

An Approach of Bitwise Private-Key Encryption Technique Based on Multiple Operators and Numbers of 0 and 1 Counted From Binary Representation of Plain Text's Single Character

Ramkrishna Das, Saurabh Dutta

Abstract—Private-key cryptography is a cryptographic system that uses the same secret key to encrypt and decrypt messages. The Existing Private-key cryptography systems are complex and not full proof in respect of security concern as the distribution of the private-key without interpretation is very hard to achieve.

Here, we have proposed an idea to increase the security of Private-key encryption technique. We have focused on the secret procedure to retrieve secret value from the private-key rather than securing the actual private-key value. The encryption is done by the secret value derived from the private-key. The secret value is being derived by making arithmetic operation between the user defined base value and a decimal value. That decimal value is derived by performing arithmetic operation between number of '0' and number of '1' counted from 8 bit representation of a plain text's character. The operators are being supplied by the user. As the numbers of '0' and '1' are different for several numbers of character, so we get different secret private-key value for several numbers of character in the plain text. Thus the security is increased.

Index Terms - Arithmetic operators, Counting of '0' and '1' from binary representation of a character, Private-key Encryption, Stream Cipher.

I. INTRODUCTION

Private-key cryptography refers to an encryption method where both the sender and the receiver share the same secret key or their keys are different, but they are related in an easily computable way. Fig-1 represents a private-key encryption method.

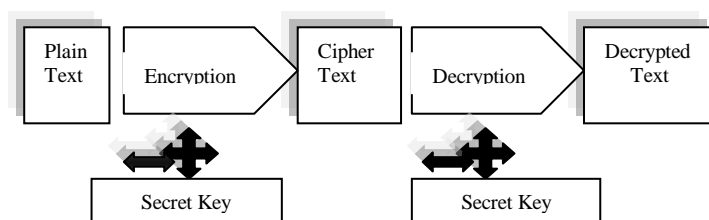


Figure 1: A Private-key Encryption Scheme

The main problem of private-key encryption technique is to distribute the private-key securely. All the existing private-key encryption systems suffer from lack of security. Here, we have proposed a scheme to increase the security of the existing private-key encryption system. We have developed a secret procedure which is responsible to retrieve the secret value from the private-key. The value is being used for encryption and decryption. The secret value is being generated by making different arithmetic operations (operations for base values, operations between N (0) and N(1)) on user defined base value and on decimal values, derived by performing arithmetic operation between number of '0' and number of '1' counted from 8 bit representation of a plain text's character. The operators are being supplied by the user. As the number of '0' and '1' are different for several number of character, so we get different secret private-key value for several numbers of character in the plain text. Combination of user defined operators and derived secret value from plain text construct the private-key. Hence the security is increased as we focus on securing the retrieving procedure rather than the private-key value. Secret value can't be retrieved without the knowledge of the retrieving procedure.

In this paper section-II describes the encryption process; Section-III describes the decryption process. Experiment results are being described in section-IV and section-V draws the conclusion.

II. ENCRYPTION PROCESS

A. Plain Text Formation: -

A unit of one character (8 bits) is being encrypted by the secret value at a time. Let 'C' is a character which is being present in the plain text. ASCII value of 'C' is 67 so corresponding 8 bit binary representation is stored in the an array PLAINTEXT with dimension 8. Fig-2 represents the entire procedure.

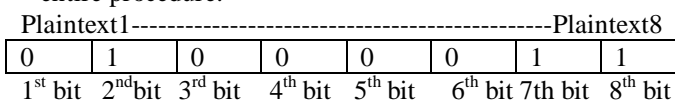


Figure 2: Formation of Plain Text

Step-A.1 Read the input file as plain text from the keyboard.

Step-A.2 Read one character at a time from the input file

Manuscript Received on May, 2013.

Ramkrishna Das, Department of Computer Applications, Haldia Institute of Technology, Haldia, India.

Saurabh Dutta Department of Computer Applications, Dr B.C.Roy engineering College, Durgapur, India.

