Volume 2, Issue 8, August 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Efficiency and Security of Data with Symmetric Encryption Algorithms

Ritika Chehal
Student M.Tech. Computer Science and Engg.
Jind Institute of Engineering & Technology
Jind, Haryana, India

Kuldeep Singh
Asst. Prof. Department of Computer Science and Engg.
Jind Institute of Engineering & Technology
Jind, Haryana, India

Abstract— In this paper, we proposed the encryption algorithm to encrypt plaintext to cipher text and the decryption algorithm to do the reverse. We applied the basic computing operations to design these algorithms in this study. We used the inserting dummy symbols, rotating, transposition, shifting, complement and inserting control byte to build the data and tables in the encryption algorithm. These operations are simple and easily to implement. Without knowing the data and tables of encryption, it is difficult to do cryptanalysis. In the decryption algorithm, we used these data and tables to decrypt cipher text to plaintext. We can easily apply these algorithms to transmit data in network and the data transmission is secure.

Keywords— Encryption; Decryption; Data transmission; Cipher text; Plaintext; Encryption techniques

I. INTRODUCTION

Many encryption algorithms are widely available and used in information security. They can be categorized into Symmetric (private) and Asymmetric (public) keys encryption. In Symmetric keys encryption or secret key encryption, only one key is used to encrypt and decrypt data. The key should be distributed before transmission between entities. Keys play an important role. If weak key is used in algorithm then everyone may decrypt the data. Strength of Symmetric key encryption depends on the size of key used. For the same algorithm, encryption using longer key is harder to break than the one done using smaller key. Asymmetric key encryption or public key encryption is used to solve the problem of key distribution. In Asymmetric keys, two keys are used; private and public keys. Public key is used for encryption and private key is used for decryption (E.g. RSA and Digital Signatures). Because users tend to use two keys: public key, which is known to the public and private key which is known only to the user. There is no need for distributing them prior to transmission. However, public key encryption is based on mathematical functions, computationally intensive and is not very efficient for small mobile devices [1]. Asymmetric encryption techniques are almost 1000 times slower than Symmetric techniques, because they require more computational processing power [2].The most common classification of encryption techniques Brief definitions of the most common encryption techniques are given as follows: encryption standard to be recommended by NIST (National Institute of Standards and Technology).DES is (64 bits key size with 64 bits block size). Since that time, many attacks and methods recorded the weaknesses of DES, which made it an insecure block cipher [3], [4]. Cryptography is the mathematical manipulation of data for the purpose of reversible or irreversible transformation [8]. A piece of data, called clear text or plaintext, is transformed into a piece of data called cipher text. This process is called encryption, the reverse process decryption.

II. KEY - BASED ALGORITHMS

In the previous page we saw a rather simple encryption algorithm which simply substituted each letter in a message by the next one in the alphabet. The decryption algorithm was, of course, substituting each letter in the encrypted message with the previous letter in the alphabet [3]. These kinds of algorithms, based on the substitution of letters, are easily broken. Most modern algorithms, however, are key-based. A key-based algorithm uses an encryption key to encrypt the message. This means that the encrypted message is generated using not only the message, but also using a 'key':

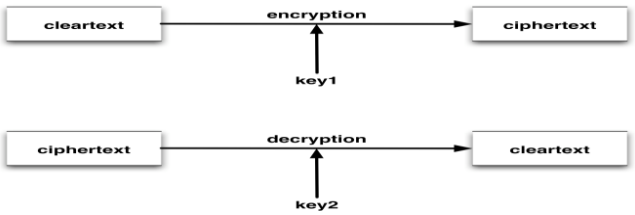

Fig.1 Key-based encryption and description

The receiver can then use a decryption key to decrypt the message. Again, this means that the decryption algorithm doesn't rely only on the encrypted message. It also needs a 'key'.

