the imposed temporary key to alter the characters and bits in the original data. The sequence is applied to the chosen number of data bits and the sequence number changes according to an algorithm which needs not to be saved or transferred over the network.

Suppose we have an ASCII sequence of characters:

65, 68, 121, 65, 98, 99

And we add an odd no. sequence counter to it.

11, 13, 15, 17, 19, 21

Now suppose add operation is used for them, so solution will be

76, 81, 136, 82, 117, 120

Data is completely changed and it is more powerful than a simple caesar cipher. We only need to know the first character of the sequence and it need not to be transferred over the network.

The sequence counters in the proposed algorithm are used at following levels:

Block level- This is the uppermost level where blocks characters of a single block or line are transposed according to the sequence counter.

Character level- Every character have an equivalent ASCII value, which can be merged with the sequence counter. The example above given in of this category.

Binary level – The binary level is lowermost level. The calculation done here is totally in form of 0 and 1. Bit level calculation provide more security because its effect are visible

to all the level above it include character level.

In the proposed algorithm we have used both stream cipher and block cipher to enhance the security by using advantages of both i.e. high diffusion and bit level security.

III. ENCRYPTION PROCEDURE OF KUDOS

KUDOS is based on the idea of maintaining a balance between security and speed of the encryption algorithm. In this algorithm, we have taken two predefined stacks along with a logic based lookup concept. The first stack holds the information about the sequence counter implemented by the user. Whereas the other stack maintains the record for the data on which encryption has to be performed. The encryption process is of five steps; and four out of these five steps can be customized. Only step 2 cannot be altered by the user. The encryption process does a variety of binary operations like Shift Left Operation on the message for protecting it against unauthorized attacks. In this operation, bits are shifted left to one place and Most Significant Bit (MSB) is placed to Least Significant Bit (LSB) as shown in the diagram below[9].

Fig. 2. Shift-left operation

The steps of encryption are given as follows:

- 1) The plain text is read from input file line by line i.e. separated by a full stop. The user has the option of choosing the sequence counter for transposition of characters. This sequence number must be a positive whole number. The shift-right operation is used here for transposing the characters. The first line characters are transposed using the sequence counter and the next line characters are transposed by incrementing the sequence counter by one. If the user opts not to choose the sequence number, the default sequence number 1 is selected. Suppose the value for first line is 1, for second line will be 2, and so on. If user specifies it as 5, value for first line will be 5, for second line it will be 6, for third line 7 etc. This step is categorized under block level encryption.
- 2) This part of encryption is user independent and is a block level encoding. Here, two strings of equal length are chosen from the text file and are shuffled. The shuffling takes place by placing one character of first string followed with one character from second string; and so on. For example: Following are the strings of length 20 characters selected:
Q W E R T Y U I O P A S D F G H J K L Z
and q w e r t y u i o p a s d f g h j k l z;
Then the output of this phase is:
Q w E r T y U i O p A s D f G h J k L z and
q W e R t Y u I o P a S d F g H j K l Z.
- 3) The next step is the stream encryption of the plaintext. The output from the previous step is taken as the input.

Character by character is read and then transformed into ASCII code. The ASCII value of each character is then incremented by some random number (sequence counter). This is the second sequence counter that user can pick. For example: The input is KUDOS, then ASCII value of each character is K – 97, U – 107, D – 90, O – 101 & S – 105.

The user chooses sequence counter 1 for 1st character, 3 for next, 5 for next and so on. KUDOS is encoded into LXIV\.

- 4) Now, the input character from last step is converted into 8-bit binary number. Here a block of 64 bits are taken into account and then a dynamic sequence counter is decided by the user. Then shift-left operation is performed on this 64-bit data based on the opted sequence counter. For example: 1 shift-left operation for first block, 4 shifts for second block, 9 shifts for third block, and so on.
- 5) The encryption process is continued on next step on the base of compliment. A sequence counter is decided to work on the 8-bit level of intermediate ciphertext. A sequence of bits will be complemented and the next block of same number of bits will remain unaffected. For example: First 2 bits will be complimented, then next 2 bits remain same; then next 4 bits are complimented and the next 4 bits remain unchanged. This sequence counter is also identified by the end user, hence customizing the whole encryption process according to his particular needs.
- 6) The steps 1 to 5 are repeated till the input plaintext is changed into the ciphertext.