An Approach Of Bitwise Private-Key Encryption Technique Based On Multiple Operators And Numbers Of 0 And 1 Counted From Binary Representation Of Plain Text's Single Character

text file. The ASCII value of 'O' is 79. We convert 79 in 8 bit binary representations and store it into array CT. The secret value has been derived from the private-key in the step-c. Now we are going to perform the XOR operation between the cipher text (CT) and the secret value derived from the Private-key (DV). Fig-9 represents the XOR procedure

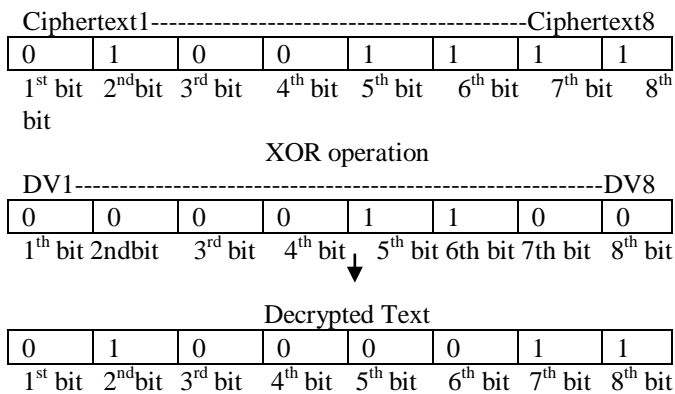


Figure 9: Generation of Plain Text by block wise XOR operation between Cipher Text and Secret Value derived from the key.

Corresponding ASCII value which is 67 is generated from this 8 bit binary string (Decrypted Text). Then corresponding character which is 'C' is generated from this ASCII value. In this way all the characters of the cipher text file are decrypted. Those characters are stored into the decrypted text file and receiver will be able to get the plain text.

IV. EXPERIMENTAL RESULT

Here we have displayed the content of the plaintext, the corresponding encrypted content of the cipher text and the corresponding content of the decrypted file. Table-1 demonstrates the entire content of the source files, encrypted file and the decrypted file.

Table I: Corresponding content of source, encrypted, decrypted file

Content of the source file (f.txt)	Content of the Encrypted file (f_en.txt)	Content of the Decrypted file (f_de.txt)
12345678 tfeeygewhfw j BHGDJRJH KMH =][[';./.,@#\$\$% ^&*()_+{ } ":?><	=>!8'\$4 ftwwiw wydywx44LFUJJ^F FY_EF ?-MK5+ ?< T/*)N*&&%& %Q9lus,(1..	12345678 tfeeygewhfwj BHGDJRJHK MH =][[';./.,@#\$\$% &*()_+{ }":? ><

We have used +, -, 10, 2 as base operator, operator between N(0) and N(1), base value and number of time base operator is being executed. The encryption is done by using the secret derived value from secret private-key which is dedicated and may or may not be distinct for each character of the plain text. The private-key value is dependent on the number of '0' and '1' of a single character in the plain text. There are N numbers of

may or may not be distinct private-key values if a plain text has N number of characters. The encryption or decryption has taken 14954 milliseconds.

We have executed our program on 15 number of files of different types (*.com, *.txt, *.exe, *.sys and *.dll). We have taken 3 numbers of files of each type. The Execution results are being displayed in the following Tables.

Table II: - Execution Result for *.com files

Source File name	Source File size (Byte)	Encrypted file size (Byte)	Encryption / decryption time (Mille seconds)
loadfix.com	1131	1131	113547
README.COM	4217	4217	224047
diskcomp.com	9216	9216	391781

Table III: - Execution Result for *.txt files

Source File name	File	Source File size (Byte)	Encrypted file size (Byte)	Encryption/ decryption time (Mille seconds)
ReadMe.txt		286	286	113906
LICENSE.TXT		4829	4829	323500
TechNote.txt		9232	9232	579831

Table IV: - Execution Result for *.exe files

Source File name	Source File size (Byte)	Encrypted file size (Byte)	Encryption/ decryption time (Mille seconds)
WINSTUB.EXE	578	578	62359
mqsvc.exe	4608	4608	213578
label.exe	9728	9728	410656

Table V: - Execution Result for *.sys files

Source File name	Source File size (Byte)	Encrypted file size (Byte)	Encryption/ decryption time (Mille seconds)
VIAPCI.SYS	2712	2712	148297
rootmdm.sys	5888	5888	273968
sffp_mmc.sys	10240	10240	480703

Table VI:- Execution Result for *.DLL files

Source File name	Source File size (Byte)	Encrypted file size (Byte)	Encryption/ decryption time (Mille seconds)
iconlib.dll	2560	2560	130156
KBDAL.DL L	6656	6656	286015
panmap.dll	10240	10240	592031

If we are looking to the relation between encryption or



decryption time and the source file size, then we get almost a straight line in figure 10. This is a graphical representation of the relation between the encryption time and the source file size. That means the relation is linear. As the source file size increase the encryption time increases and vice versa. The encryption or decryption time do not depend on the type of the file. Fig-10 represents the relation between the encryption time or decryption time and the source file size.