III. Symmetric and Asymmetric Key – Based Algorithm

Although this type of algorithms is generally very fast and simple to implement, they also have several drawbacks. The main drawback is that they only guarantee privacy (integrity and authentication would have be done some other way). Another drawback is that both the sender and the receiver need to agree on the key they will use throughout the secure conversation (this is not a trivial problem). Secure systems nowadays tend to use asymmetric algorithms, where a different key is used to encrypt and decrypt the message. Public-key algorithms, which are introduced in the next section, are the most commonly used type of asymmetric algorithms**.**

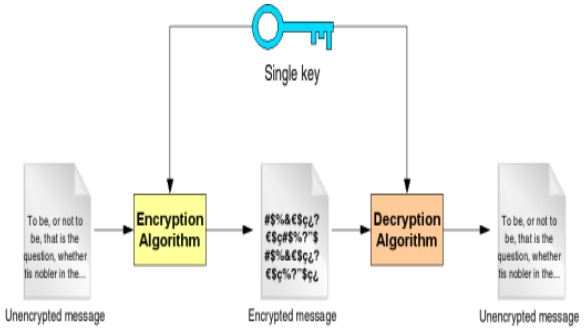

Fig. 2 Key-based symmetric algorithm

IV. Symmetric Key Encryption

Symmetric key encryption uses same key, called secret key, for both encryption and decryption. Users exchanging data keep this key to themselves. Message encrypted with a secret key can be decrypted only with the same secret key. The algorithm used for symmetric key encryption is called secret-key algorithm. Since secret-key algorithms are mostly used for encrypting the content of the message they are also called content-encryption algorithms. [4]The major vulnerability of secret-key algorithm is the need for sharing the secret-key. One way of solving this is by deriving the same secret key at both ends from a user supplied text string (password) and the algorithm used for this is called password-based encryption algorithm. Another solution is to securely send the secret-key from one end to other end. This is done using another class of encryption called asymmetric algorithm, which is discussed later. Strength of the symmetric key encryption depends on the size of the key used. For the same algorithm, encrypting using longer key is tougher to break than the one done using smaller key. Strength of the key is not liner with the length of the key but doubles with each additional bit.

Following are some of popular secret-key algorithms and the key size that they use:

TABLE1
SOME POPULAR KEYS

Key Name	Key Size(bits)
RC2	64
DES	64
3DES	192
AES	256
IDEA	128

V. Asymmetric Key Encryption

Asymmetric key encryption uses different keys for encryption and decryption. These two keys are mathematically related and they form a key pair. One of these two keys should be kept private, called private-key, and the other can be made public (it can even be sent in mail), called public-key. Hence this is also called Public Key Encryption. A private key is typically used for encrypting the message-digest; in such an application private-key algorithm is called message-digest encryption algorithm. A public key is typically used for encrypting the secret-key; in such an application private-key algorithm is called key encryption algorithm.

VI. Related Work

To give more prospective about the performance of the compared algorithms, this section discusses the results obtained from other resources. It was shown in [1] that energy consumption of different common symmetric key

encryptions on handheld devices. It is found that after only 600 encryptions of a 5 MB file using Triple-DES the remaining battery power is 45% and subsequent encryptions are not possible as the battery dies rapidly. It was concluded in [2] that AES is faster and more efficient than other encryption algorithms. When the transmission of data is considered there is insignificant difference in performance of different symmetric key schemes (most of the resources are consumed for data transmission rather than computation). Even under the scenario of data transfer it would be advisable to use AES scheme in case the encrypted data is stored at the other end and decrypted multiple times [9]. Increasing the key size by 64 bits of AES leads to increase in energy consumption about 8% without any data transfer. The difference is not noticeable. Reducing the number of rounds leads to power savings but it makes the protocol insecure for AES and should be avoided. Seven or more rounds can be considered fairly secure and could be used to save energy in some cases. A study in [5] is conducted for different popular secret key algorithms such as DES, 3DES, AES, and Blowfish. They were implemented, and their performance was compared by encrypting input files of varying contents and sizes. The algorithms were tested on two different hardware platforms, to compare their performance. They had conducted it on two different machines: P-II 266 MHz and P-4 2.4 GHz. The results showed that Blowfish had a very good performance compared to other algorithms. Also it showed that AES had a better performance than 3DES and DES. It also shows that 3DES has almost 1/3 throughput of DES, or in other words it needs 3 times than DES to process the same amount of data [6], [10]. In [7] a study of security measure level has been proposed for a web programming language to analyse four Web browsers. This study consider of measuring the performances of encryption process at the programming language's script with the Web browsers. This is followed by conducting tests simulation in order to obtain the best encryption algorithm versus Web browser.