Fig. 3. KUDOS encryption process

The whole encryption process contains five steps, four of which can be altered by the user. The system doesn't have any encryption key, though the sequence counter is opted by the user serves as the key. The total length of sequence counter is 4 bytes, one byte for each encryption step. The combination of four sequence-counters is treated as the encoding key. If the user chooses to customize the sequence counters, the combined key will be sent along with the data for decryption at other end. Else, the default sequence counters will be used for encryption. Key customization by user will help the users to adapt to the cryptosystem efficiently; and he can choose between power and speed of the algorithm to better secure the

data in an efficient way.

IV. DECRYPTION PROCEDURE OF KUDOS

As the KUDOS cryptosystem is based on symmetric encryption, the decryption process in this algorithm is exactly the reverse of the encryption. The appropriate sequence counters are identified from the given sequence table for various steps of decryption.

Fig. 4. KUDOS decryption process

- 1) The cipher text is read from the received file.
- 2) The corresponding sequence-counter is read from the central database server.
- 3) Compliment is performed on the chosen number of bits at binary level.
- 4) Binary shift-right operation is done on the modified cipher text with the help of next sequence counter.
- 5) With the help of subsequent sequence-counter, reverse binary operations are performed at ASCII level on the result of previous step.
- 6) Reverse shuffling is performed on string level data to get intermediate cipher text.
- 7) Transposition of characters of a single line is repeated to obtain the plaintext.
- 8) Steps 1 to 7 are repeated till the whole ciphertext is converted into plaintext.

V. IMPLEMENTATION LANGUAGES FOR KUDOS

The algorithm has been implemented in Java and it can be implemented in any language that supports Unicode system like C or C++.

VI. PERFORMANCE EVALUATION

The basic idea behind developing a cryptographic algorithm is defense provided to the data being transferred across an untrusted link. The adaption of any encryption algorithm depends upon the tradeoff between security and speed[10]. If the algorithm offers great speed with greater protection against unauthorized attacks, it is widely accepted. Cryptosystems developed on the concept of block encryption suffered from the problem of producing the same output for same input and same encoding key; which make them

vulnerable to 'replay attacks'[11]. However, stream ciphers suffer from speed issues as they take a lot of time for execution[12]. KUDOS partially uses both block and stream ciphers; thus reaping benefits of both and excluding disadvantages.

Encryption at three levels i.e. block level, character level and binary level boosts safety three times of other popular algorithms. Moreover, as the user can change the sequence-counters on his/ her wish every time before sending data; hence provides more flexibility and additional security to the business critical data. One of the most popular attack is the Brute-force attack, in which all possible combinations of key are tried to deduce the secret key[13]. Three levels of encoding make it harder to crack even using brute-force attack. Another remarkable approach used in KUDOS encryption algorithm is error control mechanism. The algorithm can be implemented with or without error control. In error control mode, the algorithm keeps track of the character transposition because some letters may be converted to special symbols or to other letters. Experiments show that KUDOS is 99.96% free of errors, hence maximizing defense of data.

KUDOS algorithm is pretty fast because of code optimization techniques employed in the programming. Especially loop optimization is done because loops consume maximum time of any program. The encryption and decryption timings for the KUDOS cryptosystem has been observed and listed in the following table:

TABLE I: PERFORMANCE EVALUATION OF KUDOS ALGORITHM

File Size	Encryption Time	Decryption Time
30 Kb (300 characters app.)	0.01 seconds	0.01 seconds
160 Kb (160700 chars app.)	0.22 seconds	0.23 seconds
1 Mb (000000 app.)	0.90 seconds	0.92 seconds

The memory requirements for KUDOS are low as well. The maximum size of sequence counter to store is 4 bytes of Integer value. The overhead of handling and storing sequence counters is very less. Hence, KUDOS provides superior security with greater speed and minor overhead as compared to established encryption standards.