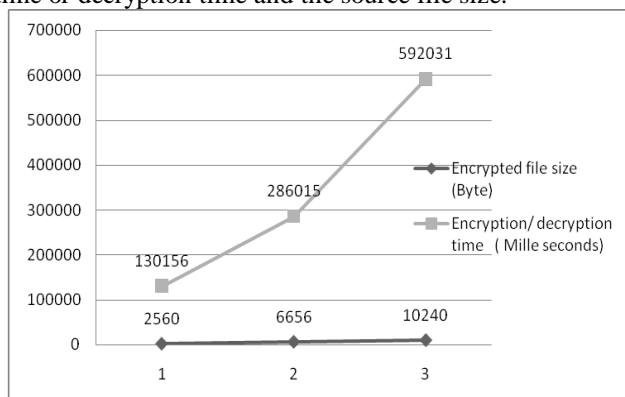


Figure 10: Relationship between Encryption time and source file size

The Pearsonian Chi-square test has been performed here to decide whether the sample of encrypted files may be supposed to have arisen from a specified population. The Pearsonian Chi-square is defined as follows: $\chi^2 = \sum \{(f_0 - f_e)^2 / f_e\}$ Here f_e and f_0 respectively stand for the frequency of a character in the source file and that of the same character in the corresponding encrypted file. [5]

If the Chi-square value is a higher one that means there is a higher frequency in source file and a lower frequency in encrypted file for a specific character. So, this proposed encryption procedure generate highly secured encrypted file. Table VII shows the Chi-square value for different types of file.

Table VII: Chi Square test Result on different files.

Source File name	Source File size (Byte)	Encrypt ed File size (Byte)	Chi-square Test Value
loadfix.com	1131	1131	31305486336.0
ReadMe.txt	286	286	276128064.00
WINSTUB.EXE	578	578	16863553536.0
VIAPCL.SYS	2712	2712	15576862720.0
iconlib.dll	2560	2560	18683279360.0

V. CONCLUSION

In this paper a new private-key encryption scheme is proposed which is based on bitwise XOR operation between the plain text and the secret value derived from the private-key.

The secret value is being calculated depending upon the number of '0' and '1' of a single character in the plain text and the different operators inputted by the user. There is several numbers of secret key values for a plain text file as the number of '0' and '1' are different for several numbers of distinct characters in the plain text file. Thus provide a great security.

Besides this, a user can also do the encryption or decryption by using different types of operators in the private-keys allotted for specific numbers of user defined blocks in a plain text file. So the security is increased as different keys are being used for encryption for different portion of plain text file, The size of the encrypted or decrypted file is same as of the plain text file. So we don't need any additional memory for storing the encrypted or decrypted file.

The execution time is depends on the file size not on the type of the file as we have done the encryption in bit level. So, the proposed scheme is better in respect of providing security for encryption, encrypted file size management, encryption or decryption time requirement.

REFERENCES

1. J. K. Mandal, S. Dutta, "A 256-bit recursive pair parity encoder for encryption", Advances D -2004, Vol. 9 n°1, Association for the Advancement of Modeling and Simulation Techniques in Enterprises (AMSE, France), www. AMSE-Modeling.org, pp. 1-14
2. William Stallings, Cryptography and Network security: Principles and practice (Second Edition), Pearson Education Asia, Sixth Indian Reprint 2002.
3. Atul Kahate (Manager, i-flex solution limited, Pune, India), Cryptography and Network security, Tata McGraw-Hill Publishing Company Limited.
4. Mark Nelson, Jean-Loup Gailly, The Data Compression Book. BPB Publication
5. Saurabh Dutta, "An Approach towards Development of Efficient Encryption technique", A thesis submitted to the University of North Bengal for the Degree of Ph.D., 2004.

AUTHOR PROFILE



Ramkrishna Das, M.Tech (Computer Science of Engineering) MCA, BCA, Research experience 1 year, Seminar Attend- 2, Foreign Journal Publications - 2. Specialization: Visual cryptography, Information Security and Cryptology



SAURABH DUTTA, PhD. (computer science), MCA, B.Sc. (Mathematics), Publication of book-2, Research Papers publications: 45, Foreign Journal Publications - 13 Peer-Reviewed International Conference Papers - 8, Indian Journal Publications -2, Published/Accepted in International Conference -7, others -3.National Conference Publications -12. Research Experience- 12 years Specialization: Information Security and Cryptology Member-IACSIT, IJIST, IEDRC, , IJISMRD, IJMCAR, IJCSS, IJMER, ISOC, IAENG, IJCSI. Reviewer-(IJNS), Taiwan AMSE, France, IJACSA, ICCIA, ACCT12, Award- Enlisted in the directory of Marquis Who's Who in the World in its 2010 edition. Biography selected for inclusion by ABI (American Biographical Institute, Inc.) in the list of International Profiles of Accomplished Leaders in its 2011 edition