VII. SIMULATION RESULT

Simulation results of encryption algorithms with different packets size as shown in Table 2.

TABLE 2
COMPARATIVE EXECUTION TIMES (IN MILLISECONDS) OF ENCRYPTION ALGORITHMS WITH DIFFERENT PACKET SIZE

Input size (Kbytes)	AES	3DES	DES	RC6	Blow fish	RC2
49	56	54	29	41	36	57
59	38	48	33	24	36	60
100	90	81	49	60	37	60
247	112	111	47	77	45	121
321	164	167	82	109	45	168
694	210	226	144	123	96	262
899	258	299	248	162	64	268
963	208	283	250	125	66	295
5345.28	1237	1466	1296	695	122	1570
7310.336	1366	1786	1695	756	107	1915
Time average	347	452	389	217	60.3	480.7
Throughput (Megabyte)	4.174	3.45	4.01	7.19	25.892	3.247

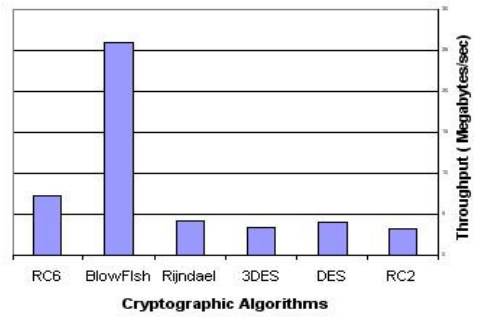

Fig. 3 Throughput of each encryption algorithm (Megabyte/Sec)

VIII. CONCLUSIONS

This paper presents a performance evaluation of selected symmetric encryption algorithms. The selected algorithms are AES, DES, and 3DES, RC6, Blowfish and RC2. Several points can be concluded from the simulation results. First; 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 Blowfish has better performance than other common encryption algorithms used, followed by RC6. Third; in the case of changing data type such as image instead of text, it was found that RC2, RC6 and Blowfish has disadvantage over other algorithms in terms of time consumption. Also, we find that 3DES still has low performance compared to algorithm DES. 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] Ruangchaijatupon, P. Krishnamurthy, *Encryption and Power Consumption in Wireless LANs-N*, The Third IEEE Workshop on Wireless LANs – September 27-28, 2001- Newton, Massachusetts.

[2] Hardjono, *Security In Wireless LANS And MANS,* Artech House Publishers 2005.

[3] W.Stallings, *Cryptography and Network Security 4th Ed,* Prentice Hall, 2005,PP. 58-309 .

[4] Coppersmith D, *The Data Encryption Standard (DES) and Its Strength Against Attacks*, I BM Journal of Research and Development, May 1994, pp. 243 - 250.

[5] Bruce Schneier, *The Blowfish Encryption Algorithm*, Retrieved October 25, 2008

[6] K. Naik, D. S.L. Wei, *Software Implementation Strategies for Power-Conscious Systems*, Mobile Networks and Applications - 6, 291-305, 2001.

[7] Daemen, J., and Rijmen, V. Rijndael*: The Advanced Encryption Standard.,* D r. Dobb's Journal, March 2001, PP. 137-139.

[8] D. Denning, *Cryptography and Data Security*, Addison-Wesley, 1982

[9] W. Diffie, M. E. Hellman, *New Directions in Cryptography*, IEEE Trans. On Inform. Theory, pp. 644-654, 1976.

[10] E. Biham and A. Shamir, *Differential Cryptanalysis of DES-like Cryptosystem*, Advances in Cryptology-CRYPTO'90 Proceedings, Berlin: Springer-Verlag, pp. 2-21, 1991