VII. CONCLUSION

The algorithm proposed in this paper is Keyless User Defined Optimal Security Encryption (KUDOS) is based on the concept of user customization. The algorithm doesn't use the traditional approach of using an encryption key; but defines a series of sequence-counters for encoding. The cryptosystem gives extra power to the user i.e. to choose the sequence-counters. Thus it is up to the user to maintain a balance between speed and security provided by the algorithm. The user can increase or decrease security and speed depending upon his/ her needs. The cryptographic algorithm is based partially on both stream and block encryption, hence the output of same input block over same input sequence-counter is dissimilar and provides enhanced security. Moreover, for security enhancement, the encryption is done at three levels: block level, character level and bit level. It makes it more complicated and harder to break even

with default values of sequence counters. Error control mechanisms can be used to safeguard the data against The KUDOS encryption algorithm is successfully tested on text data. KUDOS efficacy in text data encryption along with its supreme security; seems like an answer to the future encryption issues.

Future development will include:

- 1) Implementation of KUDOS for image and audio data.
- 2) Hardware realization of KUDOS.
- 3) Usage of compression technique along with encryption procedure.

```

import java.awt.*;
import java.awt.event.*;
import java.applet.*;
import javax.swing.*;
import java.awt.Cursor;
import javax.swing.event.*;
import javax.swing.BorderFactory;
import javax.swing.border.Border;
import javax.swing.border.TitledBorder;
import javax.swing.border.EtchedBorder;
import java.lang.*;
import javax.swing.Box;
import javax.swing.BoxLayout;
import java.awt.image.BufferedImage;
import java.io.File;
import java.io.IOException;
import java.io.*;
import javax.imageio.ImageIO;

```

Fig. 5. Sample Plaintext

```

javaj tropmi.twa.avaj tropmi
;*.twa.tneve.avaj tropmi
;*.telppa.xavaj tropmi
;*.gniws.avaj tropmi
;*.twa.xavaj tropmi
;rosruc.gniws.tneve.xavaj tropmi
;*.gniws.xavaj tropmi
;yrotcaFredroB.gniws.redrob.xavaj tropmi
;redrob.gniws.redrob.xavaj tropmi
;redrobdehctE.gniws.redrob.xavaj tropmi
;redrobdehctE.gnal.xavaj tropmi
;*.gniws.xavaj tropmi
;xoB.gniws.avaj tropmi
;tuoyalxob.twa.egami.avaj tropmi
;egamiIdereffuB.oI.avaj tropmi
;eliF.oI.avaj tropmi
;noitpecxEoI.oI.xavaj tropmi

```

Fig. 6. Intermediate output of encryption step 1

```

.vwj.tnovm..vwj.tvojmt
o*mt
l*ataa trepeiataa arap ir;p.ie;ptaoxm
j*tttoamx
v*jgtiosma
ar pr;paia; .rwp.ia;a. nrwp.iv;jo;r.Cngwi.satae erxpvi j tyoomc
s*ug.inswx.vnjvt.oama
;rrptiaprid;oe.rnBwg.iesrrbdxov.jataofme
rrBdgoi.snrwd.oe.rabax vrjdttoo.mv
jrtdoomd
lrid.ondwh.te rrbpaia; errpbie;teTrgBiescrevgjat.oama
xrBpgii;s.anaw .ra.an lrxpvij;too.mn
w*.gvijstxotmo
mt
oeaaxIBeteafEbaoi.avaj priip;iu;yglmod.rwf.ug.mi.avaj ortpeix;O1.Fiox.vvjtttoomm
;ntiopmc
EeIio..iaaaa rrppii
;* oee=a imfr;j*{1})F wnNr(Sm1rnJ.wingfme.a
F)
e
ig gei"te(auFec

```

Fig. 7. Intermediate output of encryption step 2

Fig. 8. Intermediate output of encryption step 3

Fig. 9. Final cipher text output

VIII. RESULTS

The following figures show the output of KUDOS encryption algorithm. The intermediate results for each encryption step are shown below:

REFERENCES

- [1] K. Gary, an Overview of Cryptography, an online article. Available: www.garykessler.net/library/crypto.html.
- [2] Y. Hao, E. Osterweil, D. Massey, L. Songwu, and Z. Lixia, "Deploying Cryptography in Internet-Scale Systems: A Case Study on DNSSEC", *IEEE Transactions on Dependable and Secure Computing*, vol. 8, no. 5, pp. 656-669, Sept 2011.
- [3] C. Bissell and A.K.Vladimir, "Pioneer of the sampling theorem, cryptography, optimal detection, planetary mapping" [History of Communications], *IEEE Communications Magazine*, vol. 47, no.10, pp. 24 - 32, Oct 2009.
- [4] J. Katz and Y. Lindell, *Introduction to Modern Cryptography: Principles and Protocols*, 1st ed. USA: Chapman and Hall/ CRC, 2007.
- [5] R. S. Vignesh, S. Sudharssun, and K. J. J. Kumar, "Limitations of Quantum & the Versatility of Classical Cryptography: A Comparative Study," *2nd International Conference on Environmental and Computer Science ICECS '09*, 2009, pp. 333 - 337.
- [6] J. P. Kaps, G. Gaubatz, and B. Sunar, "Cryptography on a speck of dust," *IEEE Transactions*, pp. 38-44, Feb 2007.

- [7] A. J. Menezes, P. C. V. Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, 5th ed. USA: CRC Press Inc., 2001.
- [8] G. Brassard, "A note on the complexity of cryptography," *IEEE Transactions*, pp. 232-233, Oct 2000.
- [9] Basic Cryptographic Algorithms, an online article. Available: www.itsc.state.md.us/oldsite/info/internetSecurity/Crypto/Cry-ptoIntro.htm#Algorithms.
- [10] J. Hoffstein, J. Pipher, and J.H. Silverman, *An Introduction to Mathematical Cryptography*, 1st ed. USA: Springer, 2010.
- [11] A. M. Al-Abiachi, F. Ahmad, and K. Ruhana, "A Competitive Study of Cryptography Techniques over Block Cipher," *13th International Conference on Computer Modelling and Simulation (UkSim)*, 2011, pp. 415 - 419.
- [12] S. Mathew and K. P. Jacob, "A New Fast Stream Cipher: MAJE4," *Proc. of annual IEEE, INDICON2005*, Dec 2005, pp. 60-63.
- [13] A. Ahmad, A. Biri, H. Afifi, and D. Zeghlache, "TIBC: Trade-off between Identity-Based and Certificateless Cryptography for future internet," *IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications, Evry, France*, Sept. 2009, pp. 2866 - 2870.

Akhil Kaushik has received the Master degree in Information Technology from Central Queensland University, Melbourne, Australia. Currently he is working as an Assistant Professor in "Technological Institute of Textile & Sciences", Bhiwani, Haryana, India. He has his research contribution four at International level in various proceeding like IEEE, ICNIT, ICFN, IJARCS and three at National level.

His research interest includes network security, cryptography and Artificial Intelligence

Ms. Satvika has received the Master degree in Computers Science and Engineering from Chaudhary Devlal University Sirsa, Haryana, India. Currently she is working as an Assistant Professor in "Technological Institute of Textile & Sciences", Bhiwani, Haryana, India. She has published one international research papers in IJARCS and one at national level. Her research interest includes Artificial Intelligence and network

security.

integrated circuits and cryptography.

Manoj Barnela has received Master degree in VLSI designing and Embedded systems from Guru Jambheshwar University Hisar, Haryana, India. Currently he is working as an Assistant Professor in "Technological Institute of Textile & Sciences", Bhiwani, Haryana, India. He has published three international research papers in IEEE, ICNIT, IJARCS and three at national level. His research interest includes VLSI designing, CMOS digital

Anant Kumar has done his bachelor from T.I.T&S College, Bhiwani and is currently working as software engineer in Newgen Software Technologies Limited in Okhla, Delhi, India. His research interests include Image encryption and Natural Language Processing.